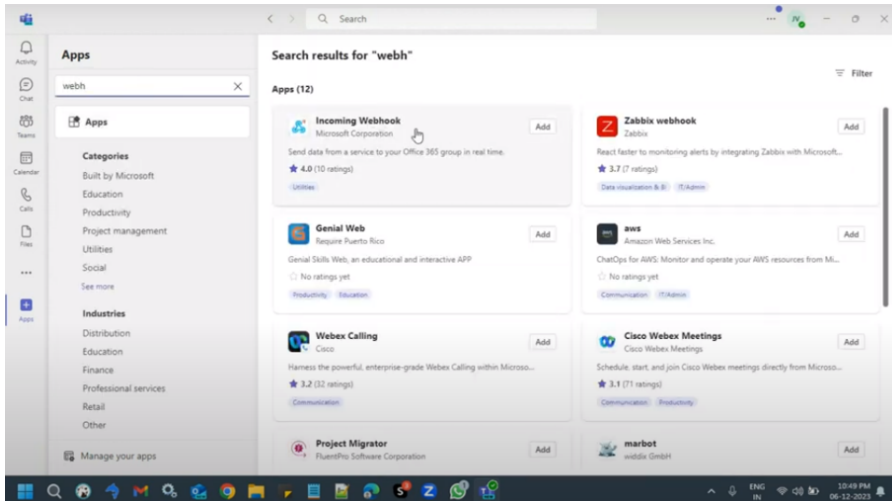


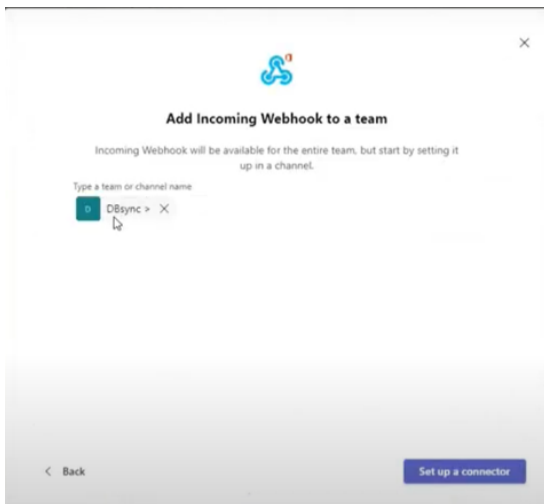
Webhooks Setup for Microsoft Teams

For activating notifications through Weebhooks on the team follow the following steps

- Login to your teams
 - Go to the apps section and search for Weebhooks



- Select Incoming Webhooks
- Select the team or channel where you want to get the notifications



- To set an incoming webhook Provide a name and click create

Connectors for "" channel in "DBsync" team


Fields marked with * are mandatory

To set up an Incoming Webhook, provide a name and select Create. *

DBsync replication 8.2

Customize the image to associate with the data from this Incoming Webhook.

Upload Image



Default Image

Create Cancel


Note: If you're a software developer and want to learn more about sending data to Office 365 using Incoming Webhooks, see Get started with Office 365 Connectors.

- Copy the URL to save it on the clipboard as this URL will be required to be pasted in the Notification section on the DBSync Notification page

Connectors for "" channel in "DBsync" team

Customize the image to associate with the data from this Incoming Webhook.

Upload Image



Copy the URL below to save it to the clipboard, then select Save. You'll need this URL when you go to the service that you want to send data to your group.

<https://avankia.webhook.office.com/webhook-1>

Url is up-to-date.

Done Remove

Note: If you're a software developer and want to learn more about sending data to Office 365 using Incoming Webhook, see Get started with Office 365 Connector Cards.

Once Teams setup is completed go to the DBSync Notification Page

- Select **'Webhooks'** as the channel type and the settings will appear on the screen as seen in the below image

Channel Type: Webhooks

Receive Notification When

Process	<input type="checkbox"/> Started	<input type="checkbox"/> Completed	<input type="checkbox"/> Errored
Object	<input type="checkbox"/> Added	<input type="checkbox"/> Updated	<input type="checkbox"/> Removed
Field	<input type="checkbox"/> Added	<input type="checkbox"/> Updated	<input type="checkbox"/> Removed

Url: GET Enter Url Here..

Test Webhook Save

- Configure Webhooks Notifications Settings**

1. Receive Notifications

Click the checkbox next to the desired event options. There are 3 main event options that can be configured are shown in the image below:

Receive Notification When	Process	<input type="checkbox"/> Started	<input type="checkbox"/> Completed	<input type="checkbox"/> Errored
	Object	<input type="checkbox"/> Added	<input type="checkbox"/> Updated	<input type="checkbox"/> Removed
	Field	<input type="checkbox"/> Added	<input type="checkbox"/> Updated	<input type="checkbox"/> Removed

- **Process**

- Started**- Click the checkbox next to the icon 'Started' if you wish to be notified when a process is started. For eg: If you enable this option you will be notified when a process such as 'Clean Copy', 'DB to Source', etc starts running.
- Completed**- Click the checkbox next to the icon 'Completed' if you wish to be notified when a process is completed. For eg: If you enable this option you will be notified when a process such as 'Clean Copy', 'DB to Source', etc completes.
- Errored**- Click the checkbox next to the icon 'Errored' if you wish to be notified when errors arrears in the process or the process fails. For eg: If you enable this option you will be notified when a process such as 'Clean Copy', 'DB to Source', etc encounters errors or fails.

