

Jira Service Management

Last Modified on 07/29/2026 3:14 pm EDT

This guide walks the Jira/Atlassian administrator through choosing the service projects in scope, creating a dedicated integration identity, granting the project and feature access the connector needs, authorizing the standard Fivetran Jira connector (OAuth 2.0 or Basic/API-token), configuring project selection and schema, registering the deleted-record webhook when required, verifying the synced tables, and submitting the connection through the Info-Tech portal. Jira Service Management Cloud is read through the standard Fivetran Jira connector; no data is written back to Jira.

Important — how Jira access works (read this first)

Jira access is not granted table by table. It is controlled by product access, project roles, permission schemes, issue security, and feature-specific permissions. The Fivetran destination tables appear only when Fivetran can actually retrieve the related Jira data through the connected identity. So the work below is mostly about giving one dedicated identity the right *project and feature access* — the tables follow from that.

A Before you start

Confirm the following before creating the connection:

- You are connecting to **Jira Service Management Cloud** — not Jira Data Center, and not a Jira Software-only site.
- You know the Jira site URL, for example `yourcompany.atlassian.net`.
- You know which Jira Service Management service projects are approved for reporting.
- You have a Jira or Atlassian administrator who can grant project access.
- You have a Jira administrator with the **Administer Jira** global permission available if manual webhook setup is needed for deleted issue/project capture (Step 6).
- You have a **dedicated integration identity** for the connector. Do not use a personal employee account unless your organization explicitly approves it.

Recommended connection approach

Use one of the two standard Fivetran Jira authentication methods below. **Do not** use Service Account OAuth 2.0 in the normal customer setup path.

OAUTH 2.0 — RECOMMENDED DEFAULT

Best when a Jira administrator can authorize the connector interactively. In the Fivetran setup form, select **OAuth**, click **Authorize**, sign in with the approved Atlassian account, review the requested access, and allow access.

BASIC AUTHENTICATION WITH API TOKEN

Use when your security team approves API-token authentication. In the Fivetran setup form, select **Basic**. Use the Atlassian account email address as the user and the Atlassian API token as the

password.

B Jira / Fivetran tables required

These are the Fivetran destination table families the Customer Data Store expects to validate. The Jira administrator does not grant these tables directly — instead, the connected Jira identity must have enough Jira access for Fivetran to retrieve the source data.

Required core table families (always in scope)

ISSUE	Core ticket records — issue ID, issue key, project, issue type, status, priority, created/updated/resolution dates.
PROJECT	Project and service-project context used to scope the extract.
ISSUE_TYPE	Native Jira ticket type — the source for Incident / Service Request / Problem / Change / Unknown mapping.
STATUS , STATUS_CATEGORY	Native workflow status and Jira's status grouping.
PRIORITY	Native Jira priority used for severity mapping.
RESOLUTION	Jira workflow resolution value.
USER	Requester and agent references resolved from issues, comments, and worklogs.
COMMENT	Ticket comments — public and internal — used for detail views and First Contact Resolution.
WORKLOG	Logged work time against issues — the worklog / time-effort source.
FIELD , FIELD_OPTION , ISSUE_FIELD_HISTORY , ISSUE_MULTISELECT_HISTORY	Custom field metadata and change history — reopen/escalation and custom-field-driven category/department mapping are derived here.

Conditional Jira Service Management table families (only if the feature is used and access allows it)

REQUEST_TYPE	Best JSM-native source for customer-facing request category. Do not use request type as ticket type.
ORGANIZATION	Customer organization context. Use as department only if the client confirms organization equals department.
SLA	Required for JSM SLA reporting. If missing, confirm the connected identity is in the Service Desk Team / agent role for the selected service projects.

