

TestRun Configuration

Last Modified on 01/15/2026 9:26 am EST

TestRun allows you to test your scripts without creating actual entities on the destination side. This feature lets you preview and compare the replica before making changes in production, ensuring your configuration works as expected.

How to Use TestRun

1. Create a **New Version** or click **Open latest Draft**.
2. Click **Edit** to start modifying your configuration.
3. Make any necessary changes to your sync rules.
4. Click **Start TestRun** to begin the simulation.

 exalate

ⓘ You're in draft mode now. You can edit and save and publish this version.

Scripts

Version 2

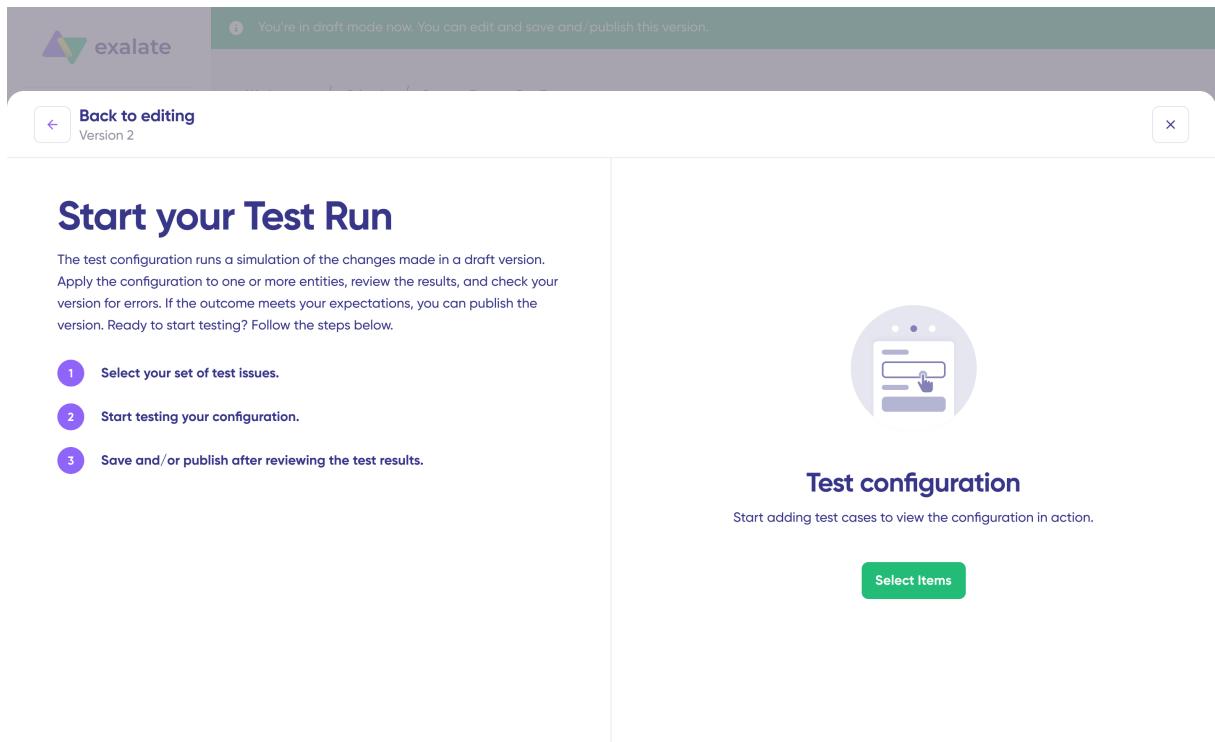
Start Test Run Save script X

Outgoing script	Incoming script
<pre>1 replica.type = issue.type 2 replica.summary = issue.summary 3 replica.description = issue.description 4 replica.priority = issue.priority 5 replica.attachments = issue.attachments 6 replica.status = issue.status 7 8 //Comment these lines out if you are interested in sending the full list of versions and 9 components of the source project. 10 replica.project.versions = [] 11 replica.project.components = [] 12 13 /** 14 Custom Fields 15 16 replica.customFields."CF Name" = issue.customFields."CF Name" 17 */ 18 19 20 21 22 23 24</pre>	<pre>1 if(firstSync){ 2 issue.projectKey = nodeHelper.getProjectByKey("DT") 3 // Set type name from source issue, if not found set a default 4 issue.typeName = nodeHelper.getIssueTypeByName(replica.type?.name) 5 ?:nodeHelper.getIssueTypeByName("Task") 6 issue.summary = replica.summary 7 issue.description = replica.description 8 issue.comments = commentHelper.mergeComments(issue, replica) { 9 it.executor = nodeHelper.getUserByEmail(it.author?.email) 10 } 11 issue.attachments = attachmentHelper.mergeAttachments(issue, replica) 12 issue.priority = replica.priority 13 // Define the mapping between Jira Cloud statuses and 14 GitHub statuses 15 def statusMap = ["To Do": "open", 16 "Done": "closed"] 17 // Apply the status mapping 18 def remoteStatusName = replica.status.name 19 issue.setStatus(statusMap[remoteForm.StatusName] ?: remoteStatusName) 20 21 /** 22 User Synchronization (Assignee/Reporter) 23 24 Set a Reporter/Assignee from the source side, if the user can't be found set a default</pre>

