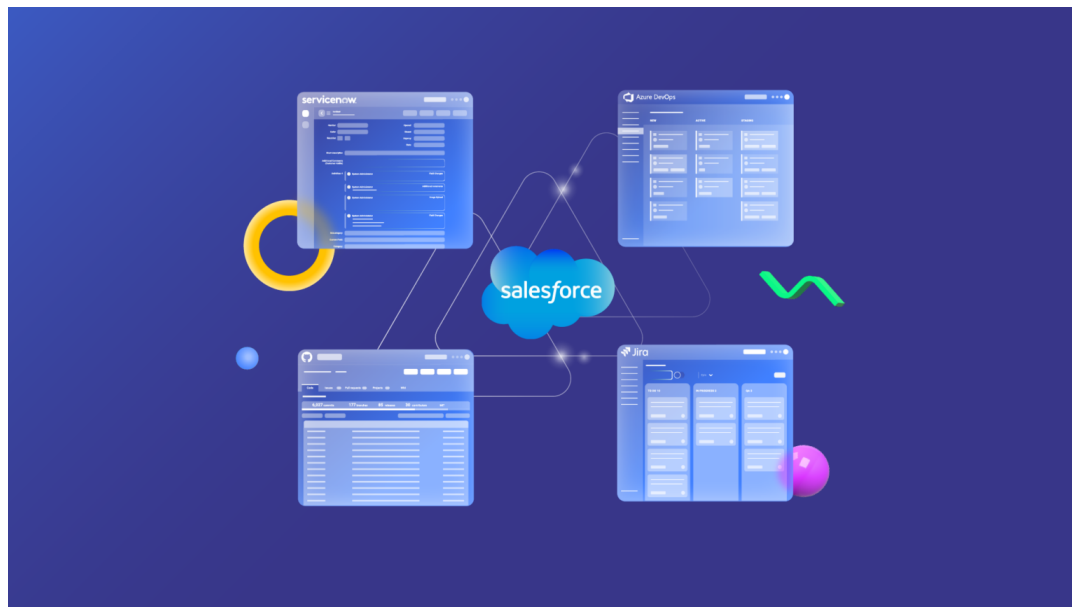




# Salesforce Third-Party Integration: Set up a Bidirectional Sync between Salesforce and Other Tools



## Table of contents

### Why Integrate Salesforce Using a Third-Party Integration Solution

- Common Use Cases

### Choosing the Right Solution for a Salesforce Third-Party Integration

- Decentralized Integration
- Flexibility
- Reliability

### How to Integrate Salesforce with other Tools through a Third-Party Solution

- Step 1: Install Exalate on Salesforce
- Step 2: Install Exalate on Jira
- Step 3: Connect Salesforce and Jira
- Step 4: Configure the Connection Rules
- Step 5: Set Up Triggers to Automate Data Exchange
- Step 6: Start Synchronizing Salesforce and Jira

### Conclusion



Salesforce is perfect for teams working with customer data. Feature-rich, and easy to use, it lets you control every aspect of the sales process, without advanced technical know-how.

There are ways to get more from it, however. The data you store can be shared with teams using other tools like Jira, Zendesk, etc. That means your sales department can work with other departments more easily, with data filtered and shared automatically.

What's covered in this article?

- [Why integrate Salesforce using a third-party integration solution](#)
- [Choosing the right solution for a Salesforce third-party integration](#)
- [How to integrate Salesforce with other tools through a third-party solution](#)

## Why Integrate Salesforce Using a Third-Party Integration Solution

Though Salesforce includes a broad range of functionality, a third-party integration can take you beyond its default capabilities.

**Note:** *If you want to learn more about the native Salesforce functionality, you can read this [Salesforce to Salesforce integration](#) guide.*

Specialized applications bring features that can help you connect to other platforms, with little development effort. Here are some examples of how that could work.

### Common Use Cases

Salesforce is typically used by sales teams, but you can use a third-party integration to connect to other platforms.