SERVICE	Useful when the client uses JSM services and service relationships.
APPROVAL	Required only if approval reporting is in scope.
ASSET_OBJECT	Required only if Jira Assets is in scope and the connected identity can access the relevant object schemas.
COMPONENT	Useful for component / application / service / assignment-context mapping only when the client confirms the meaning.
ATTACHMENT and file sync	Use only if attachment replication is approved by the client and Info-Tech.
JSM feedback fields	Used for CSAT score/date when native JSM satisfaction feedback is enabled. Standard Fivetran may not provide a complete CSAT comment audit trail.

Write-access note: The Customer Data Store does not write Jira issues, comments, worklogs, projects, fields, or configuration back into Jira. Fivetran uses Jira webhooks only to capture *deleted* objects such as issues and projects. If Fivetran cannot register that webhook with the connected identity, a Jira administrator must create it manually using Step 6.

Steps in this guide

- Step 1: Choose Jira Service Management project scope
- Step 2: Create or choose the integration identity
- Step 3: Grant project and feature access
- Step 4: Authorize the standard Fivetran Jira connector
- Step 5: Configure Fivetran project selection and schema
- Step 6: Register the deleted-record webhook (when required)
- Step 7: Verify the integration identity can read Jira
- Step 8: Submit the connection through the Info-Tech portal

Step 1: Choose Jira Service Management project scope

Jira Administration or the Jira project list

Decide which Jira projects belong in ITSM reporting *before* creating the connection. Getting scope right here prevents non-ITSM projects from polluting the extract.

1. **Identify the Jira Service Management service projects** that represent ITSM work. Exclude Jira Software projects, backlog projects, business projects, test projects, and archived projects unless the client explicitly approves them.
2. **Keep a written list of approved project keys** — for example `ITSM` , `HELP` , or `SD` .

3. **Decide on agent-created issues.** Confirm whether agent-created issues inside the selected service projects should be included even when the issue has no JSM request type. The default recommendation is to include them and map request type to `Unknown` .
4. **Do not select all Jira projects by default.** Fivetran can sync all projects or selected projects — use **selected projects** and choose only the approved service projects.

Step 2: Create or choose the integration identity

Atlassian account administration + the Atlassian API-token page (for Option B)

Use a dedicated identity that is only used for this connection. Pick one of the two authentication options below to match how you will authorize Fivetran in Step 4.

OPTION A — OAUTH 2.0 WITH A DEDICATED ATlassian ACCOUNT

1. Create or choose a dedicated Atlassian account for the connector.
2. Grant it Jira Service Management product access as an **agent** if SLA, request, organization, customer, and comment data are required.
3. Grant it access to every in-scope service project.
4. Use this account when authorizing the Fivetran Jira connector in Step 4.

OPTION B — BASIC AUTHENTICATION WITH API TOKEN

1. Log in to the dedicated Atlassian account and open the Atlassian account API-token page.
2. Select **Create API token**; name it clearly, e.g. `CIO Analytics Fivetran Jira` .
3. Set an expiration date that follows your security policy, then select **Create**.
4. Copy the token and save it in your approved password manager — Atlassian does not let you recover it later.
5. In Fivetran, use the account email as the user name and the API token as the password.

NEVER SEND SECRETS BY EMAIL, CHAT, TICKET COMMENTS, SCREENSHOTS, OR SCREEN SHARE.
Enter OAuth authorization details or API tokens only into the secure connection flow.

Step 3: Grant project and feature access

Each in-scope service project's team / access settings

The connected identity must be able to see the same Jira records the Customer Data Store needs to report on. Access differs between company-managed and team-managed projects.

1. **Company-managed projects:** open each in-scope service project, go to **Invite team**, enter the integration

identity's email and select **Invite**. Confirm it is added to the **Service Desk Team / agent** role. If the project uses a custom permission scheme, grant access through the standard project roles (especially Service Desk Team) rather than attaching one-off permissions to the account.

