Customer Voice Response on the Contact Form

My user case was quite simple. I wanted a **sub-grid** on my **contact** that showed all the survey responses that had been completed with the relevant fields in the view. From these I want to create a number field that calculated the average NPS score.

Customer Service Responses(Regarding)

When trying to add a **subgrid** in the F**orm Designer** it shows the **Customer Voice, Alert, Invites and Responses** as **regarding** fields on these entities.

This means that for this **subgrid** to populate the **Customer Voice Survey Response** must be **regarding** the Contact and this field doesn't populate automatically.
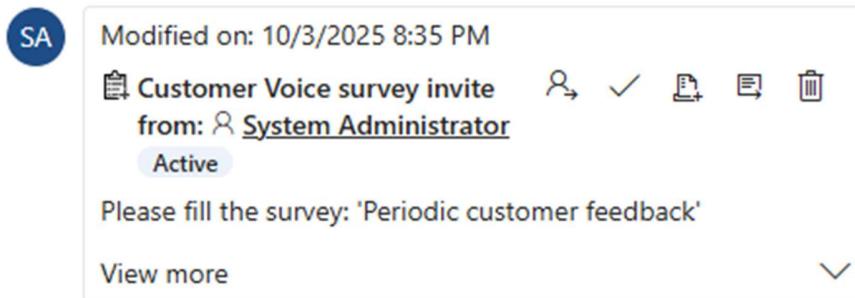
Customer Voice Alert, Invite and Responses are Activities

The Customer Voice Alert, Customer Voice Invite and Customer Voice Response are **activities** that end up in the Timeline of a Contact. The invite is an activity type and uses the **"TO"** field and the response is an activity type and uses the **"FROM"** field. Both fields are **PARTYID** fields and polymorphic in nature.

Invite:

Response:

Testing Sending Invites Via Email Client

One thing that I realized early on is that the **Customer Voice Responses** when sent out via the inbuilt email client doesnt use the regarding field. It is only available if I used the advance upload .csv file. This file came with RegardingID,RegardingEntityName in the download.

This is a hassle because because most end users know nothing about getting RegardingIDs and RegardingEntityName or:

"_partyid_value@Microsoft.Dynamics.CRM.lookuplogicalname": "contact",

 "_partyid_value": "5a25e427-1892-f011-b4cb-000d3a170c2c",
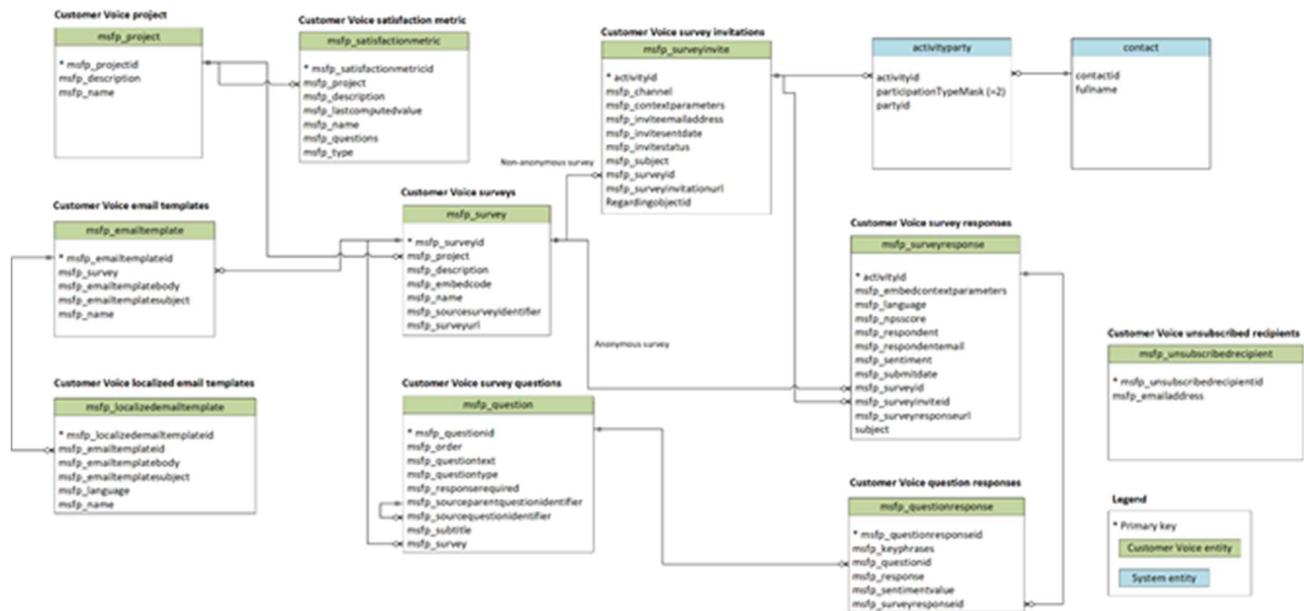
In the start I wanted to test the Customer Voice and this is the quickest way. Once Customer Voice and Dynamics is hooked up then distributing the Survey via the email client with known(or unknown) contacts will create the invite and the potential response as shown about in the TO/FROM field on the respective activity table.