Describe what to send from this system. e.g., Exclude attachments... ↗

Describe how to apply incoming data to this system. e.g., Map statuses... ↗

5. Click **Select Issues**



The test configuration runs a simulation of the changes made in a draft version. Apply the configuration to one or more entities, review the results, and check your version for errors. If the outcome meets your expectations, you can publish the version. Ready to start testing? Follow the steps below.

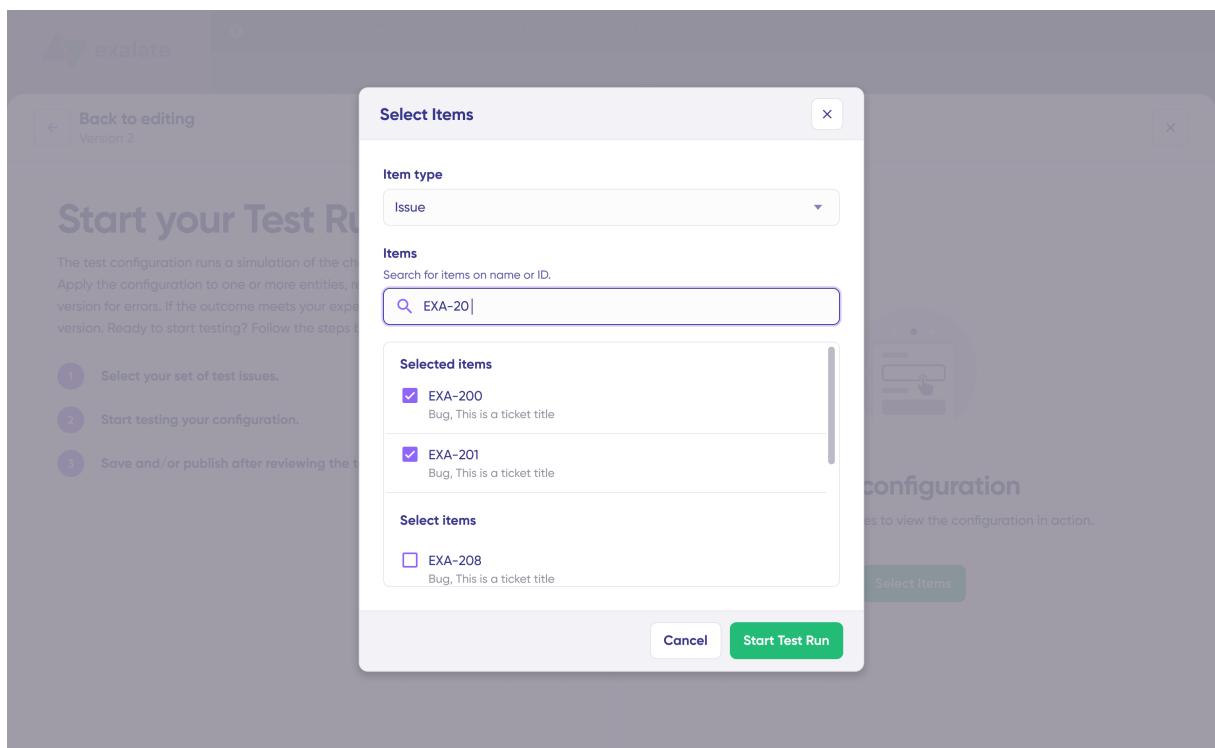
- 1 Select your set of test issues.
- 2 Start testing your configuration.
- 3 Save and/or publish after reviewing the test results.

Test configuration

Start adding test cases to view the configuration in action.

Select Items

6. Search for entities by an entity key and choose one or more entities to apply the test configuration to.



Start your Test Run

The test configuration runs a simulation of the changes made in a draft version. Apply the configuration to one or more entities, review the results, and check your version for errors. If the outcome meets your expectations, you can publish the version. Ready to start testing? Follow the steps below.

