Jira and Zendesk Integration

Last Modified on 01/28/2026 12:35 pm EST

Companies using Jira can collaborate with teams or organizations using Zendesk to improve the quality of product and service delivery.

This would involve getting both systems to interact with each other and then syncing the correct fields and entities to get data flowing based on configurations and use case requirements.

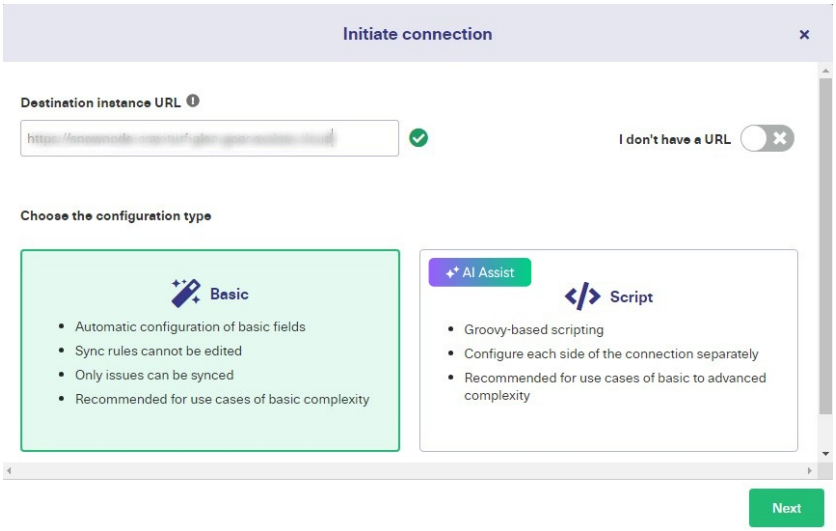
A tool like Exalate allows you to set up a Jira Zendesk integration with minimal fuss. It is also compatible with Jira Service Management, Jira Software, and Jira Work Management.

You can install Exalate on Jira Cloud or Jira on-premise or even set up a private connection in case any of your instances are behind a firewall. You can also deploy Exalate for Zendesk and Jira on Docker.

Exalate Installation and Configuration Steps

Here is a brief overview of how to configure an integration between Zendesk and Jira.

1. Install the Exalate app for Zendesk from the Marketplace. You can get started from the integrations page.
2. Install the Exalate app on Jira. You can start from the Atlassian marketplace.
3. Choose the connection type: Basic or Script.



To learn how to set up a connection between Jira and Zendesk using the Script mode, go to the Getting Started guide for a detailed visual breakdown. Just choose the two platforms you want to configure, and you'll get a comprehensive explanation of how to proceed.

Advanced Jira and Zendesk Integration Use Cases (Using AI Assist)

Exalate allows you to set up basic syncs between standard fields and entities on both Jira (Cloud and On-premise) and Zendesk.

But if you want to explore advanced connection scenarios, Exalate can also help you script connections quickly and accurately.

Exalate also supports AI-powered Jira integration with Zendesk thanks to the scripting engine in

the configuration panel.

It also provides several Script Helpers to reduce the effort of scripting connections from scratch. For a step-by-step breakdown, check out a detailed Jira Zendesk integration guide.

Here are some use cases for Zendesk integration with Jira.

Use Case 1: Map and Sync Statuses between Jira and Zendesk

If you want the status of a Jira issue to be reflected in the corresponding Zendesk ticket, you need a scripting rule to make sure the correct statuses are mapped on both sides.

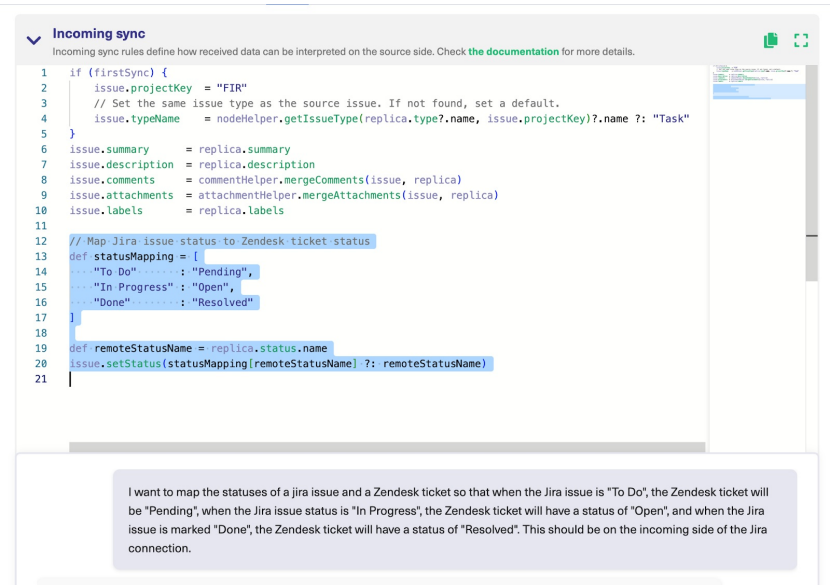
This will help teams working on the same project to align workflows and become more productive.