If you have sales teams in different locations, you can integrate their data, and exchange tickets. Perhaps they sometimes acquire customers that are best handled by other teams. An integration can help your teams share their details automatically.

Developers using Jira can share progress on new items with the sales team via an integration. The sales team can make developers aware of customer requests for features, and developer feedback can be automatically sent back to the original ticket.

Project managers working in Azure DevOps can share targets with the sales team, and the sales team can have that data automatically synchronized with the other team's system.

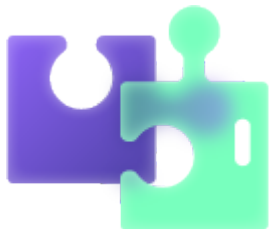
There are many other ways a third-party integration can contribute to your business. All it takes is a little imagination.

## Choosing the Right Solution for a Salesforce Third-Party Integration

The integration solution we have chosen for this article is called [Exalate](#). Exalate is an integration solution that enables two-way syncs between multiple tools like Salesforce, Jira, Zendesk, ServiceNow, HP ALM, GitHub, Azure DevOps, and more.

There are a few criteria that you need to take into consideration when choosing an integration solution for your specific use case.

### Decentralized Integration



With a decentralized integration, neither side owns the connection. If different teams manage each platform, they retain their autonomy and have full control over what they share, and how incoming data is mapped to their systems. Exalate gives each team an independent configuration screen, allowing them to choose what is shared.

## Flexibility

Exalate is hugely customizable and suitable for many scenarios. Its scripting engine allows you to filter and map data using advanced logic. Exalate also has a drag-and-drop mode for simpler use cases.

## Reliability



Exalate is designed with reliability in mind. If one side of the connection drops, it can recover gracefully. There's no need for manual intervention. Once it is running, you can forget about it.

## How to Integrate Salesforce with other Tools through a Third-Party Solution

An advantage of using Exalate as a Salesforce third-party integration solution is that it uses a common interface on each platform.

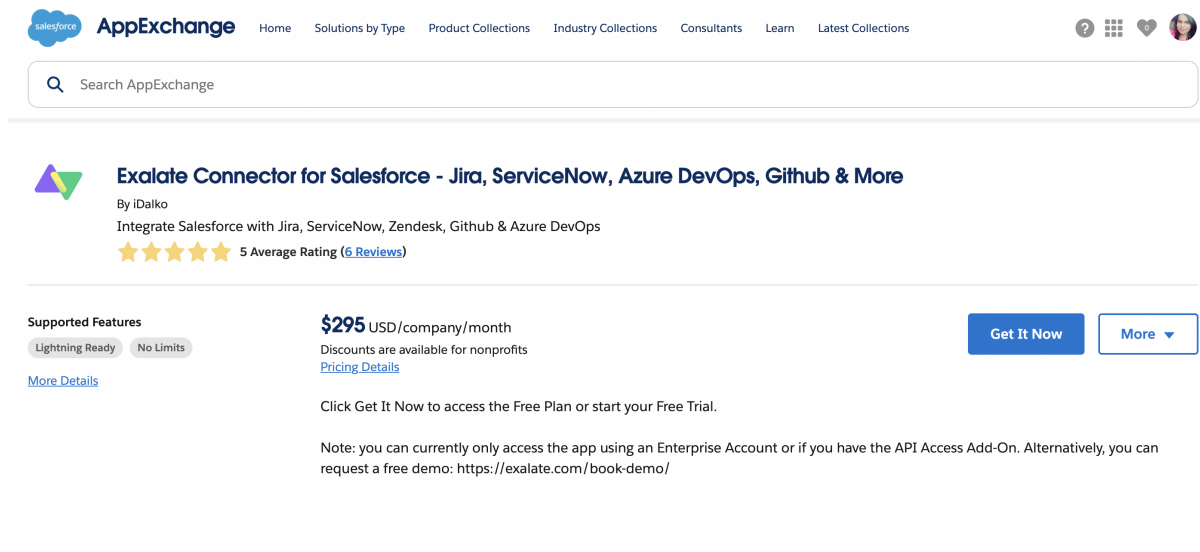
In this example, we'll connect it to Jira Cloud, but the process is the same for other platforms. You can integrate any of these with Salesforce, or each other, just as easily.

