

Migrate JWT DC to JWT Cloud

On this page

[Cloud availability and migration instructions](#) | [Webinars](#) | [Changelog](#)

Introduction

Considering your move to the Cloud? Make sure to check out our [automated app migration path](#).

For readability reasons, we only refer to **Data Center** or **DC** on the following pages, but explicitly include **Server** here unless otherwise mentioned.

Migrating from Data Center to Cloud is not a trivial task - and JWT's configuration is no exception. Basically, workflow extensions provided by 3rd party apps are incompatible between Data Center and Cloud due to fundamental technical differences between those two platforms.

One of the main challenges in this area is the general availability of features on both sides. JWT DC has grown steadily being among the first apps on the Atlassian Marketplace, and some features became since then, obsolete or have been replaced by newer, more powerful functions. However, we provide all the information needed to adequately represent all functions and thus establish feature parity between Data Center and Cloud. In addition, it's important to understand we have not and will not bring all features or single functions to the Cloud.

So what is the current state?



With our latest release of JWTC, paired with the release of JWT DC 3.3.0, we're more than happy to announce that JWT now comes with an [automated app migration path](#).



In this documentation, we describe which JWT workflow functions are available in Jira Cloud as well and how existing JWT DC workflow functions can be migrated.

Not all workflow functions known in JWT DC are available in JWT Cloud. The main reason being the different architecture of Jira Cloud as described in [A general note about JWT for Jira Cloud](#), e.g. you can't rely on an execution order of post functions when transitioning an issue or you only can use Jira expressions in conditions and validators. Besides that, the Cloud platform itself is not as mature as the DC platform which means that JWT Cloud app does not yet have the maturity its DC variant has. The entire team is working hard on continuously closing or at least narrowing the gap with JWT DC.

In the further course of this page, you will find a list of all JWT DC functionalities and how they may or may not be migrated to JWT Cloud.



Cloud availability and migration instructions

Conditions



 JWT DC	 JWT Cloud availability	Notes
Compare two values condition	✓	See Migrate 'Compare two values condition' .
Condition on a list of issue keys	-	<div>This function is deprecated in JWT DC and is thus not covered in this manual.</div>
Condition based on cascading select list value	⚠	Mappable by Jira expression condition : see Migrate 'Condition based on cascading select list value' .
Condition based on JQL query	-	

Condition based on regular expression	⚠	Mappable by Jira expression condition : see in Migrate 'Condition based on regular expression' .
Condition on JWT project property	-	
Condition on linked issues	⚠	Mappable by Jira expression condition : see Migrate 'Condition on linked issues' .
Condition on sub-tasks	⚠	Mappable by Jira expression condition : see Migrate 'Condition on sub-tasks'
Except assignee	⚠	Mappable by Jira expression condition : see Migrate 'Except assignee' .
Except reporter	⚠	Mappable by Jira expression condition : see Migrate 'Except reporter' .
Except users in a field	⚠	Mappable by Jira expression condition : see Migrate 'Except users in a field' .
Fields required	✓	See Migrate 'Fields required' .
Hide transition from user (bulk operation only)	-	
Hide transition from user (JWT function only)	-	
Logical condition	⚠	Mappable by Jira expression condition : see Migrate 'Logical condition' .
Only users in a field	⚠	Mappable by Jira expression condition : see Migrate 'Only users in a field' .
User is not in project role	⚠	Mappable by Jira expression condition : see Migrate 'User is not in project role' .
Users are/aren't in project role (condition)	-	

Validators

 JWT DC	 JWT Cloud availability	Notes
Compare two values validator	✓	See Migrate 'Compare two values validator' .
Fields required or changed	✓	See Migrate 'Fields required or changed' .
Logical validator	⚠	Mappable by Jira expression validator : see Migrate 'Logical validator' .
Users are/aren't in project role (validator)	-	
Validation based on JQL query	-	
Validation based on regular expression	⚠	Mappable by Jira expression validator : see Migrate 'Validation based on regular expression' .
Validation of JWT project property	-	
Validation of linked issues	⚠	Mappable by Jira expression validator : see Migrate 'Validation of linked issues' .
Validation of sub-tasks	⚠	Mappable by Jira expression validator : see Migrate 'Validation of sub-tasks' .
Validation on a list of issue keys	-	<div>This validator is deprecated in JWT DC and is thus not covered in this manual.</div>

Post functions

 JWT DC	 JWT Cloud availability	Notes
--	--	-------

Add comment	✓	See Migrate 'Add comment' .
Add or disable option in (multi-) select list, radio button, or checkbox field	-	
Add or disable option in cascading select list field	-	
Add or remove watchers	-	
Assign to project role	-	
Clone issue link	-	This function is deprecated in JWT DC and is thus not covered in this manual.
Copy a parsed text to a field	⚠	Mappable by Update fields : see Migrate 'Copy a parsed text to a field' .
Copy cascading select list value	-	
Copy excerpted value	-	
Copy JWT project property	-	
Copy JWT user property	-	
Copy or add custom field value to issue description	⚠	Mappable by Update fields : see Migrate 'Copy or add custom field value to issue description' .
Create issue	✓	See Migrate 'Create issue' .
Copy field values from multiple issues	-	
Copy field values from linked issues or sub-tasks	-	
Create issue link	-	
Delete issue link	-	
Format field value	-	
Inhibit temporary fields' auto-clearance	-	This function will be deprecated in JWT DC and is thus not covered in this manual.
Log work	-	
Mathematical or date-time expression calculator	⚠	Mappable by Update fields : see Migrate 'Mathematical or date-time expression calculator' .
Move issue	-	
Regular expression renderer	-	
Send email	✓	See Migrate 'Send email' .
Set issue reporter to current user	⚠	Mappable by Update fields : see Migrate 'Set issue reporter to current user' .
Set or create JWT project property	-	
Set or create JWT user property	-	
Transition issue	✓	See Migrate 'Transition issue' .
Update field based on rules	-	
Update linked issue or sub-task	✓	See Migrate 'Update linked issue or sub-task' .
Update or copy field values	✓	See Migrate 'Update or copy field values' .

Parser expressions

The JWT Cloud expression parser differs from the one used by JWT DC. So they need to be migrated like described in [Migrate JWT parser expressions](#), with detailed instructions related to [parsing modes](#), [field codes](#), and [parser functions](#).

Automation rules

Automation rules are and will not be part of JWT Cloud. The main reason is the availability of [Atlassian's own automation engine](#).

Calculated fields

Calculated fields are and will not be part of JWT Cloud. In order to address our target audiences even more precisely, we have decided to deliver our calculated fields functionality separately in the cloud in the form of [Smart Fields for Jira](#).

JWT calendars

JWT calendars are not part of JWT Cloud. This might change related to customer demand.

JWT JQL functions

JWT JQL functions are not part of JWT Cloud. This might change related to customer demand.

Temporary fields

Temporary fields are not part of JWT Cloud. This might change related to customer demand.

Webinars

Changelog

Date	Notes
14 Dec 2023	Webinar section added

22 Feb 2023	Extended due to introducing an automated app migration path
13 May 2022	Extended with the comprehensive list for manual migration of all relevant elements
01 Apr 2022	Updated due to the release of Smart Fields for Jira
15 Oct 2021	Initial public draft

If you still have questions, feel free to refer to our [support team](#).