2. **Team-managed projects:** open the service project, go to **Project settings / Service space settings**, open the **Access** or project role/members area, and add the integration identity with an agent-equivalent role for that project.
3. **Confirm visibility.** The identity must be able to browse each selected project and view in-scope issues, comments, worklogs, attachments, request participants, customers, organizations, and SLA information where those areas are in scope.
4. **Review issue security schemes.** If an issue security level hides records from the integration identity, Fivetran will not retrieve those hidden records.
5. **Jira Assets (if in scope):** confirm the identity can access the relevant object schemas. **SLA:** if SLA data is required, the identity must be in the Service Desk Team / agent role for the selected service projects.

Step 4: Authorize the standard Fivetran Jira connector

The secure Fivetran connection flow launched from the Info-Tech portal or the approved Fivetran setup page

Authorize Fivetran using the method chosen in Step 2. Use whichever path matches your authentication option.

OAuth 2.0 PATH

1. Set **Authentication Type** to **OAuth**.
2. Click **Authorize**.
3. Log in with the approved Atlassian account that has access to the Jira Cloud site.
4. Review the requested access, then click **Allow access**. You are redirected back to Fivetran.

BASIC / API-TOKEN PATH

1. Set **Authentication Type** to **Basic**.
2. In the **User** field, enter the email address of the Atlassian account used to create the API token.
3. In the **Password** field, enter the Atlassian API token.

Step 5: Configure Fivetran project selection and schema

Jira Cloud settings in the Fivetran setup form

Point Fivetran at the right destination schema and restrict the sync to the approved projects.

1. **Enter the destination schema name** provided by Info-Tech or your onboarding contact.
2. **Enter the Jira host** without `http://`, `https://`, or `www` – for example `yourcompany.atlassian.net`.
3. **Enter port** `443`.

4. **Choose issue sync mode From selected projects** and select only the Jira Service Management service projects approved in Step 1.
5. **Leave attachment/file sync off** unless Info-Tech and the client explicitly approve attachment replication.
6. **Click Save & Test.**

Step 6: Register the deleted-record webhook (when required)

Jira Settings → System → Advanced → WebHooks (requires the Administer Jira global permission)

Fivetran fetches Jira data through the REST API and uses a webhook to capture deleted objects such as issues and projects. Manual webhook setup is needed only when the connecting Jira user does **not** have the Administer Jira global permission and deleted-entity capture is required. If that applies, complete these steps exactly.

1. In the Fivetran Jira connection setup form, set **Customer Created Webhook** to **ON** and copy the **Webhook URL** it displays.
2. Log in to Jira as a user with the **Administer Jira** global permission.
3. Go to **Jira Settings** (gear) → **System**, then under **Advanced** select **WebHooks**.
4. Click **+ Create a WebHook**. Name it, e.g. `_fivetran_custom_webhook_`, and set **Status** to **Enabled**.
5. In the **URL** field, paste the Webhook URL copied from Fivetran.
6. In **Issue related events**, select the **deleted** event in the Issue column.
7. In **Project related events**, select the **delete** event in the Project column.
8. In **Jira Software related events**, select the **created**, **deleted**, and **updated** events in the Sprint column.
9. Click **Create**.

Why this exists: admin-level access is needed to register the webhook that captures deleted Jira issues and related entities. Creating this webhook does **not** give the Customer Data Store permission to create, edit, or delete Jira business records — it only lets Jira notify Fivetran when source records are deleted.

Step 7: Verify the integration identity can read Jira

Run a test API call from your machine, Postman, or another approved API client

Before submitting the connection, confirm the integration identity can authenticate to Jira and read the data the connector will pull. These are Jira Cloud REST API calls run against the credential from Step 2 — the same access Fivetran will use. The examples use Basic authentication (Atlassian account email + API token); if you set up OAuth, the equivalent check is that the authorization in Step 4 completed and Fivetran's **Save & Test** in Step 5 succeeded. Replace `YOUR_EMAIL`, `YOUR_API_TOKEN`, and `yourcompany.atlassian.net` with your values.