The process starts with installing Exalate on both platforms, then connecting them. Next are some optional configuration steps for controlling how data is shared, and when data exchange happens.

Let's go through each step.

## Step 1: Install Exalate on Salesforce

You can do the first two steps in either order, but let's install Exalate on Salesforce first. Open the AppExchange marketplace. Search for "Exalate", then click "Get It Now".



The screenshot shows the Salesforce AppExchange marketplace page for the Exalate Connector. The page header includes the Salesforce logo, the AppExchange logo, and navigation links for Home, Solutions by Type, Product Collections, Industry Collections, Consultants, Learn, and Latest Collections. A search bar is present with the text "Search AppExchange".

The main content area features the Exalate logo and the title "Exalate Connector for Salesforce - Jira, ServiceNow, Azure DevOps, Github & More". Below the title, it says "By iDalko" and "Integrate Salesforce with Jira, ServiceNow, Zendesk, Github & Azure DevOps". There is a 5-star rating with 6 reviews.

Under "Supported Features", there are two tags: "Lightning Ready" and "No Limits". The pricing is listed as "\$295 USD/company/month" with a note that discounts are available for nonprofits and a link to "Pricing Details". There are two buttons: "Get It Now" and "More".

Below the pricing, there is a note: "Click Get It Now to access the Free Plan or start your Free Trial." and another note: "Note: you can currently only access the app using an Enterprise Account or if you have the API Access Add-On. Alternatively, you can request a free demo: <https://exalate.com/book-demo/>".

Choose either production or sandbox mode depending on your needs, along with who has access to Exalate, and whether to grant access to third-party websites.

Once you've made your choices and "Installation Complete" appears, click "Done". Now you have to request a node. In Salesforce, you can find the newly installed app by typing "Exalate" in the search bar.

If you click "Request Node", you'll get an email with a verify button. Click that to set up your node and complete your installation.

## Step 2: Install Exalate on Jira

Next, switch to Jira and install Exalate there. We're using Jira Cloud, but you can read about how to set up Jira on-premise [here](#).

The screenshot shows the Jira Marketplace search results for 'Exalate'. The search bar contains 'exalate' and shows 10 results. Three results are visible:

- Exalate Jira Issue Sync & more**: Two-way Jira sync tool. Supports Jira to Jira Integration, ServiceNow, Github, Zendesk, Salesforce, Azure DevOps integration... IT & helpdesk, Integrations, Tasks, Workflow. 5 stars, 80 reviews, ADDED, CLOUD FORTIFIED.
- Exalate: ServiceNow to Jira Integration**: Experience a seamless, real-time and two-way ServiceNow Jira Integration. Admin tools, Continuous integration, IT & helpde... 5 stars, 3 reviews, 271 downloads, CLOUD SECURITY PARTICIPANT.
- Exalate: Zendesk to Jira Integration**: Experience a seamless, real-time and two-way Zendesk Jira Integration. Admin tools, Continuous integration, IT & helpde... CLOUD SECURITY PARTICIPANT.

First, search for “Exalate” in the Atlassian Marketplace. Click “Exalate Jira Issue Sync & more” and install it via the “Try it free” button, followed by the “Start free trial” button.

