

# Dynamics Sink in dataflow

Last updated by | Zhuoyang Zhang | Apr 27, 2022 at 2:05 AM PDT

Dynamics connector is based on Generic Rest connector in backend, so it can be verified with PostMan and help trouble shoot.

## 1. How Dynamics sink worked in dataflow?

Dynamics Connector Backend implemented by rest api, so cx's error message is mostly about http request failure.

If you are not familiar with how to trouble shoot with rest, you can refer this TSG first:

[https://supportability.visualstudio.com/AzureDataFactory/\\_wiki/wikis/AzureDataFactory/614170/Generic-Rest-sink-in-dataflow](https://supportability.visualstudio.com/AzureDataFactory/_wiki/wikis/AzureDataFactory/614170/Generic-Rest-sink-in-dataflow)

## 2. How to trouble shoot?

Most Dynamics Sink failure is caused by fields name or value is not correct. Example Error Message:

```
{"StatusCode":"DFExecutorUserError","Message":"Job failed due to reason: DF-REST 001 - Rest - Error response received from the server (url: https://admsdynamics.crm.dynamics.com/api/data/v9.0/new_dataflow_orders, request body: Some{"new order id":"90001","new product id":183f8baa-0833-ec11-b6e5-000d3a5c463a,"new floating point num":9.99,, "new customer contact@odata.bind":"/contacts(216ce57e-17bb-ec11-983f-000d3a5c1df1)","new_name":"test_item"})), request method: POST, status code: 400)
```

From the error message, we can get the request url and request body information. And exception message should have some hints to tell which field name is not correct.

Go to dynamics portal and open the developer tool, do the same CRUD operation and check the request body. Compare your request payload from portal and the request body in your error message. And correct your column name/value.

The screenshot displays the Dynamics 365 developer tool interface. On the left, a form titled 'New dataflow\_order' is shown with various fields. The 'Name' field is set to 'test\_item'. The 'Owner' is 'ADMS Dynamics365'. The 'col\_customer' field is set to 'zhuoyang zhang'. The 'col\_datetime\_zone\_id' field is set to '4/13/2022'. The 'col\_date\_local' field is set to '4/14/2022'. The 'col\_decimal\_num' field is set to '---'. The 'col\_floating\_point\_num' field is set to '9.99'. The 'col\_multi\_option\_set' field is set to '---'. The 'col\_multi\_line\_text' field is set to '---'. The 'col\_option\_set' field is set to 'Item2'. The 'col\_pricing (Base)' field is set to '---'. The 'col\_pricing' field is set to '---'. The 'col\_two\_options' field is set to 'Yes'. The 'col\_whole\_number1' field is set to '---'. The 'look\_product' field is set to 'apple'. The 'order\_id' field is set to '90001'. On the right, the Chrome DevTools Network tab is open, showing a recording of network activity. The console shows a message: 'Recording network activity... Perform a request or hit Ctrl + R to record the reload. Learn more'.

The screenshot displays the Dynamics Sink configuration in the dataflow tool. The left pane shows the 'test\_item' configuration with fields like col\_customer, col\_datetime\_zone\_id, col\_date\_local, col\_decimal\_num, col\_floating\_point\_num, col\_multi\_option\_set, col\_multi\_line\_text, col\_option\_set, col\_pricing (Base), col\_pricing, col\_two\_options, col\_whole\_number1, look\_product, and order\_id. The right pane shows the 'new\_dataflow\_orders' table structure with columns: Name, Headers, Payload, Preview, Response, Initiator, Timing, and Cookies. The 'Payload' column is highlighted, showing a JSON payload for a new order.

Example:

Portal request payload:

```
{
  "new_order_id": "90001",
  "new_look_product@odata.bind": "/new_products(183f8baa-0833-ec11-b6e5-000d3a5c463a)",
  "new_floating_point_num": 9.99,
  "new_customer_contact@odata.bind": "/contacts(216ce57e-17bb-ec11-983f-000d3a5c1df1)",
  "new_name": "test_item"
}
```

**RCA:** column name in sink table should be "new\_look\_product" instead of "new\_product\_id".

**Solution:** Cx should rename the column "new\_product\_id" to "new\_look\_product".

**3. Known Limitation:** For lookup datatype, currently we can only support default primary key. For alternate key, we have a workaaround to support.

**Example1:** value of new\_look\_product is not default primary key of entity new\_product, but alternateKey(new\_product\_name) of entity new\_product.

new\_look\_product: product\_name\_1

Workaround:

add derived transformation to change column name to "new\_look\_product@odata.bind", change column value to "/new\_products(new\_product\_name='product\_name\_1')"

**Example2:** if lookup field has 2 alternate key columns, e.g., new\_product\_name, new\_product\_number

Workaround:

add derived transformation to change column name to "new look product@odata.bind", change column value to "/new\_products(new\_product\_name='product\_name\_1', new\_product\_number='product\_number\_1')"