



# Flow loops

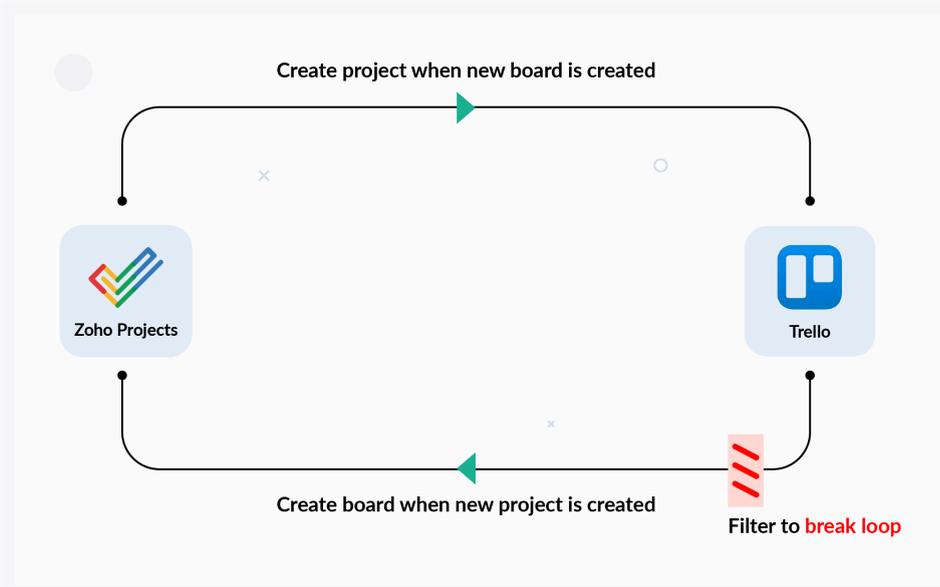
## ^ Table of contents

### ▸ How to prevent infinite flow looping?

#### ▸ Symptoms of an infinite flow loop

#### ▸ How to prevent infinite flow looping in different scenarios

- Scenario 1: Syncing data bidirectionally between two apps



- Scenario 2: When you sync data between multiple apps that might have links to the initial app

- Scenario 3: When you sync data between two apps and one app has a custom function or logic feeding data into the other.

#### ▸ Troubleshooting

## How to prevent infinite flow looping?

**Looping** is a scenario in which a set of connected tasks execute repeatedly. When such loops are not regulated, they execute endlessly, creating **infinite loops**. A flow loop happens when a flow gets invoked repeatedly without

any intended trigger events. A typical case is when two flows perform actions in such a way that they trigger each other. This can cause the flows to send the same data to and fro repeatedly, creating numerous duplicate entries in your connected apps. This also leads to exhaustion of your available tasks within a brief period.

## Symptoms of an infinite flow loop

Infinite flow loops may show some or all of these symptoms when the flow is switched on:

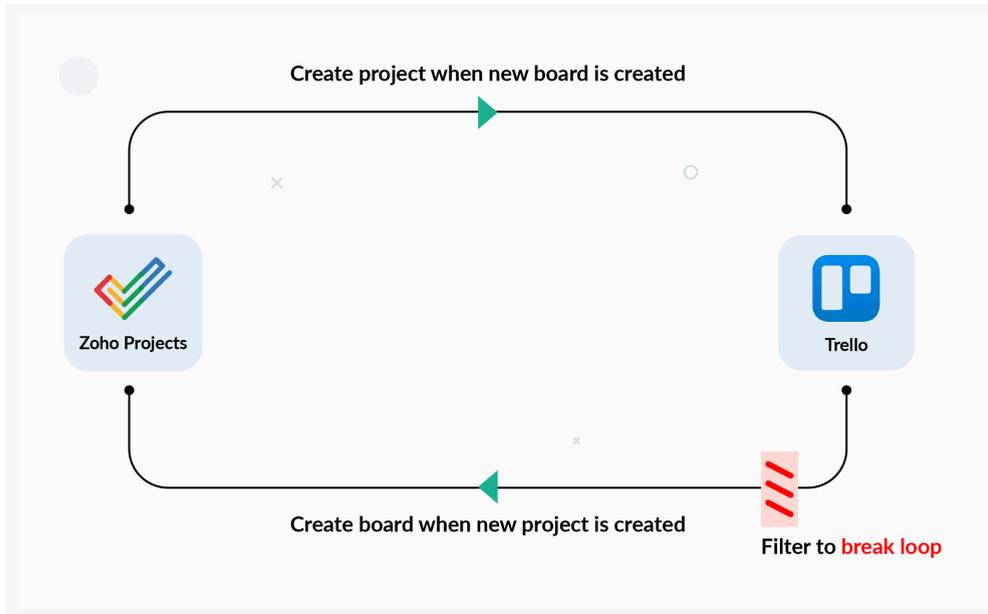
- Your task count is increasing rapidly and unexpectedly
- The flow is triggered when there are no actual changes in the source application
- Multiple duplicate entries are created in the connected apps without actual record creation
- Records are modified numerous times with the same redundant data without actual updates in the connected apps

## How to prevent infinite flow looping in different scenarios

Looping can occur in different ways.

### Scenario 1: Syncing data bidirectionally between two apps

Suppose you want to sync newly created data between Zoho Projects and Trello. Every project created in Zoho Projects must have a corresponding board in Trello, and vice versa. However, when a board is created in Trello as a result of the first flow's action, it triggers the reverse flow automatically, creating a new project in Zoho Projects. This continues because the flow is set up to trigger on any newly created entry. The solution is to understand how your apps are connected, and to implement logical conditions in your flows that prevent duplication of data.