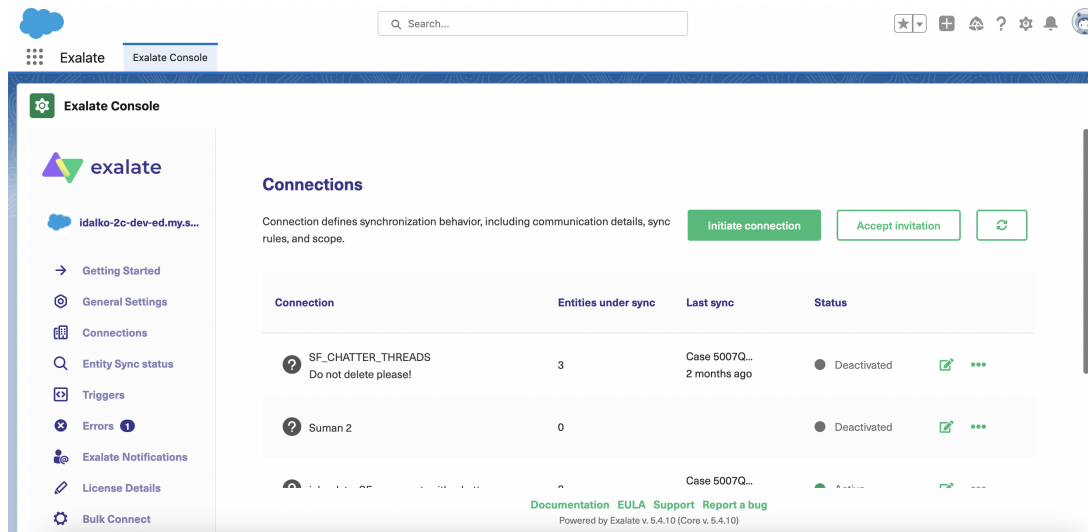
After a brief install process, click “Get Started”.

Now you’re ready to connect your instances.

### Step 3: Connect Salesforce and Jira

Now that Exalate is installed on both platforms, let’s create a connection between Salesforce and Jira.

You can start from either side of the connection. This guide uses Salesforce, so find the Exalate screen on your Salesforce installation, then click “Connections”. This screen shows any existing connections but will be empty the first time you use it.



The screenshot displays the Exalate Console interface. The top navigation bar includes the Exalate logo, a search bar, and utility icons. The main content area is titled "Exalate Console" and features a sidebar with navigation options: Getting Started, General Settings, Connections, Entity Sync status, Triggers, Errors (1), Exalate Notifications, License Details, and Bulk Connect. The "Connections" section is active, showing a table of existing connections. The table has columns for Connection, Entities under sync, Last sync, and Status. Two connections are listed: "SF\_CHATTER\_THREADS" (3 entities, last sync 2 months ago, deactivated) and "Suman 2" (0 entities, last sync Case 5007Q..., deactivated). Below the table are links for Documentation, EULA, Support, and Report a bug, along with version information: Powered by Exalate v. 5.4.10 (Core v. 5.4.10).

Connection	Entities under sync	Last sync	Status
SF_CHATTER_THREADS Do not delete please!	3	Case 5007Q... 2 months ago	Deactivated
Suman 2	0	Case 5007Q...	Deactivated



To create your first connection, click the “Initiate connection” button. Next, you’ll see a “Destination instance URL” box. Enter your Jira instance’s address, so that Exalate can verify it is available.

Next, you should select the Basic mode or the Script mode.


The Basic mode uses automatic configuration, with simple visual tools. The Scripting mode uses programming logic to set complex rules that control how your data is shared.

### Initiate connection ×


**Destination instance URL** ⓘ

 ✓ I don't have a URL

**Choose the configuration type**

 **Basic**

- Automatic configuration of basic fields
- Sync rules cannot be edited
- Only Cases can be synced
- Recommended for use cases of basic complexity

 **Script**

- Groovy-based scripting
- Configure each side of the connection separately
- Recommended for use cases of basic to advanced complexity

Next

Let's cover them in turn.

### Continue with the Basic Mode

The Basic mode takes you through a few simple dialogue screens. You can head straight to the other side of your connection by clicking a button.

