

# Botron Configuration Manager for Jira

## On this page

[Prerequisites](#) | [Limitations](#) | [issuesFromJQL](#) | [Modules](#)



Botron's [Configuration Manager for Jira](#) is a powerful tool to seamlessly **export** and **import** Jira **configurations** between server instances.

As of release 2.8.0 JWT for Jira Server and Data Center integrates with Configuration Manager and **all workflow elements** (conditions, validators and post functions and parser expressions) and **custom fields** provided by JWT are officially supported with minimal [limitations](#).

## Prerequisites

- JWT for Jira Server and Data Center (2.8.0 and above) must be installed on both the source and target instance
- Botron Configuration Manager for Jira must be installed on both the source and target instance

## Limitations

JWT uses [virtual fields](#) and [field codes](#) to represent Jira data. All field codes will be correctly updated to their equivalent value when exporting a configuration to the target instance.

There is one place though, where users might not use a field codes but a static custom field reference - in a JQL query, or more precisely, in the **issue sFromJQL** parser function.

## issuesFromJQL

The issuesFromJQL parser function, when used in the **advanced parsing mode** cannot automatically transform the custom field ids/names used in the query.

### Example

Custom field IDs:

Source instance	Destination instance	Comment
customfield_10301	customfield_10481	If the source and the target instance do not contain the exact same custom field configuration (e.g. a custom field is created during the import process), it is very likely that the IDs of those field do not match in the source and target instance.

```
issuesFromJQL("cf[10301] ~ \"\" + {%10301} + \"\" AND \"Botron single text\" ~ \"\" + {%10301} + \"\")
```

will be changed to

```
issuesFromJQL("cf[10301] ~ \"\" + {%10481} + \"\" AND \"Botron single text\" ~ \"\" + {%10481} + \"\")
```

You can see that the **field codes** were replaced but the custom field ids/names have to be updated manually as part of a post-migration process.



We are constantly in touch with Botron to hopefully overcome the limitations in the foreseeable future.

## Modules

If you have **not** installed Configuration Manager for Jira, you will notice some disabled JWT modules by default.

This is the standard behavior and you do not have to worry about it.

<b>Configuration Manager SPI Implementation for Calculated Text Field</b>	(calculated-text-field-spi-configuration-handler)
<b>Configuration Manager SPI Implementation for Calculated Number Field</b>	(calculated-number-field-spi-configuration-handler)
<b>Configuration Manager SPI Implementation for Calculated Date-Time Field</b>	(calculated-date-field-spi-configuration-handler)
<b>Configuration Manager SPI Implementation for JWT Conditions</b>	(com.fca.jira.plugins.workflowToolbox-spi-workflow-participant-condition-handler)
<b>Configuration Manager SPI Implementation for JWT Validators</b>	(com.fca.jira.plugins.workflowToolbox-spi-workflow-participant-validator-handler)
<b>Configuration Manager SPI Implementation for JWT Post-functions</b>	(com.fca.jira.plugins.workflowToolbox-spi-workflow-participant-postfunction-handler)