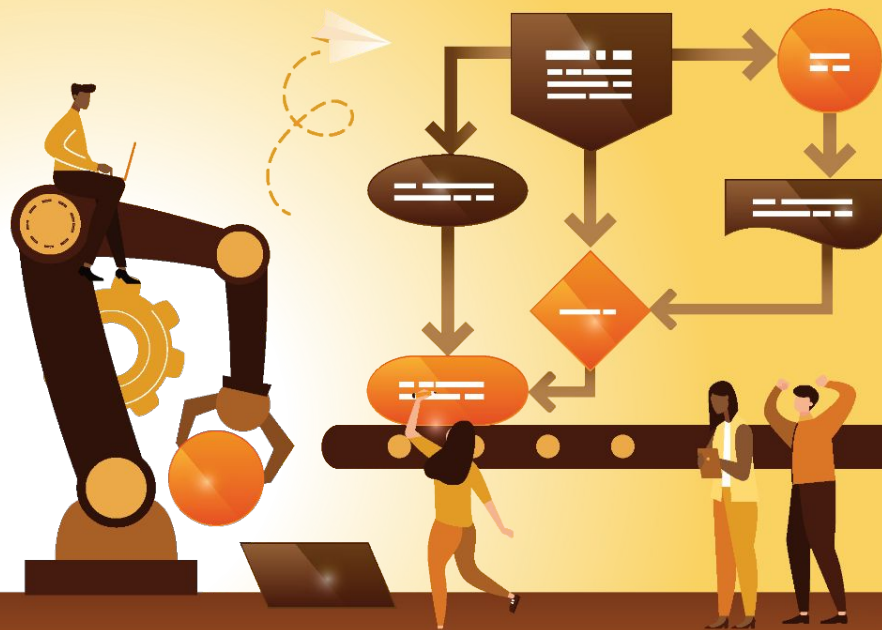


# Getting Started With **ZERION DATAFLOW**



*Presenter:*  
**Berit Weymers**  
*Customer Success Manager*



*Moderator:*  
**Liza Armstrong**  
*Marketing Manager*

# GoToWebinar Control Panel

The screenshot shows the GoToWebinar control panel with the following callouts:

- Hide/Collapse Control Panel:** Points to the right-pointing arrow icon in the top-left corner of the control panel.
- Microphone Status: Muted (red):** Points to the red microphone icon with a slash through it, indicating the current audio status.
- Make Webinar Full Screen:** Points to the full-screen icon (two overlapping rectangles).
- Raise Hand or Ask Presenter for Audio Rights:** Points to the hand icon with a green dot, used for requesting audio permissions.

Below the control panel, a text box says: "Type questions or comments anytime during the webinar here", with an arrow pointing to the "Questions" section of the interface.

The "Questions" section contains a text input field with the placeholder "[Enter a question for staff]" and a "Send" button.

At the bottom of the control panel, it displays: "Multi sessions different registrants", "Webinar ID: 980-960-603", and the "GoToWebinar" logo.



# — AGENDA —

- Zerion Dataflow in a Nutshell
- Building a Dataflow
- Dataflow Best Practices
- Resources



# POLL

# Dataflow Automation in a Nutshell

Dataflow Automation (DFA) is a Zerion product that provides Big Data capabilities including long term storage, data management, processing and integration.

It allows you to centralize data throughout your organization and build actions to send, update, create, delete with the following connections and services:

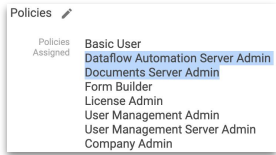
- iFormBuilder
- Documents
- SQL Server
- Email
- HTTP Post
- Formstack
- ZoHo

# Dataflow Automation Key Terms

What is a...	
<b>Dataflow</b>	A diagram that maps the movement of data. Similar to a workflow, but with an emphasis on data rather than processes. A dataflow can be thought of as a container for the project.
<b>Dataflow Object</b>	An object on the Dataflow. There are four types of objects: Webhook, RecordSet, Action, and Event.
<b>Record</b>	A single line of data. A record is structured as a JSON object and is analogous to a row in a table.
<b>JSON</b>	Short for JavaScript Object Notation, this is the structure that records are organized in.
<b>Run/ Transaction</b>	When data moves through a Dataflow. Tracked by a value known as a RUN_ID.

