

Performance settings

JWT is a **powerful tool** that can handle a lot of your Jira data! However, on large instances, JWT can have an impact on the overall **performance**, especially when performing complex or unoptimized calculations.

To stay on top of Jira's performance, you can **easily restrict JWT's CPU and memory usage!**

SINCE 3.1.4

The screenshot shows the 'Performance settings' page in Jira Administration. The page is divided into several sections:

- Maximum execution time:** Four sliders control the execution time for different areas: Automation rules (10 s), Calculated fields (10 s), JQL functions (30 s), and Workflow functions (10 s). Each slider has a refresh icon and a green checkmark.
- Memory Usage:** A text input field for 'Maximum number of issues used in expression parser functions' is set to 50000.
- JQL functions:** A text input field for 'Maximum number of issues returned by JQL subqueries' is set to 10000.
- Buttons:** 'Restore defaults' and 'Save' buttons are located at the bottom right.

Maximum execution time

You can define the maximum execution time of each JWT area **individually**.

All values are in **seconds**. If a calculation takes longer as the specified value, the below written rules will apply.

You can also **disable** the execution limit for each part. If you do so, you might experience **performance issues!** ⚠️ **It is therefore not recommended to turn off the execution limits!**

Area	Default value	Configuration
------	---------------	---------------

Automation rules	10s	<p>If the underlying calculation for a rule</p> <ul style="list-style-type: none"> • takes longer than the defined value or • has reached the memory usage (see below) <p>it will abort the whole calculation without returning a result and an error will be thrown. Information about the error will be written to the Jira log.</p>
Calculated fields	10s	<p>If the underlying calculation for a field</p> <ul style="list-style-type: none"> • takes longer than the defined value or • has reached the memory usage (see below) <p>it will abort the whole calculation without returning a result and an error will be thrown. Information about the error will be written to the Jira log.</p>
JQL functions	10s	<p>If the underlying calculation for a JQL function</p> <ul style="list-style-type: none"> • takes longer than the defined value or • has reached the memory usage (see below) or • has reached the subquery limit (see below) <p>it will abort the calculation. Only issues calculated at this point will be returned.</p> <p>In addition, an error will be displayed in the issue navigator. Information about the error will be written to the Jira log.</p>
Workflow functions	10s	<p>If the underlying calculation for a workflow function (condition, validator, post function)</p> <ul style="list-style-type: none"> • takes longer than the defined value or • has reached the memory usage (see below) <p>it will abort the whole calculation without returning a result and an error will be thrown. Information about the error will be written to the Jira log.</p>

Memory usage

In addition to time limits, you can define the **number of issues** that should be processed in a single calculation.

Memory usage

In order to save resources, you can set a limit for **issues to be processed in a single parser function**.

The default value is **50.000**.

JQL functions

Used for [JWT JQL functions](#) where you can define a JQL subquery, such as:

- [linkedTo\(\)](#)
- [issueSelection\(\)](#)

Setting a limit will **only limit the number of issues returned by the subquery**, not by entire JQL function itself.

The default value is **1.000**.

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