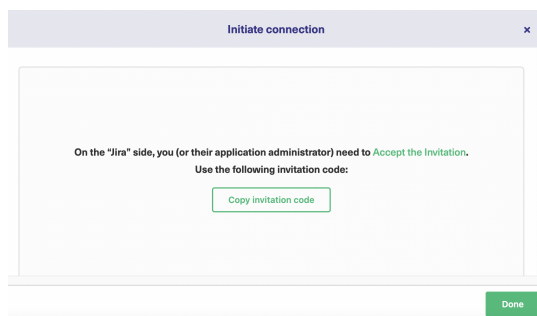
Once there, choose a project and click "Confirm". Your connection is now ready and you can test it by picking a test issue to synchronize.

If you don't have admin access to the other side of the connection, use the Script Mode guide to complete the process.

### Continue with the Script Mode

In the Script mode, you need to choose a name for your connection, and can optionally add a description, both of which can help you make sense of things later on, especially if you have multiple connections.

After adding these details, click "Initiate". You'll see a "Copy invitation code" button. Click that, and then click the "Done" button.



On the Jira connection screen, click the “Accept invitation” button and paste the code into the space provided. The code should be on your clipboard. If not, go back and copy it again.

×Accept invitation

**Please paste the invitation code, received from your partner**

**Enter the invitation code**

VEt0JEWtRiGtH2ZCISlmh0zEYVommm0100dRrWz0VtE0RnB0VZEM0yT2Vd0ZKtEXp20mK0WdJ0H0KZ0dEY200m0dZEXmB0F0  
ZS5jbG91ZCISlmIzc3VIVHJhY2tlclVybCI6Imh0dHBzOi8vaWRhbGtvLTJjLWRldi1lZC5teS5zYWxlc2ZvcmlmNvbSIsIm5v  
ZGVUeXBlljoiU0FMRVNGT1JDRSIsImIzc3VIVHJhY2tlclZlcnNpb24iOiI3LjAuMCIsIm5vZGVhcHBWZXJzaW9uljoiNy4wLjAi  
LCJub2RlVmVyc2lvbil6IjUuNC4xMCIsImV4YWxhdGVQbHVnaW5WZXJzaW9uljoiNS40LjEwliwibWluSHViT2JqZWNOc1ZI  
cnNpb24iOiIxLjluMCIsIm1heEh1Yk9iamVjdHNWZXJzaW9uljoiMS4xNC4wliwibWluUmVzdFZlcnNpb24iOiIxLjAuMCIsIm1h  
eFJlc3RWZXJzaW9uljoiNS4yLjAiLCJwb2xsT25seSI6ZmFsc2UsImV4YUNvbXBWZXJzaW9uljoiNS40LjEwliwicmF3RXhhQ  
29tcFZlcnNpb24iOiI1LjQuMTAiLCJpbnN0YW5jZVZVpZCI6IjM1NmMzZTJkLTcyZmMtNGRkMC1hZDU4LTRmNjA1NjU0OGI  
yOSIsImJhc2ljQ29ubmVjdGlvbil6eyJmaWVsZHMlOlt0dfX0slmNvbM5IY3Rpb24iOmsibmFtZSI6IiNhbGVzZm9yY2VfdG9fS  
mlyYSIsImRlc2NyaXB0aW9uljoiIiwiaWY29tbXVuaWNhdGlvbil6eyJzZW5kUHJvdG9jb2wiOiJESVJFQ1RfSFRUUCIsInJlY2Vpd  
mVQcm90b2NvbCI6IkrJUKVDVF9IVFRQIiwicmVtb3RlVXJsljoiHR0cHM6Ly9qY2xvdWRub2RlLWFrcmEtdWRpYy1ub3Vw  
LWFtbWkuZXhhbGF0ZS5jbG91ZCISlmIzc3VIVHJhY2tlclVybCI6Imh0dHBzOi8vaWRhbGtvGxheS5hdGxhc3NpYW4ubmV  
0In0slmxvY2FsSW5zdGFuY2VOYW1lljoiSmlyYSIsInJlbnB0ZUluZmFtZSI6IiNhbGVzZm9yY2UifX0=