1. **Confirm authentication works (start here).** This returns the identity Jira sees for the credential:

```
curl --user "YOUR_EMAIL:YOUR_API_TOKEN" \
-H "Accept: application/json" \
"https://yourcompany.atlassian.net/rest/api/3/myself"
```

A successful response returns the integration account's `accountId` and `displayName`. `401 Unauthorized` means the email or API token is wrong; `403 Forbidden` means the account exists but is blocked from the API.

2. **Confirm the approved projects are visible.** Replace `ITSM` with an approved project key from Step 1:

```
curl --user "YOUR_EMAIL:YOUR_API_TOKEN" \
-H "Accept: application/json" \
"https://yourcompany.atlassian.net/rest/api/3/project/ITSM"
```

A successful response returns the project's details. `404 Not Found` here usually means the identity has not been granted access to that project — return to Step 3.

3. **Confirm issues in scope are readable.** This searches issues in an approved project:

```
curl --user "YOUR_EMAIL:YOUR_API_TOKEN" \
-H "Accept: application/json" \
"https://yourcompany.atlassian.net/rest/api/3/search/jql?jql=project%20%3D%20ITSM&maxResults=1"
```

A successful response returns a `total` count and at least one issue. If `total` is 0 but the project has tickets, the identity may be blocked by an issue security scheme — re-check Step 3.

4. **Confirm JSM data is readable (if in scope).** This checks the Service Management API the connector uses for requests and SLA:

```
curl --user "YOUR_EMAIL:YOUR_API_TOKEN" \
-H "Accept: application/json" \
"https://yourcompany.atlassian.net/rest/servicedeskapi/servicedesk"
```

A successful response lists the service desks the identity can see. `403 Forbidden` or an empty list usually means the identity is not in the **Service Desk Team / agent** role — this is the most common reason SLA and request data fail to sync, so fix it in Step 3 before submitting.

5. **When these calls succeed**, the identity has the access Fivetran needs. The Info-Tech team validates the synced tables and field mapping on their side after the first sync — you do not need to query the destination warehouse yourself.

Step 8: Submit the connection through the Info-Tech portal

Info-Tech portal link provided by your onboarding contact

Submit the connection details through the secure portal. Do not share OAuth credentials or API tokens outside the secure connection flow.

OAuth credentials and API tokens are sensitive. Do not send them to Info-Tech by email, chat, ticket

attachment, or screen share — use only the secure connection flow.

Head over to <https://us.app.cioanalytics.ai/>

1. Open the Info-Tech portal link from your onboarding contact and go to the Jira Service Management setup page.
2. Provide the destination schema name and confirm the approved project scope.
3. Complete the secure Fivetran authorization (OAuth) or enter the API-token connection detail in the secure flow — never by email, chat, or screen share.
4. Submit and wait for the validation result. If an error appears, recheck the matching step above (most connection failures trace back to project access in Step 3 or scope in Step 5).

After the connection validates, CIO Analytics takes over the rest of the process automatically.

1. Your Jira Service Management data syncs automatically for the first time. Depending on how much history is being pulled, the first sync can take anywhere from a few hours to a few days.
2. Once that first sync completes, an Info-Tech analyst will reach out to you to continue your onboarding. No further action is needed from you in the meantime.

NOTES ON THIS DOCUMENT

- Jira navigation varies between company-managed and team-managed projects, and by Atlassian release. If a menu item is not in the exact location shown, use the closest matching project settings, access, or system administration page.
- Jira access is granted through product access, project roles, permission schemes, and issue security — not table by table. If a Fivetran table family is missing after the first sync, the cause is almost always project or role access on the integration identity rather than the connector configuration.
- The Customer Data Store is read-only. Nothing is written back to Jira; the webhook in Step 6 exists solely so Jira can notify Fivetran when source records are deleted.
- Questions about business purpose, data scope, or onboarding sequencing can be directed to your Info-Tech onboarding contact.