Entity Map
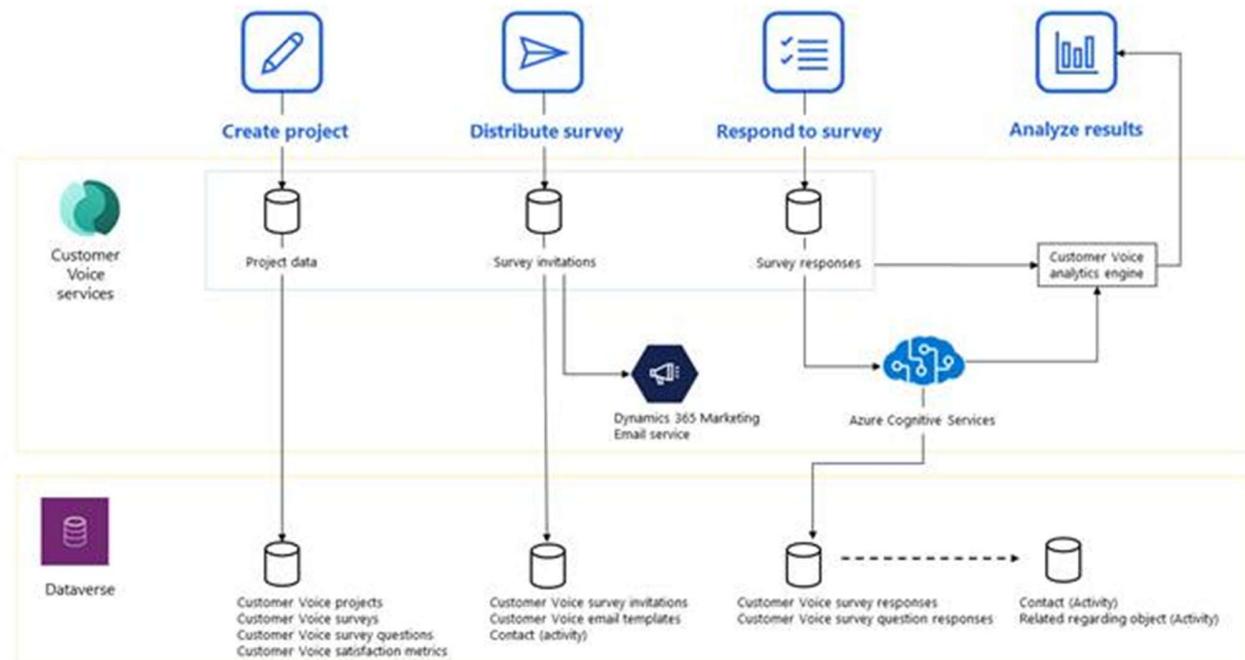
This is important as all the Customer Voice Survey Entities are in Dynamics 365 and these can be shown on records and used in varies ways to notify about the Survey Responses, Metrics and Alerts. The Entity Reference is here: **https://learn.microsoft.com/en-us/dynamics365/customer-voice/developer/entity-reference**

Data Flow

Also, understanding the flow of data as it relates to the survey and how to find these entities in the system is also important. The Data Flow is in this diagram and found here: **https://learn.microsoft.com/en-us/dynamics365/customer-voice/data-flow**

5

I Added the Important Entities to the Sales App in a New Areas

It should be noted that from the data flow diagram that the flow is one way from Customer Voice App to dataverse and the documentation states not to delete data that surveys rely on via dataverse like metrics.