For instance, if the Jira status is To Do, the Zendesk ticket status changes to Pending. When the Jira status is In Progress, the Zendesk incident should be Open. When the Jira issue is Done, the Zendesk ticket should be marked Solved.

Enter the code snippet or script mapping you want for your use case, or use AI Assist to generate the code by typing in a detailed prompt describing what you want to sync.

The prompt to the AI Assist can be something like this:

> *"I want to map the statuses of a Jira issue and a Zendesk ticket so that when the Jira issue is "To Do", the Zendesk ticket will be "Pending", when the Jira issue status is "In Progress", the Zendesk ticket will have a status of "Open", and when the Jira issue is marked "Done", the Zendesk ticket will have a status of "Resolved". This should be on the incoming side of the Jira connection."*



Go through the generated output to confirm if it aligns with your expectations. You can continue refining the prompt until you nail down the requirements for your Zendesk and Jira integration.

```
def statusMapping = [
   "To Do"       : "Pending",
   "In Progress" : "Open",
   "Done"        : "Resolved"
]

def remoteStatusName = replica.status.name
issue.setStatus(statusMapping[remoteStatusName] ?: remoteStatusName)
```

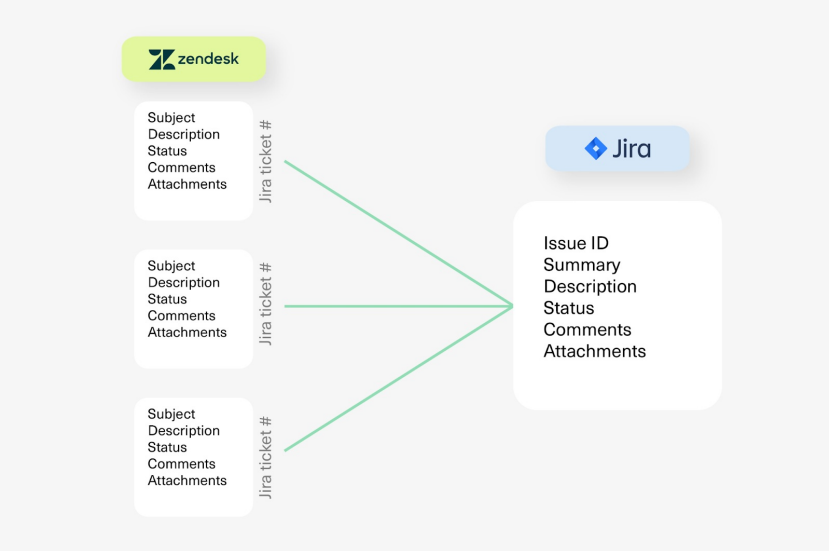
Click Discard if the generated code is incorrect. If the generated script is correct, click Insert Changes. Once you're satisfied with the scripting, then click Publish to save and implement changes.

Your browser does not support HTML5 video.

Note: Review the AI Assist prompting guidelines to improve your prompts and output.

Use Case 2: Sync Custom Fields to Merge Multiple Zendesk Tickets to one Jira Issue

If your sync requirements go beyond default fields and entities, then you need a Zendesk Jira integration API to help you fetch and transform data from custom fields.



To sync multiple Zendesk tickets to one Jira issue, you need to set up a custom field in Jira to store the Jira issue key.

Here is the code snippet generated by AI Assist for the Jira incoming sync.

```
replica.customFields."Target Issue Key" = issue.customFields."Target Issue Key"
```

You can discard, accept, or refine the prompt to nail down the specifics of your use case.

Next, you need to write a script to fetch data from all linked tickets based on their respective keys stored in the custom field.

The code snippet in the Jira outgoing sync should look like this:

