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
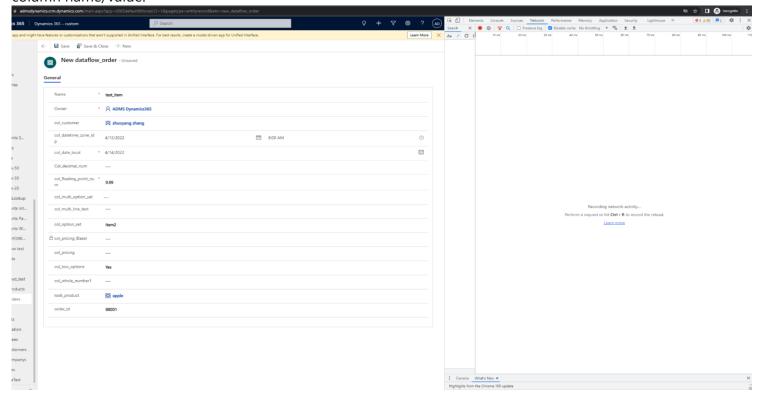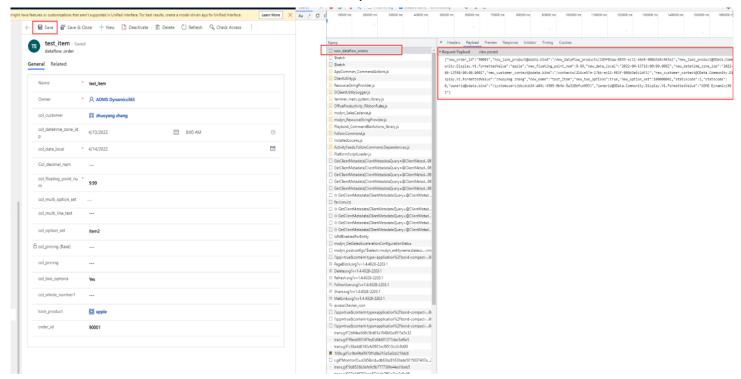
2. How to trouble shoot?

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.

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 workaournd 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')"
```