Back to My User Story.
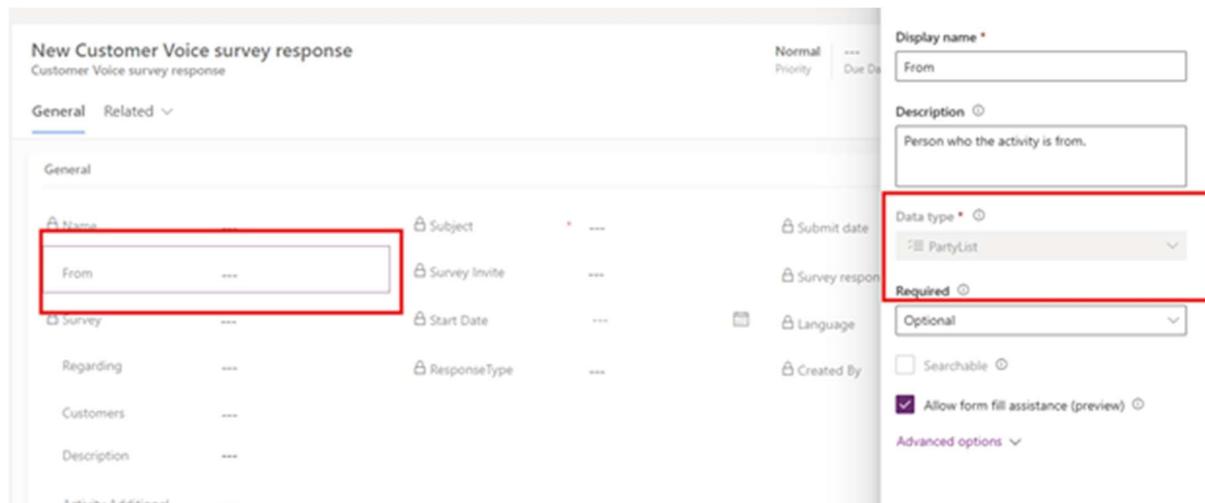
I want to leverage the out of the box relationships and populate the regarding field on the Customer Voice Response automatically. That is to copy the data from the **"FROM"** field to the "**REGARDING**" fieldon the Customer Voice Response.

How Hard Can This Be?

My original thought was to make a business rule that populated the regarding field.
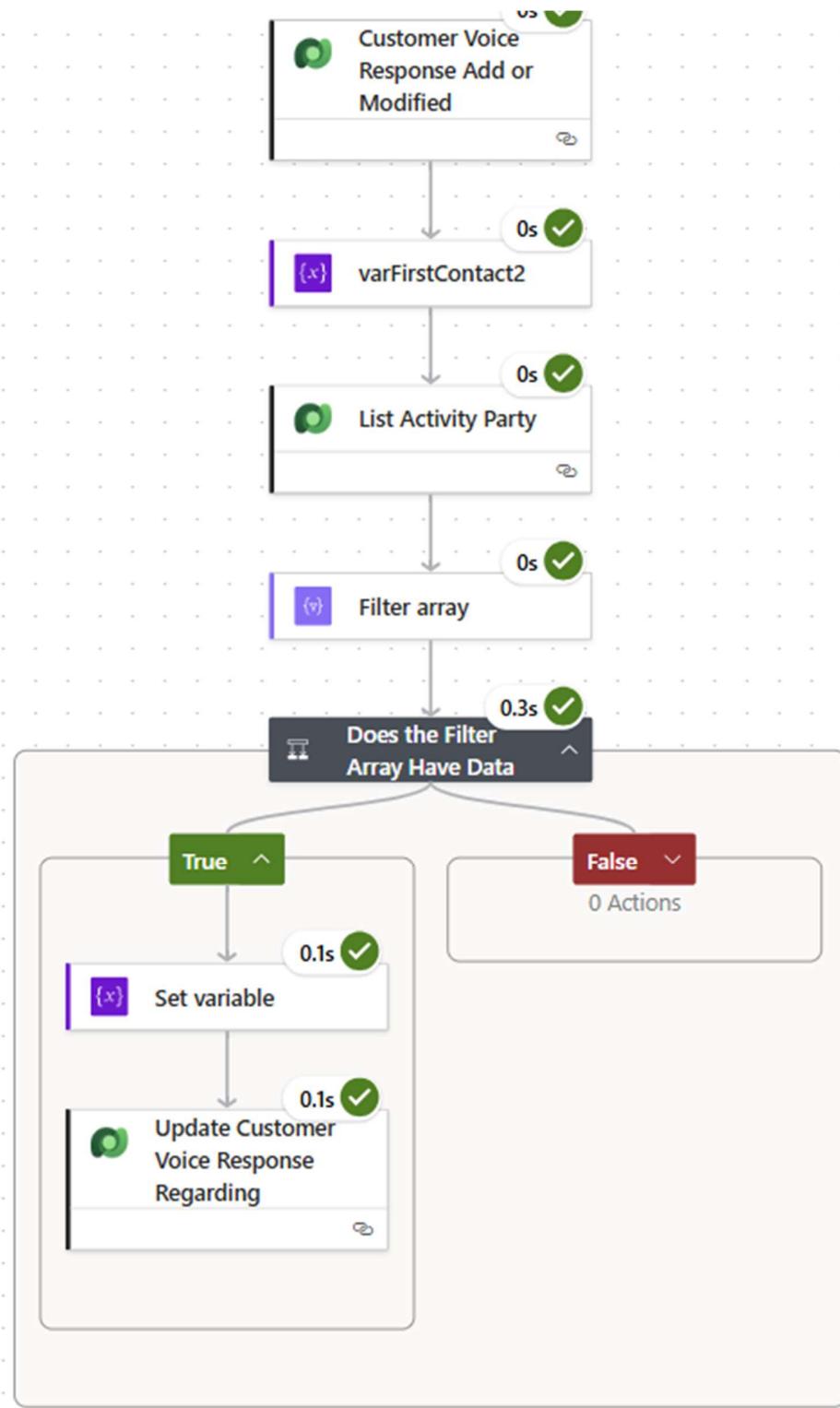
As it turns out that this is not possible because both are polymorphic and point to more than one entity and the **FROM** field in a **PartyID** field which is a special field Dataverse