Next

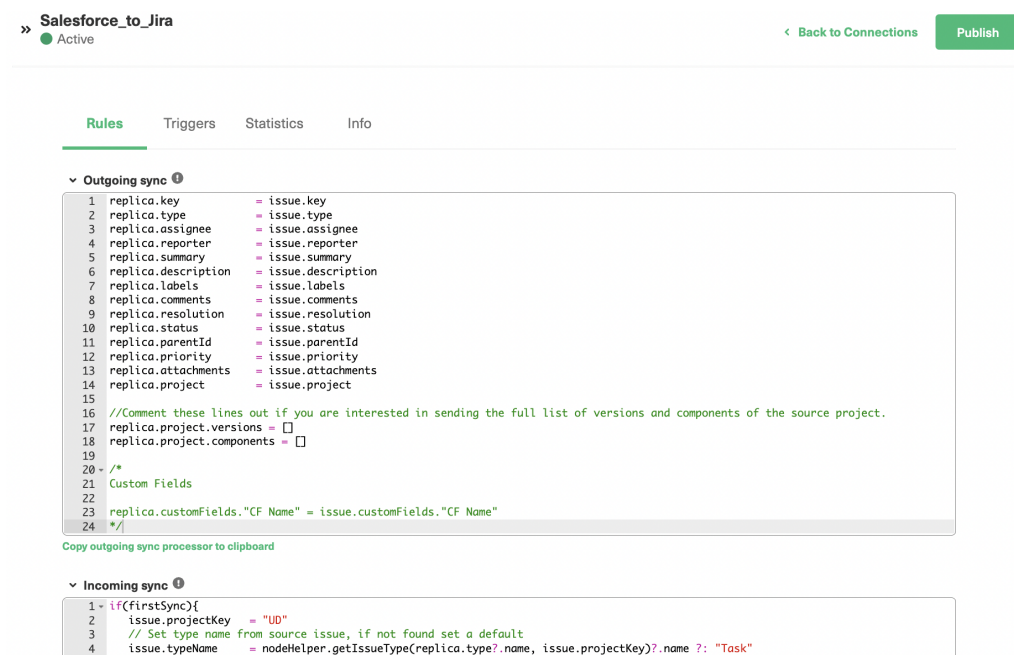
Click next, and Exalate will ask you a few questions, as in the Basic mode. When you're done, the connection is created and you can move on to the next steps.

## Step 4: Configure the Connection Rules

Step 4 lets you control what is shared, and how fields from each side of the connection are mapped onto each other.

To begin, find the newly created connection on the Exalate connections screen. Then click the "edit connection" button.

There are multiple tabs on the screen that appears, and "Rules" should be visible by default. Select it if not.



The screenshot shows the configuration interface for a connection named "Salesforce\_to\_Jira". At the top, there is a "Publish" button and a "Back to Connections" link. Below the connection name, there are tabs for "Rules", "Triggers", "Statistics", and "Info". The "Rules" tab is selected, showing two sections: "Outgoing sync" and "Incoming sync".

**Outgoing sync**

```
1 replica.key = issue.key
2 replica.type = issue.type
3 replica.assignee = issue.assignee
4 replica.reporter = issue.reporter
5 replica.summary = issue.summary
6 replica.description = issue.description
7 replica.labels = issue.labels
8 replica.comments = issue.comments
9 replica.resolution = issue.resolution
10 replica.status = issue.status
11 replica.parentId = issue.parentId
12 replica.priority = issue.priority
13 replica.attachments = issue.attachments
14 replica.project = issue.project
15
16 //Comment these lines out if you are interested in sending the full list of versions and components of the source project.
17 replica.project.versions = []
18 replica.project.components = []
19
20 /*
21 Custom Fields
22
23 replica.customFields."CF Name" = issue.customFields."CF Name"
24 */
```

Copy outgoing sync processor to clipboard

**Incoming sync**

```
1 = if(firstSync){
2   issue.projectKey = "UD"
3   // Set type name from source issue, if not found set a default
4   issue.typeName = nodeHelper.getIssueType(replica.type?.name, issue.projectKey)?.name ?: "Task"
```

Here, you can see two sets of rules, one for incoming items and one for outgoing. These describe how each item is mapped to its equivalent on the other platform.

