Webhooks

Overview

Organizations might find it hard to keep track of the various processes in progress or errors that might occur. Email notifications for when a process is complete is currently provided as an in-built feature but might not be sufficient to keep track of errors, or start, and finish of a process. Also receiving notification within the server of the company or any communication channel such as Slack can be extremely beneficial to the users to keep track of all events. It is also very essential for the user to be notified about the three most important stages of an activity:

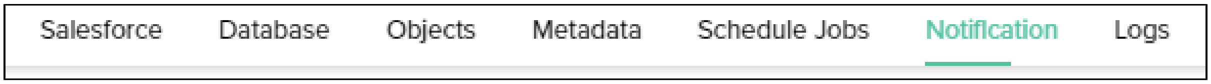
1. When the Job is started
2. When the Job is completed
3. When there is an error

Webhooks notification system solves these issues to the users and makes the process of keeping track of the various events much easier.

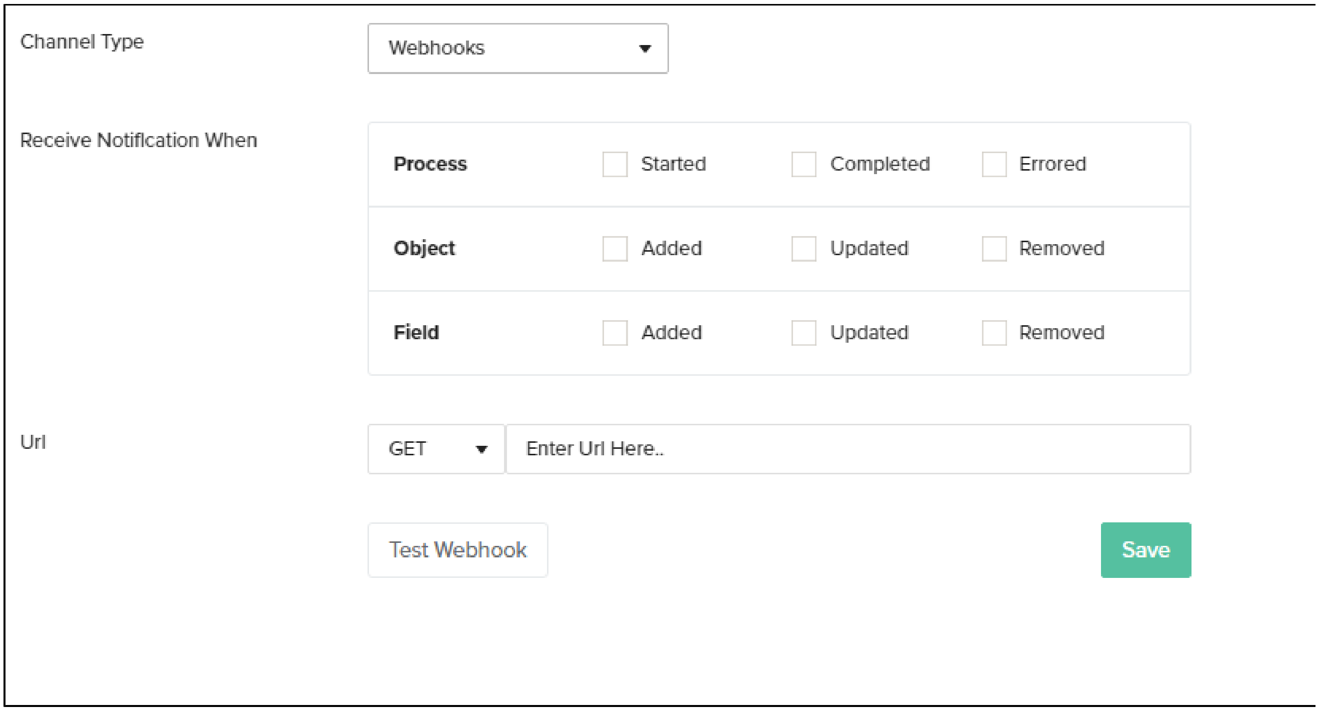
Setting up of Webhooks

To set up webhooks:

- Login to DBSync Cloud Replication instance
- Select the desired profile for which you want to activate Webhooks notification system
- From the Top Navigation toolbar Click on **'Notifications'**



