

How to Sync Issue Types in Jira Cloud

Last Modified on 01/15/2026 6:37 pm EST

This article describes how to synchronize issue types in Jira Cloud.

There are multiple use cases in issue type synchronization. With Exalate, you can handle any possible combination of the issue type mapping.

Source Side

Outgoing sync

To send the issue type, add the script below:

```
replica.type = issue.type
```

Destination Side

Incoming sync

There are different approaches to how you can set the issue type on the receiving side, depending on your specific use case. To add the issue type on a synced issue you should add the script to the incoming sync. Below you can find some of the most common cases:

- Set the issue type name from the source issue, if not found set a default one

```
// Set type name from source issue, if not found set 'Task' as the default target issue type
issue.type = nodeHelper.getIssueType(replica.type?.name, issue.projectKey ?: issue.project?.key) ?: nodeHelper.getIssueType("Task", (issue.projectKey ?: issue.project.key))
```

```
// Set type name from source issue without project key, if not found set 'Task' as the default target issue type
issue.type = nodeHelper.getIssueType(replica.type?.name) ?: nodeHelper.getIssueType("Task")
```

- Set the issue type name from the source issue

```
// Set type name from source issue without project key
issue.type = nodeHelper.getIssueType(replica.type?.name)
```

```
// Set type name from source issue with project key
issue.type = nodeHelper.getIssueType(replica.type?.name, issue.projectKey ?: issue.project?.key)
```

- Set the issue type according to the mapping, specified with the help of Groovy scripts

```
//map issue types between source and destination instances.  
def issueTypeMapping = [  
  // "remote issue type" : "local issue type"  
  "Bug" : "Problem",  
  "Task" : "Assignment"  
]  
  
issue.type = nodeHelper.getIssueType(issueTypeMapping[replica.type?.name] ?: nodeHelper.getIssueType("Assignment"))
```

Note: In [Basic connections](#), issue types are mirrored: tasks are synced as tasks, improvements as improvements, and so on. When you sync, for example, epic(or any other issue type that is not found on the destination side) Exalate creates either an issue with the [default issue type](#), if the default issue type is set in the project configuration, or **Task** if not

[Product](#)
[About](#)
[Release History](#) 
[Glossary](#) 
[API Reference](#) 
[Security](#) 

[Pricing and Licensing](#) 

Resources

[Subscribe for a weekly Exalate hack](#) 

[Academy](#) 

[Blog](#) 

[YouTube Channel](#) 

[Ebooks](#) 

Still need help?

[Join our Community](#) 

[Visit our Service Desk](#) 

[Find a Partner](#) 