

Automatically create an issue link after issue creation on email by "Enterprise Mail Handler for Jira" app

On this page

- [Features used to implement the example](#)
- [Example: Automatically create an issue link after issue creation on email by "Enterprise Mail Handler for Jira" app](#)
- [Other examples of that function](#)
- [Related Usage Examples](#)

Features used to implement the example

- [Create issue link](#)

Example: Automatically create an issue link after issue creation on email by "Enterprise Mail Handler for Jira" app

We are currently using the Jira app [Enterprise Mail Handler for Jira](#) to create tickets based on emails in **project A**. When it creates a ticket, it fills in the summary, description, and custom field "**ClientEmail**". We have a second **project B** (called "**Personal Directory**"), that has tickets with unique data in each **ClientEmail** field.

We want to use [Jira Workflow Toolbox](#) to set up a post-function on **project A** workflow to link tickets to **project B** (called "**Personal Directory**"), based on each unique email address in field **ClientEmail**. I have been successfully created links based on JQL, but it was for a specific example (see screenshot below). We want [Jira Workflow Toolbox](#) to search for data in **ClientEmail**, and link issue in project A to ticket in project B that already has that same data in its field **ClientEmail**.

The screenshot shows the Jira Workflow Toolbox configuration page for the 'Create issue link' function. The page has a navigation bar at the top with links: Projects, Add-ons, User Management, Issues, System, and Audit Log. The main heading is 'Update parameters of the Create issue link Function for this transition.' Below this, there is a section 'Update parameters of the Create issue link Function for this transition.' with a sub-section 'Issues to be linked:'. This section contains a checkbox 'Field containing issue keys' which is checked. Below it, a text box contains the JQL query: 'ClientEmail ~ "gregbrailsford@gmail.com" AND project = "Personal Directory"'. To the right of this text box is a dropdown menu labeled 'Summary' with a downward arrow. Below the text box, there are three bullet points: '- Field codes with format %(name) may be inserted in the JQL Query and will be replaced with field values at execution time.', '- You can reference parent and child values of cascading select fields writing %(parent.X) for parent value, and %(parent.Y) for child value.', and '- You can also insert values of multi-level cascading select fields writing %(parent.Y), with 1 being the level to be read (root level is 0)'. Below these bullet points is a link 'Available field codes'. At the bottom of the page, there is a section 'Issue link type to be created:' with two radio buttons: 'Is blocked by' (selected) and 'Blocks'.

Instead of a specific value, you should use field code for **ClientEmail** field. You can find its field code in the field list you can find below the configuration parameters of post-function [Create issue link](#). In the following screenshots I show you how to do it. Notice that in my example field **ClientEmail** has field code `%(13600)`, but it depends on each specific Jira instance. Configuration of post-function [Create issue link](#):

Issues to be linked:

☐ Field containing issue keys
This field must contain a comma or blank separated list of issues keys, e.g., "CRM-1, CRM-2, CRM-3" or "CRM-1 CRM-2 CRM-3".

☒ Issues returned by a JQL query
Issues returned by entered JQL query will be linked to current issue.

Summary

ClientEmail ~ %{13600} AND project = "Personal Directory"

- Field codes with format `%{nnnnn}` may be inserted in the JQL Query and will be replaced with field values at execution time.

- You can reference parent and child values of cascading select fields writing `%{nnnnn.0}` for parent value, and `%{nnnnn.1}` for child value.

- You can also insert values of multi-level cascading select fields writing `%{nnnnn.l}`, with l being the level to be read (root level is 0).

Available field codes

Issue link type to be created:

☐ is blocked by
☐ blocks
☐ is cloned by
☐ clones
☐ is duplicated by
☐ duplicates
☒ relates to
☐ relates to

Once configured, transition configuration will look like this:

Conditions 0

Validators 0

Post Functions 7

The following will be processed after the transition occurs

Add post function

1. Current issue will be linked using issue link **relates to** to every issue returned by the following JQL query: **ClientEmail ~ #ClientEmail# AND project = "Personal Directory"**

I show you a screenshot showing a part of the field list showing the field code for field "ClientEmail":

%{12600}	Carrier	Text Field (single line)
%{10900}	City	Select List (single choice)
%{10901}	City in Country	Select List (cascading)
%{13600}	ClientEmail	Text Field (single line)
%{12101}	Code review priority	Select List (single choice)
%{10100}	Colours	Checkboxes
%{12700}	Consequence of Bug	Text Field (single line)

Other examples of that function

Page: [Automatically create an issue link after issue creation on email by "Enterprise Mail Handler for Jira" app](#)

Related Usage Examples

- [Validate only issue links created in transition screen](#)
 - [example](#)
 - [validator](#)

Page: Create issue links based on a custom field value avoiding duplicates

Page: Creating issue links to issues with the same "Summary"

Page: Parse description for creating issue links

Page: Replace certain issue link types with different ones

- issue-links
- Require issue link when resolving as duplicate
 - example
 - validator
 - issue-links
- Ensure that all issues linked with a certain issue link type have "Due Date" field set
 - example
 - validator
 - issue-links
- Block an epic's transition depending on linked issues status and due date
 - example
 - validator
 - issue-links
 - transition
- Add and remove a single or a set of items from multi valued fields
 - example
 - post-function
 - custom-field
 - issue-links
 - sub-task
- Writing a comment to blocked issues when blocking issues are resolved
 - example
 - post-function
 - issue-links
- Prevent issue from moving forward if it's dependent on non-accepted tickets
 - example
 - validator
 - issue-links
 - transition
- Enforce linked issues in a specific project to be "Closed" before closing issue
 - example
 - validator
 - issue-links
 - transition
- Block or hide a transition for an issue depending on its issue links
 - example
 - validator
 - issue-links
 - transition
- Prevent transitioning when there is a blocking issue
 - example
 - validator
 - issue-links
 - sub-task
 - transition
- Prevent issue from being "Closed" if blocking issues aren't yet closed
 - example
 - validator
 - issue-links
 - transition
- Block creation of issue type X if it has not been linked with link type Y to issue type Z on the "Create Issue" screen
 - example
 - validator
 - issue-links
- Prevent issue from being closed if it has links of type "is blocked by" to open issues
 - example
 - condition
 - validator
 - issue-links
 - transition
- Transition linked issues in currently active sprint
 - example
 - post-function
 - issue-links
 - transition
- Parse description for creating issue links
 - example
 - post-function
 - issue-links