- **Object**

- Added**- Click the checkbox next to the icon 'Added' if you wish to be notified when any new object is detected in your Source or Database. For eg: If you enable this option you will be notified when any new object such as 'Account', 'User' etc is detected on your Salesforce or your Database instance.
- Updated**- Click the checkbox next to the icon 'Updated' if you wish to be notified when any change in the object is detected in your Source or Database. For eg: If you enable this option you will be notified when any change in the object such as 'Account', 'User' etc is detected on your Salesforce or your Database instance.
- Removed**- Click the checkbox next to the icon 'Removed' if you wish to be notified when an object is removed/deleted from your Source or Database. For eg: If you enable this option you will be notified when an object such as 'Account', 'User' etc is removed or deleted from your Salesforce or your Database instance.

- **Fields**

- Added**- Click the checkbox next to the icon 'Added' if you wish to be notified when any new object field is detected in your Source or Database. For eg: If you enable this option you will be notified when any new object (Account) field ('Billing Street'),(Billing City), etc is detected on your Salesforce or your Database instance.
- Updated**- Click the checkbox next to the icon 'Updated' if you wish to be notified when any change in the object field is detected in your Source or Database. For eg: If you enable this option you will be notified when any change in the object (Account) field ('Billing Street'),(Billing City) is detected on your Salesforce or your Database instance.
- Removed**- Click the checkbox next to the icon 'Removed' if you wish to be notified when an object field is removed/deleted from your Source or Database. For eg: If you enable this option you will be notified when an object (Account) field ('Billing Street'),(Billing City), etc is removed or deleted from your Salesforce or your Database instance.

Methods of Configuration Notifications

POST method:

Url	GET ▼	Enter Url Here..	Save
	GET		
	POST		
	Test Webhook		

- This method is used if the user wants to define the payload.
- From the URL drop-down menu choose the POST method option.
- Fill in all the details: URL, HEADERS, CONTENT TYPE, and PAYLOAD

For example:

Let us consider that a user wants to get notified in TEAMS.

Enter details as follows:

- a. URL:

POST

URL: <https://avankia.webhook.office.com/webhookb2/cfef3183-e63b-4f97-a5dc-5c038f23ce19@c86b5726-2a50-4b12-9703-ce9be669e77d/IncomingWebhook/95072733741343179cb2e3f28a4567ae/28fcde24-5b09-4aae-b25d-761b199a6f33>

a. Content-Type:

Content-type:

Content type: application/JSON

a. Payload:

Payload: (Freeform)

```
{
  'text' : 'date=$date; format ="yyyy-MM-dd hh:mm:ss"$ profile=$profile$ command=$command$ zoneld=$zoneld; default="EST"$
  errorCount=$errorCount$'
}
```

- The screen will appear as below:

The screenshot shows a webhooks configuration interface with the following fields and values:

- Url:** A dropdown menu set to "POST" and a text input field containing "Enter Url Here..".
- Headers:** A text input field containing "Application/JSON".
- Content Type:** A text input field containing "Content Type".
- Payload:** A table with one row and one column. The row is numbered "1" and the column contains the JSON payload:

```
{
  'text' : 'date=$date; format ="yyyy-MM-dd hh:mm:ss"$ profile=$profile$ command=$command$ zoneld=$zoneld; default="EST"$
  errorCount=$errorCount$'
}
```

At the bottom of the form, there are two buttons: "Test Webhook" and "Save".

- You can format your calls for multiple parameters.
- The above example: The user required details such as:
 - date, time, profile, and error.
- There is a specific list of supported expressions that have to be followed for writing the various required parameters in the URL while running the POST method.
- These can be found listed under "Supported Expressions". Various conditions for these parameters can also be listed.

- Once you enter all parameters Click SAVE
- Click Test Connection and verify the same.

In this manner, the user can define their own Payload and provide the parameters in the standard expressions format with/without conditions to receive the respective POST notifications in their communication channel.

• Supported Expressions:

Option	Expression	Notes
Current date	<code>\$date;</code> <code>format="dateformat"\$</code>	Format is an optional attribute. However, it is recommended to add one. Do NOT escape quotes in date format as it can lead to errors.
Current event	<code>\$event\$</code>	Values: Started, Errored or Completed
Error count	<code>\$errorCount\$</code>	Denotes the number of objects that encountered an error
Is error in the backup	<code>\$isError\$</code>	Values: true if the event is errored else false
Profile Name	<code>\$profile\$</code>	
Total records processed	<code>\$recordsProcessed\$</code>	Total number of records processed ie. inserted + updated for all objects
Command	<code>\$command\$</code>	Provides a clear representation of the command that has been executed.
Zone ID	<code>\$ZoneId\$</code>	Shows the Time zone selected

Using Conditions:

If else conditions are supported, however, we need to specify a boolean variable from the supported list for conditions:

Syntax:

`$if(variable1)$value1$elseif(variable2)$value2$else$value3$endif$`

Example

`$if(isError)$errorCount$else$0$endif$`

- Once All the steps are done you can Go to the team/channel selected to receive the notification and see if the notification is activated and received on the same as shown below