When you sync tickets, the data is copied between these fields, from the existing ticket to the new one.

By default, the mappings use identical, or similar fields. In the above screenshot, you can see that *replica.description* maps to *entity.Description* in the outgoing rules.

You can change, remove or adjust any of these lines.

If you want to remove a line, delete it or comment it out. You could also add specific text. For example, you could change the description line to read: *replica.description* = "Synced from Salesforce". That way all items in Jira that come from Salesforce will have that description.

You could match items to others if that suits you better. Perhaps you want to make the summary on one platform list the status on the other. In that case, you could change the summary line to *replica.summary* = *entity.Status*

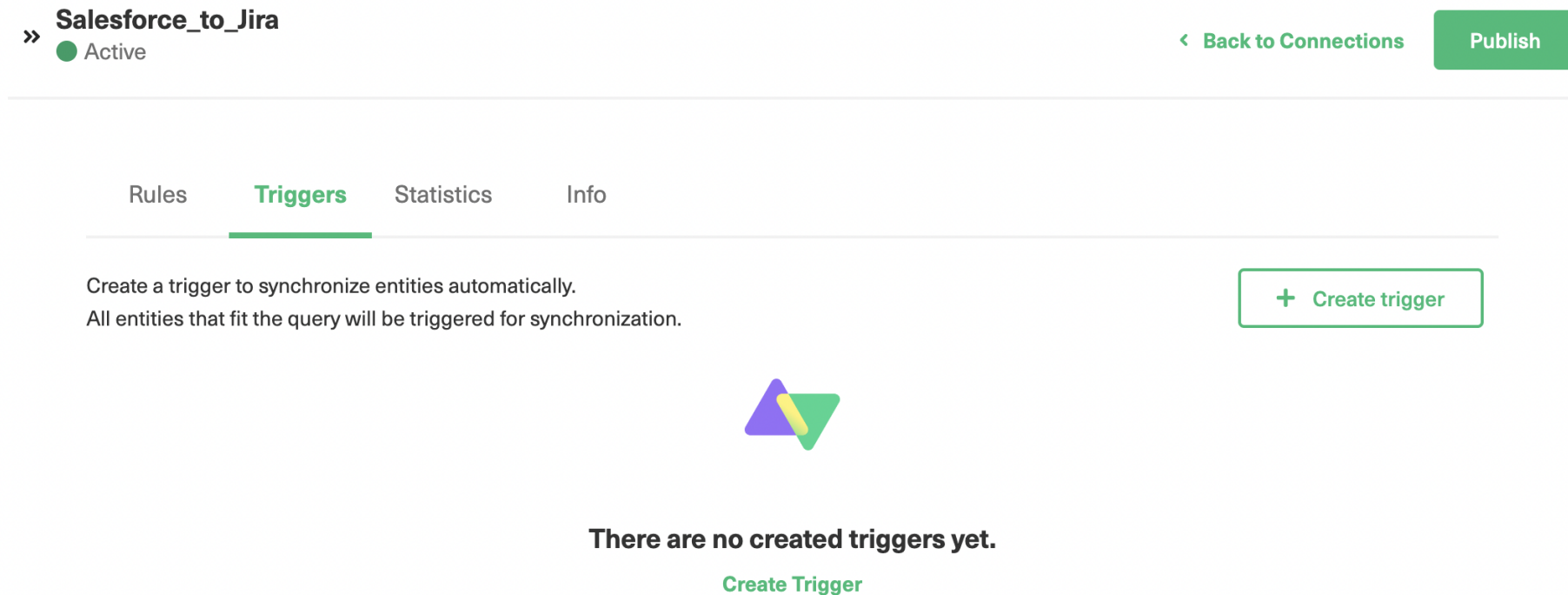
You can go further and use code to make decisions based on the values being exchanged. That's beyond the scope of this article, but if you have any coding skills you can probably think of a few ideas yourself.