- 1 Select your set of test issues.
- 2 Start testing your configuration.
- 3 Save and/or publish after reviewing the test results.

Select Items

Item type: Issue

Items: EXA-20

Selected items:

- EXA-200
Bug. This is a ticket title
- EXA-201
Bug. This is a ticket title

Select items:

- EXA-208
Bug. This is a ticket title

Cancel Start Test Run

7. Click **Start TestRun** again to see the **outgoing and incoming replicas** for each selected issue.

Replica sent from "I" on "Azure R500" side

1

hubIssue:
 areaPath: rmb500
 iterationPath: rmb500
 voters:
 project:
 idStr: b86835be-7fa-4c07-9d1a-ac205b008b97
 key: rmb500
 name: rmb500
 fixVersions:
 internalMap:
 areaPath: rmb500
 iterationPath: rmb500
 priority:
 id: 3
 name: 3
 labels:
 customKeys:

0:

hubIssue:
 voters:
 fixVersions:
 projectKey: KAN
 internalMap:
 labels:
 customKeys:
 issueType:
 id: 10001
 name: Task
 entityProperties:
 components:
 attachments:
 customFields:
 Vulnerability:
 id: 10033

Feedback

Refine Your Configuration

- If adjustments are needed, click **Back to Editing**, modify the scripts, and then select **Re-run TestRun** to test the new version on the same set of issues.

Outgoing script
From Azure R500

```

1 replica.key = workItem.key
2 replica.assignee = workItem.assignee
3 replica.summary = workItem.summary
4 replica.description = nodeHelper.stringifyHTML(workItem.description)
5 replica.type = workItem.type
6 replica.status = workItem.status
7 replica.labels = workItem.labels
8 replica.priority = workItem.priority
9 replica.comments = nodeHelper.stringifyFromComments(workItem.comments)
10 replica.attachments = workItem.attachments
11 replica.project = workItem.project
12 replica.areaPath = workItem.areaPath
13 replica.iterationPath = workItem.iterationPath
14 /*
15 */
16 // Custom Fields (CF)
17 // How to send any field value from the source side to the destination side.
18 // 1/ Add the value to the replica object. Use the Field Name from the field or the API name:
19 // https://docs.exalate.com/docs/how-to-sync-work-item-fields-obtained-through-a-rest-api-call
20 // 2/ Uncomment this next statement out and change accordingly:
21 // replica.customFields."CF Name" = issue.customFields."CF Name"
22 /*
23 */
24 // Exalate API Reference Documentation: https://docs.exalate.com/docs/exalate-api-reference-documentation
25

```

Describe how to apply incoming data to this system. e.g., Map statuses...

Incoming script
Into Jira Cloud

```

1 if (firstSync) {
2   issue.projectKey = "KAN"
3 }
4 issue.type = nodeHelper.getIssueType(replica.type?.name, issue.projectKey) ?: nodeHelper.getIssueType("Task", issue.projectKey)
5 issue.summary = replica.summary
6 // Prepare areaPath to the status
7 if (replica.areaPath) {
8   issue.description = "!" + replica.areaPath + "\n" + (replica.description ?: "")
9 } else {
10   issue.description = replica.description
11 }
12 issue.comments = commentHelper.mergeComments(issue, replica)
13 issue.attachments = attachmentHelper.mergeAttachments(issue, replica)
14 issue.labels = replica.labels
15 /*
16 */
17 // Custom Fields (CF)
18 // To add incoming values to a Jira custom field, follow these steps:
19 // 1/ Find the Display Name of the CF. Note: If you have multiple custom fields with the same name, then you can sync it using the custom field ID instead of its name. Know more about the steps here:
20 // https://docs.exalate.com/docs/how-to-synchronize-custom-fields-in-jira-cloud
21 // 2/ Check how the value is coming over from the source side, by checking the "Entity Sync Status" of an issue in sync and then selecting the "Show Remote Replica".
22 // 3/ Add it all together like this:
23 issue.customFields."CF Name".value = replica.customFields."CF Name".value
24 /*
25 */
26 /*
27 */
28 /*
29 */

```

Describe how to apply incoming data to this system. e.g., Map statuses...

Feedback

- To change the test issues, click the **Edit icon**, add or remove issues, and restart the TestRun.

Product

Finalize and Publish

[Release History](#)

- Once satisfied with the results, click **Save Scripts** to store the changes.
- [Glossary](#)
- [API Reference](#)
- Click **Publish** to apply the updated configuration to your live synchronization.

[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](#) 