# Dataflow Automation Overview

1. Access Dataflow by logging in to [console.zerionsoftware.com](https://console.zerionsoftware.com). Must have Dataflow policy assigned to username from iFormBuilder User Management.

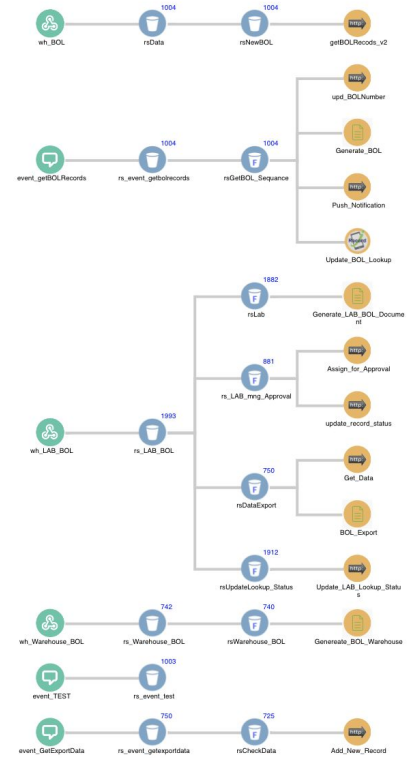


2. Data is Posted via Webhook, which is created in DFA or an event (event is based on if an action is successful or if it failed).

3. Data is placed in a RecordSet bucket.

4. Data is refined and transformed with Filters making use of JavaScript.

5. Refined data is configured to perform specified actions including: being sent to SQL Server DB, Documents, iFormBuilder updates (forms, lists, users, etc), HTTP Post and HTTP Request.



# Common Dataflow Use Cases

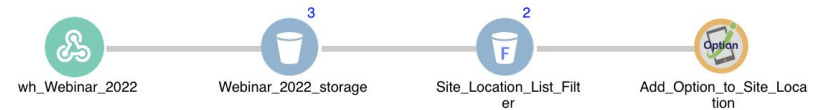
- Update Option Lists based on users adding text to “Other” in form.
- Have iFormBuilder data cleansed, filtered or flattened and automatically sent to specified SQL Database.
- Re-assign or Assign iFormBuilder Records to users based on specified criteria defined in Dataflow.
- Send Filtered or Transformed iFormBuilder Data to Documents for custom PDF.



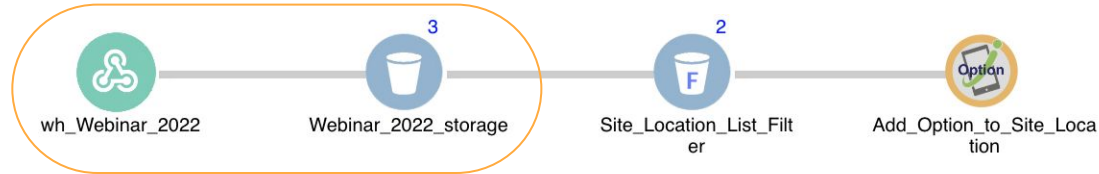


# Getting Started with Building a Simple Dataflow

Need: Ability to update an Option List when users type in a new option in the “Other” text field in iForm.