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



Channel Type: Webhooks ▾

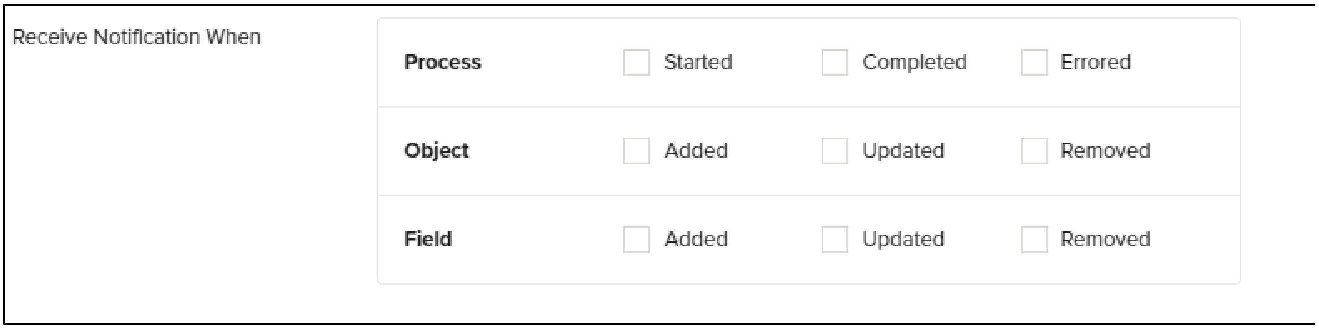
Receive Notification When

Process	□ Started	□ Completed	□ Errored
Object	□ Added	□ Updated	□ Removed
Field	□ Added	□ Updated	□ Removed

Url: GET ▾ Enter Url Here..

Configure Webhooks Notifications Settings

1. **Receive Notifications**
   Click the check-box 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	□ Started	□ Completed	□ Errored
Object	□ Added	□ Updated	□ Removed
Field	□ Added	□ Updated	□ Removed

- **Process**

a. **Started**- Click the check-box 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 'Clean Copy', 'DB to Source' etc starts running.
b. **Completed**- Click the check-box 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 'Clean Copy', 'DB to Source' etc completes.
c. **Errored**- Click the check-box next to the icon 'Errored' if you wish to be notified when error arrears in the process or the process fails. For eg: If you enable this option you will be notified when a process such 'Clean Copy', 'DB to Source' etc encounter errors or fails.

- **Object**

a. **Added**- Click the check-box 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.
b. **Updated**- Click the check-box 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.
c. **Removed**- Click the check-box 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.

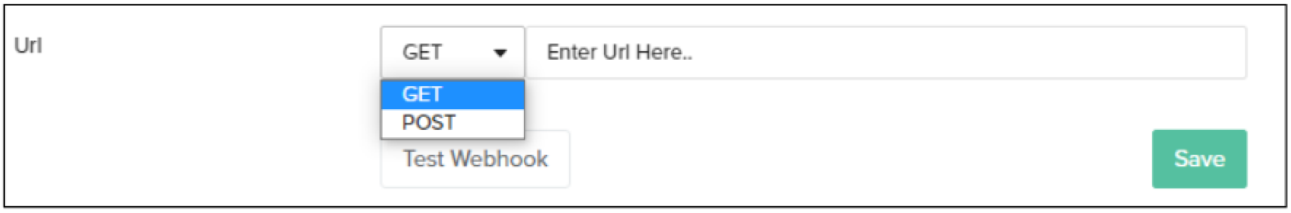
- **Fields**

a. **Added**- Click the check-box 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.
b. **Updated**- Click the check-box 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.
c. **Removed**- Click the check-box 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.

2. **Methods of Configuration Notifications**

There are 2 methods you can use to configure Webhooks:- CHANGE SCREEN (DUE TO NOTIFICATION SCREEN CHANGE)

a. **GET method:**



- This method is used if you want to use your own API to get notified using webhooks
- From the URL drop-down menu choose the GET method option.
- URL (supporting variable injection) - notification.webhook.url
- This method allows parameters in URL
- Example of a URL for Get method:

Method: GET

https://<www.url.com>?computername=<serverName>datetime=$date;

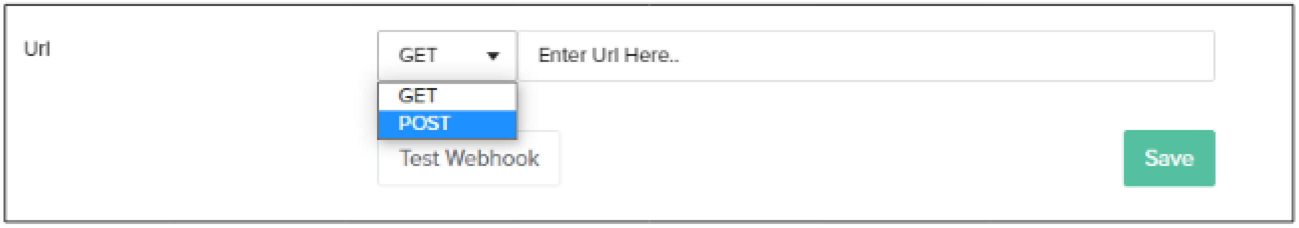
format ="yyyy-MM-dd'T'hh:mm:ss"$&profile=$profile$&errorCount=$if(isError)$$errorCount$$else$0$endif$&recordProcessed=$recordsProcessed$

