How to Sync Urgency and Priority between ServiceNow and Jira

Last Modified on 01/28/2026 12:02 pm EST

This article shows examples of how to sync priority and urgency between ServiceNow and Jira instances.

Warning: Despite our best efforts, code can change without notice due to a variety of factors. If you encounter an issue in any of the code shown here and find that a specific block of code is not correct, or is causing errors, please check with the Community to find an updated version.

- Syncing priority between ServiceNow and Jira
- Syncing urgency between ServiceNow and a Jira custom field (deprecated after version 6.0.0)
- Syncing priority (deprecated after version 5.1.1)

Syncing Priority between ServiceNow and Jira

To exchange priority between ServiceNow and Jira you need to specify the mapping.

By default, it is not possible to sync priority from Jira to ServiceNow, because the priority field is read-only in ServiceNow. For more details, check the ServiceNow documentation.

As a workaround, you can create a custom field in ServiceNow and sync Jira priority there.

ServiceNow

Outgoing sync

```
replica.priority = entity.priorityValue
```

Jira

Incoming sync

```
//Priority sync
   def priorityMapping = [
     // Snow incident priority <-> Jira issue priority
       "1 - Critical": "High",
       "2 - High": "High",
       "3 - Moderate": "Medium",
       "4 - Low": "Low",
       "5 - Planning": "Lowest"
  ]
 // set default priority in case the proper priority could not be found
   def defaultPriority = "Low"
   def priorityName     = priorityMapping[replica.priority?.name] ?: defaultPriority // set default priority in case the pro
per urgency could not be found
   issue.priority = nodeHelper.getPriority(priorityName)
```

Syncing urgency between ServiceNow and a Jira Custom Field (deprecated after version 6.0.0)

DEPRECATED

This urgencyValue method is deprecated after **version 6.0.0**. You can use **entity.urgency** to sync urgency for ServiceNow. It becomes available starting from **version 5.4.0.**

This example shows how to sync ServiceNow urgency to an **Urgency** select list custom field in Jira.

ServiceNow

Outgoing sync

```
replica.urgency = entity.urgencyValue
```

Incoming sync

```
 // Urgency sync
   def urgencyMapping = [
      // Jira issue urgency <-> Snow incident urgency
       "High": "1 - High",
       "Critical": "1 - High",
       "Medium": "2 - Medium",
       "Low": "3 - Low"
   ]
   def defaultUrgency = "3 - Low"
   def urgencyName     = urgencyMapping[replica.urgency.value] ?: defaultUrgency // set default urgency in case the
proper urgency could not be found
   entity.urgencyValue = urgencyName
```

Jira

Outgoing sync

```
replica.urgency = issue.Urgency
```

Incoming sync

```
// //Urgency sync
     def urgencyMapping = [
      // Snow incident urgency <-> Jira issue urgency
         "1 - High": "High",
         "1 - High": "Critical",
         "2 - Medium": "Medium",
         "3 - Low": "Low"
  ]
    def defaultUrgency = "Low"
    def urgencyName     = urgencyMapping[replica.urgency] ?: defaultUrgency // set default urgency in case the proper
 urgency could not be found
    issue.Urgency = urgencyName
```

Syncing Priority (deprecated after version 5.1.1)

DEPRECATED

Because of a bug, this method syncs priority as urgency. This method was deprecated after version 5.1.1. For versions after 5.1.1, it is preferable to use the other methods, mentioned in this article.

ServiceNow

Outgoing sync

```
replica.priority = issue.priority
```

Incoming sync

```
def priorityMapping = [
       // Jira issue priority <-> SNOW incident urgency
         "High" : "1 - High",
         "Highest" : "1 - High",
         "Medium" : "2 - Medium",
         "Low" : "3 - Low",
         "Lowest" : "3 - Low"
   ]
def defaultPriority = "3 - Low"
def priorityName     = priorityMapping[replica.priority?.name] ?: defaultPriority // set default urgency in case the prop
er urgency could not be found
entity.priority = nodeHelper.getPriority(priorityName)
```

Jira

Outgoing sync

```
replica.priority = issue.priority
```

Incoming sync

Set local priority based on the received urgency from the Servicenow side. If the priority name does not exist on the local side - set default priority to "Low".

```
def priorityMapping = [

        // SNOW incident urgency <-> Jira issue priority
          "1 - High" : "High",
          "2 - Medium" : "Medium",
          "3 - Low" : "Low"
    ]
def priorityName = priorityMapping[replica.priority?.name] ?: "Low" // set default priority in case the proper urgency could not be found
issue.priority = nodeHelper.getPriority(priorityName, issue)
```

ON THIS PAGE

Product

About Us ⧉

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 ⧉