# Building a Dataflow Steps



## CONNECTIONS, WEBHOOK & RECORDSET

1. Log-in to [console.zerionsoftware.com](https://console.zerionsoftware.com).
2. Set-up Connection(s) (connecting to iFormBuilder for Demo). If connected, you do not need to set-up again.
3. Create New Dataflow with RecordSet and Webhook.
4. Copy Webhook URL. In iFormBuilder Form Assignment, select parent form and add Post Data Endpoint for Dataflow. Paste Dataflow Webhook and save.
5. Under Data, select form and click List. Click “Trigger Post Processing”.
6. Check that records have reached Dataflow.

# Building a Dataflow Steps



## FILTER

7. Add a new RecordSet to end of Dataflow. This will be used for our Filter.
8. In original RecordSet, copy data from “Record Data”.
9. Go back to the RecordSet that was added and add a Filter under Data Refinery menu.
10. Step 1- Paste copied data in “Sample Data”.
11. Step 2- Add a default filter to point to Option List DCN. This filter is used to filter out all options that are “Other” in the list. Example: `(record.site_location_categories === "other")`
12. Step 3- Run Preview. If it is configured correctly, you will only see record data if the option list has “other” selected and you will see [] if “other” is not the selection. You can edit Sample Data to test. Save.
13. Manually Rerun storage RecordSet.

# Building a Dataflow Steps



## iFormBuilder OPTION LIST ACTION

14. Add Form Builder Option List Action under Action Menu.
15. Name the action and select the “Connection” that was added earlier.
16. Under Command, select “Create or Update”
17. Copy Profile ID from iFormBuilder and Paste in Profile ID. Select Option List from drop-down.
18. In Option List Attributes, Update Key Value and Label to point to the “other” text element in the form. Example: `{{record.other_option}}`. Save.
19. Complete Record on Device and test with adding “Other Location” text.
20. Check Dataflow for new record.
21. Sync device to check for latest option list addition.



# POLL

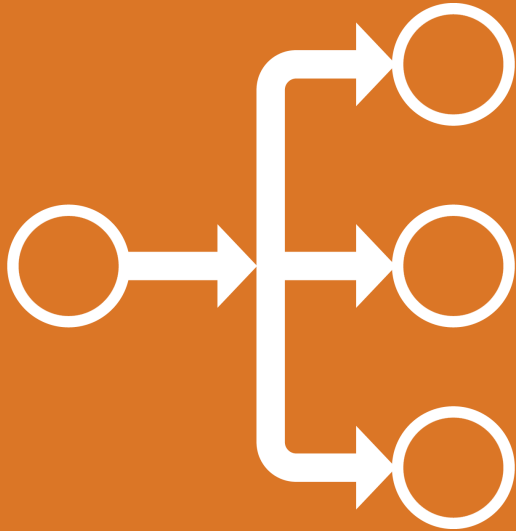
# Dataflow Best Practices



# Dataflow Best Practices

- **Plan:** Draw Dataflow in either a flowchart software or on a piece of paper prior to building. Building the dataflow should be following a blueprint that you made, not building it as you go.
- **Naming Convention:** Use a naming convention that makes the flow of data clear. For example, if you have an HTTP Request called **GET\_Username**, name the event that the result goes to **event\_GET\_Username** to show that the two are connected, and one is fed directly by the other.
- **Record Storage:** Do not apply transformations/filters on the RecordSet connected to a webhook/event. Use the RecordSet for storage of original data, without alterations.
- **Filters:** Use Filters on RecordSets instead of Actions if you're not confident (easier to troubleshoot).
- **Transformation Code:** In transformation code, assign all variable values at the top then ideally you don't have to edit any of the code below even when small data changes (like Profile Id or Page Id) occur.
- **TEST A LOT.** Use expected values, null values, long values, short values, and empty values.

# Dataflow Resources



- [Getting Started Guide](#)
- [Dataflow Automation Articles](#)
  - [Dataflow Objects](#)
  - [RecordSet](#)
  - [Webhook](#)
  - [Action](#)
  - [Event](#)
  - [SQL Connection](#)
- [Dataflow Automation Video Playlist](#)





# POLL

# THANK YOU!



/zerion



Zerion



@zerionsoftware