## Step 5: Set Up Triggers to Automate Data Exchange

In the previous step, you used sync rules to decide what maps to what. In this step, you'll create a trigger that decides when items are exchanged. You can have as many triggers as you like, setting different conditions for sharing.

Perhaps you want to share high-priority items or items assigned to a particular person. You can create separate triggers for both of those things.

To get started, click the “Triggers” tab on the edit connection screen you accessed in the previous step. Then click the “Create trigger” button.



The screenshot shows the configuration page for a connection named "Salesforce\_to\_Jira". At the top left, there is a double chevron icon followed by the connection name and a green dot labeled "Active". At the top right, there is a green button labeled "Publish" and a link labeled "< Back to Connections". Below this is a horizontal navigation bar with four tabs: "Rules", "Triggers" (which is highlighted with a green underline), "Statistics", and "Info". Below the navigation bar, there is a text area that says "Create a trigger to synchronize entities automatically. All entities that fit the query will be triggered for synchronization." To the right of this text is a green button with a plus sign and the text "+ Create trigger". In the center of the page, there is a large, faint watermark logo consisting of two overlapping triangles, one purple and one green, with a yellow pencil tip pointing upwards. Below the watermark, the text reads "There are no created triggers yet." followed by a green link labeled "Create Trigger".

On the add trigger screen, there are several controls. Firstly, you can choose the type of entity the trigger synchronizes. There's a notes section to describe the trigger, and a switch, labeled “Active?”, to switch the trigger on and off.

There's also a "Use search query" switch that gives you a choice of how to do things. If you don't use a query, you can specify values for fields that will activate the trigger if met.

For more advanced decision-making, you can use a search query. That uses code for more advanced control. In Salesforce, you use SOQL (Salesforce Object Query Language), and JQL (Jira Query Language) in Jira.

As an example, you could use *Name = 'demo opportunity'* to select items with the specified name. You can use programming constructs like AND and OR to combine conditions.

For more on using SOQL with Exalate, [read this](#). For JQL triggers, [check this guide](#).

When you're done, click the "Add" button to create the trigger.

## Step 6: Start Synchronizing Salesforce and Jira

Now the platforms are connected, you can start adding more triggers that match the data you want to share. The integration will automatically share this data between Salesforce and Jira. It's a good idea to monitor what happens at first to make sure it matches your requirements.

Now that it's working, you can make it do even more for you by creating advanced filters and sharing rules. Experimenting with the triggers and sync rules can give you new ideas on how you can use them, so try a few things out.

## Conclusion

When you want to share data between platforms your choice of solution is key. Get it right and your teams will benefit.

Exalate, as a Salesforce third-party integration solution, is designed to deliver reliability and flexibility in a decentralized integration. It lets you integrate multiple platforms easily without leaving the tool(s) you are familiar with.

You can choose exactly how and when data is shared. Once set up, your integration does the hard work for you, leaving your teams free to get on with what they do best.

### ***Recommended Reads:***

- [Jira Salesforce Integration: How to Set up a Two-Way Sync between Different Teams](#)
- [Salesforce to Salesforce Integration: Sync Multiple Salesforce Instances Bidirectionally](#)
- [How to Set up an Azure DevOps Salesforce Integration](#)
- [GitHub Salesforce Integration: How to Set up a Sync in 6 Steps](#)
- [How to Set up a Salesforce ServiceNow Integration](#)
- [Salesforce Zendesk Integration](#)