- You can format your calls for multiple parameters.
- The above example: The user required details such as:

Computer name, date, time, profile details, error count and number of records processed

- There is a specific list of supported expressions which has to be followed for writing the various required parameters in the URL while running the GET 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

b. POST method: CHANGE SCREEN (DUE TO NOTIFICATION SCREEN CHANGE)-CROP



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

For example:

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

Enter details as follows:

a. URL:

Slack

POST

URL: https://hooks.slack.com/services/T00000000/B00000000/XXXXXXXXXXXXXXXXXXXXXXXX

a. Content Type:

Content-type:

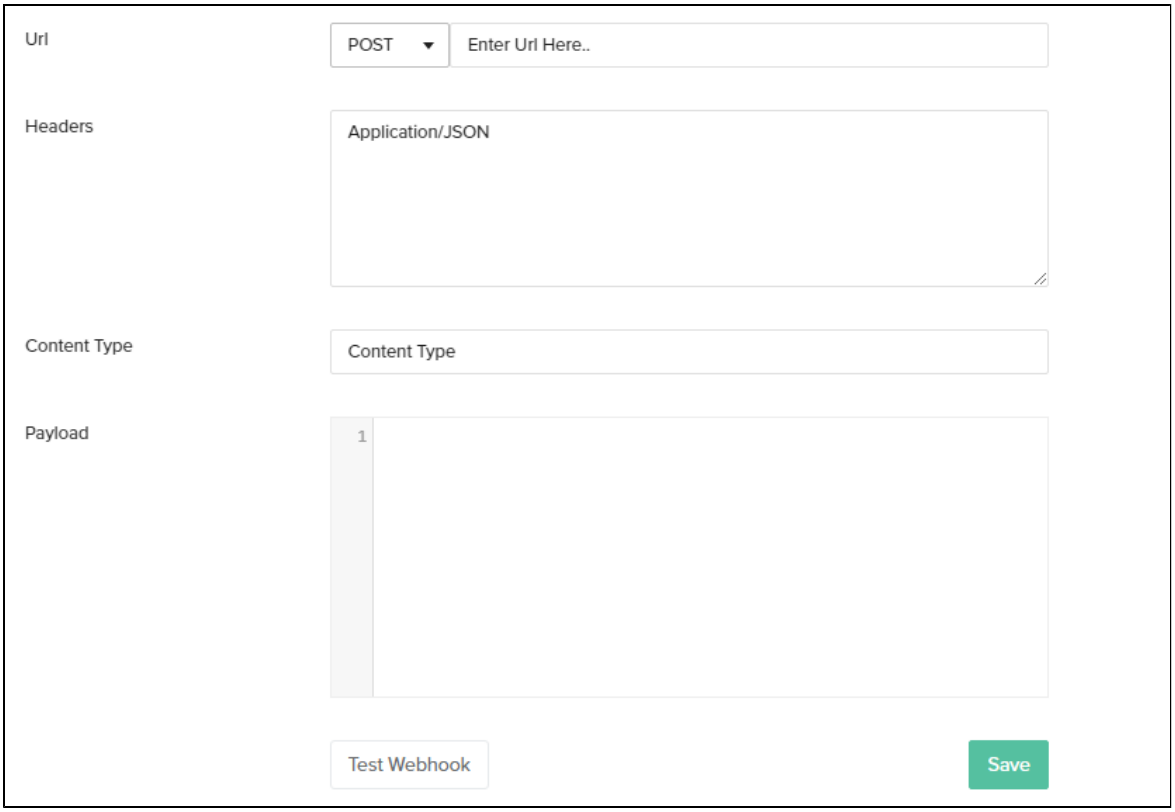
Content-type: application/json

a. Payload:

Payload: (Freeform)

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

- The screen will appear as below:



Url POST Enter Url Here..

Headers Application/JSON

Content Type Content Type

Payload 1

- 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 has to be followed for writing the various required parameters in the URL while running the GET 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 such as in this case SLACK.

NOTE:

- In both the methods if the user has check-boxes to receive notification when STARTED, immediately when the connection is SAVED and TESTED a notification will be sent to the SERVER or to the communication channel such as SLACK that the process has STARTED.
- To get the URL for POST method for a communication channel such as SLACK please refer to the following link https://api.slack.com/messaging/webhooks .

Supported Expressions:

Option	Expression	Notes
Current date	$date; format=”dateformat”$	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	$event$	Values: Started, Errored or Completed
Error count	$errorCount$	Denotes the number of objects that encountered an error
Is error in the backup	$isError$	Values: true if the event is errored else false
Profile Name	$profile$	

Total records processed	$recordsProcessed$	Total number of records processed ie. inserted + updated for all objects
Command	$command$	Provides a clear representation of the command that has been executed.
Zone ID	$ZoneId$	Shows the Time zone selected

Using Conditions:

If else conditions are supported, however we need to specify 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$