## Solution:

**Step 1:** Ensure that the trigger in one of the flows contains a field that you have control over. For example, you could create a custom field, say *Status*, a field that you can fill in with a keyword of your choice.

**Step 2:** Click **Logic** in the flow builder and drag the **Decision** block onto the builder between your trigger and action. Set the decision to run only if the custom field from step 1 contains your keyword. For example, you could configure your flow to run only if your keyword is entered in the custom field created (in this example, *Status*). Whenever you create a project, remember to use your keyword (for example, "key1").

**Step 3:** Set up the second flow.

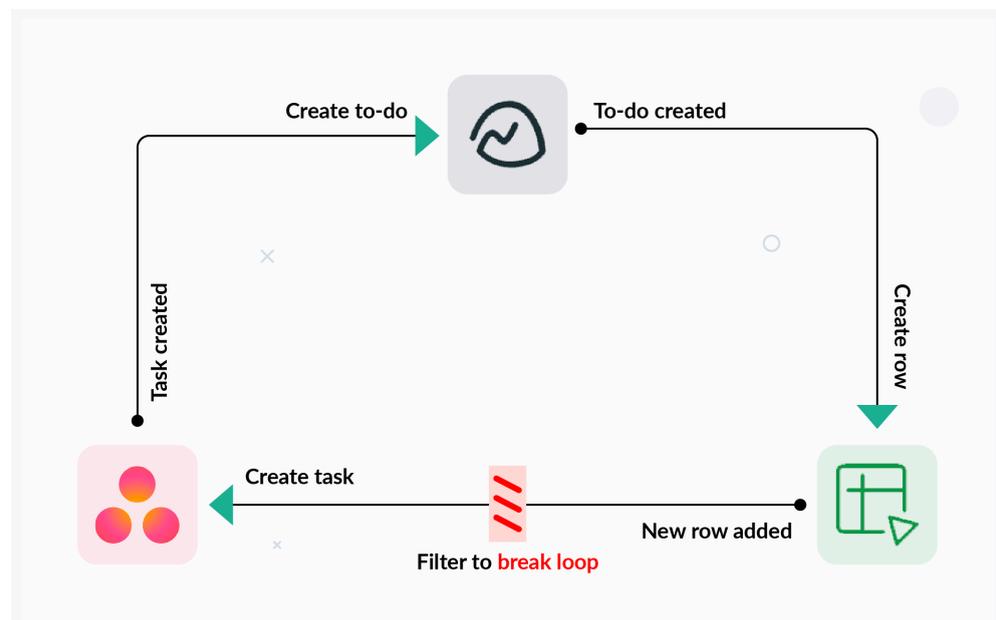
**Step 4:** Whenever you create a project, make sure you include "key1" in the custom field in Zoho Projects. This avoids the exchange of duplicate entries between the connected applications.

**Note:** This method works by using a user-controlled field that indicates to the flow that the record creation was performed by a user, and not an automated duplicate entry. Remember to use a newly created custom field or typically unused fields, like *Description* or *Notes*, for keywords to keep the auto-generated or important fields like *Title* or *Project ID* undisturbed.

If you are using a fetch action to sync newly created records between two apps, add a *Decision* block that allows the flow to continue only if the fetched record values return null. This ensures that the second app does not hold similar data. Configure the second flow in the same way.

## Scenario 2: When you sync data between multiple apps that might have links to the initial app

Suppose you have created a flow that takes in new tasks created in Asana and creates to-dos in Basecamp. There might be another flow that creates a new row in Zoho Sheet for every to-do created in Basecamp. Later on, a flow is created that syncs the task data between Zoho Sheet and Asana. This completes a loop, and there is no filter to stop the flows from executing repeatedly.



### Solution:

This happens when multiple flows connecting different apps form a chain that returns data to the initial app, resulting in a loop.

Identify all the flows that connect your apps. If there are flows that sync data, make sure that none of the actions in your flows trigger other flows. If such a connection is necessary, make sure to set *Decision* blocks to let the flows run only if the records do not match the original.

### Scenario 3: When you sync data between two apps and one app has a custom function or logic feeding data into the other.

Suppose you have an already-existing custom function that syncs data from Zoho CRM to Zoho Books. If you create a flow that syncs data from Zoho Books to Zoho CRM using Zoho Flow, it can complete the loop and generate duplicate entries.

### Solution:

Identify already existing connections between the apps. This includes direct integration using code, external data syncing applications and others.

## Troubleshooting

Wherever you come across the symptoms that are usually associated with infinite looping, try to resolve the issues yourself by carrying out the following steps:

### **TURN OFF the concerned flows immediately**

Immediately turn off all the flows that connect your concerned apps when you detect something is wrong with its records. Wait for a few minutes to see if the data duplication is still going on. If it's not, check if the scenario falls under the above-mentioned scenarios and try the solutions given.

### **Check your flow logs**

Go to Settings and navigate to History to view your Task History. This gives you the status of how your flows are running, and lets you identify problems as soon as they occur. Click on a task to view the detailed log of that particular execution.

### **Check for other integrations or codes**

If turning off the concerned flow does not stop the behavior, look for other integrations using code or other third party applications that connect your concerned apps.

### **Contact support**

If everything else fails to solve the issue, you can write to our support team at [support@zohoflow.com](mailto:support@zohoflow.com)