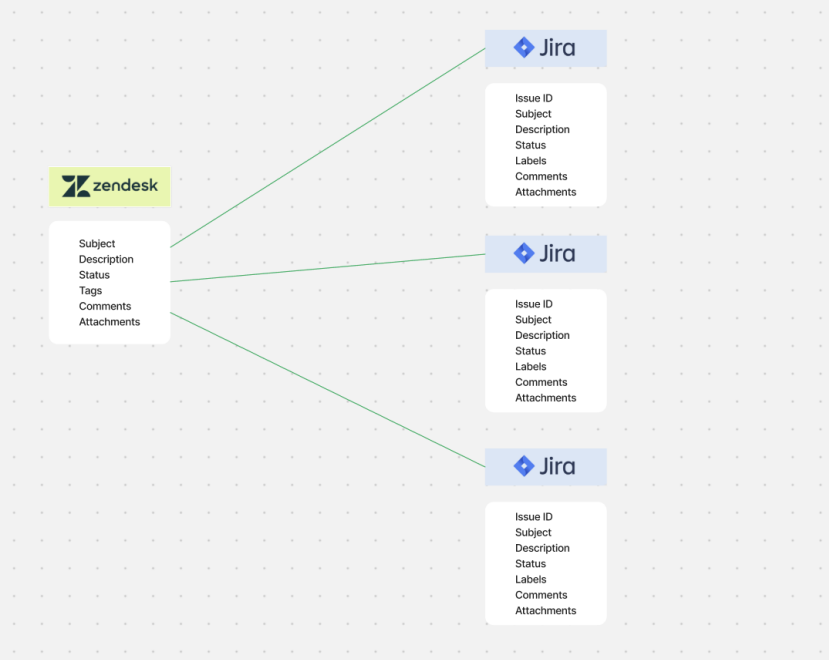
```
def remoteIssueUrn = replica.customFields."ABC-123"?.value


if(remoteIssueUrn && firstSync){
def localIssue = httpClient.get("/rest/api/2/issue/"+remoteIssueUrn)


if(localIssue == null) throw new com.exalate.api.exception.IssueTrackerException("Issue with key "+remoteIssueUrn+" was not found"
)

issue.id = localIssue?.id
issue.key = localIssue?.key
return;
}
```

The Jira issue ID from the Zendesk custom field “Jira Ticket #” is saved in the remoteIssueUrn variable. The httpClient.get() method fetches the details of the Jira issue.

If the Jira issue is not found, then an "Issue not found" exception is raised. However, if the Jira issue exists, then set the Jira issue ID to the localIssue ID.

By the same token, you can sync multiple Jira issues to a single Zendesk instance. Say multiple teams are using a single Zendesk instance, but you want to sync with multiple Jira instances.

You can specify the fields from Zendesk to go to each Jira instance, as well as the automated triggers for the sync.



This requires simply mapping the ticket separately and setting trigger conditions to control and automate the synchronization.

Use Case 3: Sync Side Conversations between Jira and Zendesk

Within a Zendesk ticket, the agent can start a side conversation with a developer on the Jira side, which will create a separate issue geared toward that specific topic. Subsequent replies will appear as comments under the newly created issue.

To get this working, you need to set up macros and triggers in your Zendesk instance.

Use Case 4: Sync Priority between Zendesk and Jira

Your Zendesk Jira integration setup can help you automatically replicate the priority of tasks on

both sides to convey the importance of the ticket or issue.

Instead of manually changing the priority on both sides, a simple script can ensure that both systems obtain the updated priority level in real time.

Here is a sample prompt:

> *"I want to sync Jira issue priority to match the Zendesk priority so that the mapping will be "Critical" to "High", "Medium" to "Medium", and "Minor" to "Low"."*

Here is the code snippet generated by AI Assist for the Jira incoming side.

```
def priorityMapping = [
    "Critical" : "High",
    "Medium"   : "Medium",
    "Minor"    : "Low"
]

def remotePriorityName = replica.priority?.name
issue.priority = nodeHelper.getPriority(priorityMapping[remotePriorityName] ?: "Medium"
)
```

You can discard, accept, or refine the prompt to nail down the specifics of your use case.

Your browser does not support HTML5 video.

AI Assist, like any other AI, can make mistakes. So, try to be as precise and detailed as possible with your prompts.

If you have a specific use case to discuss, book a demo with our engineers.

Note: The code snippet might not work precisely as intended due to changes to the environment or other reasons. If you encounter any problems, contact us for clarification.

Automate Jira Zendesk Integration Using Triggers

Exalate uses Jira Query Language (JQL) to set trigger conditions on the Jira side.

```
project = PRO AND status = Open AND label = urgent
```

Any open projects with the name "PRO" and the label "urgent" will be synced automatically.

You can use the Zendesk search syntax to specify the filter query.

```
type:ticket status:open
```

All currently open Zendesk tickets will be synced when this trigger activates.

Supported Jira and Zendesk Entities

You can sync any entities or issues between Jira and Zendesk.

Check out the comprehensive list of supported Zendesk entities as well as the comprehensive list

of supported Jira Cloud and On-premise entities.

This is a sample mapping between Zendesk incidents and the Jira issues:

Zendesk ticket <> Jira issue

- summary/subject↔ summary
- description ↔ description
- priority ↔ priority
- state ↔ status
- requester ↔ reporter
- comments ↔ comments
- attachments ↔ attachments
- tags ↔ labels
- custom fields ↔ custom fields
- third-party plugin fields
- any field available via REST APIs

Video Tutorials

- Watch the installation and configuration tutorial for Jira and Zendesk.
- Watch the installation and configuration demo for all connectors.

Other resources

- Download the Jira Zendesk integration eBook.
- Talk to Aida, your AI-powered integration sidekick, and get answers to your questions faster.
- Check out the detailed security and architecture whitepaper.
- Visit the Exalate Academy to get access to learning materials.

Subscribe to ` to get email updates and expert tips about the product.

ON THIS PAGE

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