Because of this I couldn't use a business rule to what I needed and I add to turn to power automate. Due to the use of an activity entity it was a bit tricky to figure out and thanks to ChatGPT – I was able to get this to work.

Power Automate:

```
                    ┌──────────────────────────┐
                    │  ◉  Customer Voice        │  0s ✅
                    │     Response Add or       │
                    │     Modified             │
                    │                      🔗  │
                    └──────────────────────────┘
                               │
                               ▼
                    ┌──────────────────────────┐  0s ✅
                    │ {x}  varFirstContact2     │
                    └──────────────────────────┘
                               │
                               ▼
                    ┌──────────────────────────┐  0s ✅
                    │  ◉  List Activity Party   │
                    │                      🔗  │
                    └──────────────────────────┘
                               │
                               ▼
                    ┌──────────────────────────┐  0s ✅
                    │ {▽}  Filter array         │
                    └──────────────────────────┘
                               │
                               ▼
                    ┌──────────────────────────┐  0.3s ✅
                    │  ⊼  Does the Filter    ∧ │
                    │     Array Have Data       │
                    └──────────────────────────┘
                       │                    │
                ┌──────┴──────┐      ┌───────┴────────┐
                │  True    ∧  │      │  False      ∨  │
                └─────────────┘      │   0 Actions    │
                       │             └────────────────┘
                       ▼
             ┌──────────────────┐  0.1s ✅
             │ {x}  Set variable │
             └──────────────────┘
                       │
                       ▼
             ┌──────────────────────┐  0.1s ✅
             │  ◉  Update Customer   │
             │     Voice Response    │
             │     Regarding         │
             │                  🔗   │
             └──────────────────────┘
```

8

Basically the flow finds the **Activity Party Value** and **GUID** of the contact and makes it possible to populate the regarding field.

List Activity Parties

When a Customer Voice Survey Response is made I am listing the activity parties this filter to grab the **"FROM"** (participationtypemask eq 1) Field:

This returns a bunch of activity party data from the Customer Voice Response Activity and I am my case I am particularly looking for this chunk:

"_partyid_value@OData.Community.Display.V1.FormattedValue": "Patrick Kitchell",

"_partyid_value@Microsoft.Dynamics.CRM.associatednavigationproperty": "partyid_contact",

"_partyid_value@Microsoft.Dynamics.CRM.lookuplogicalname": "contact",

"_partyid_value@odata.type": "#Guid",

"_partyid_value": "5a25e427-1892-f011-b4cb-000d3a170c2c",

This field "_partyid_value": "5a25e427-1892-f011-b4cb-000d3a170c2c"," is the contact ID that is populated in the **"FROM"** Field

Now that I have this information - I need to be able to grab it and move it. I am using the

Filter Array to grab this information:

Parameters   Settings   Code view   About
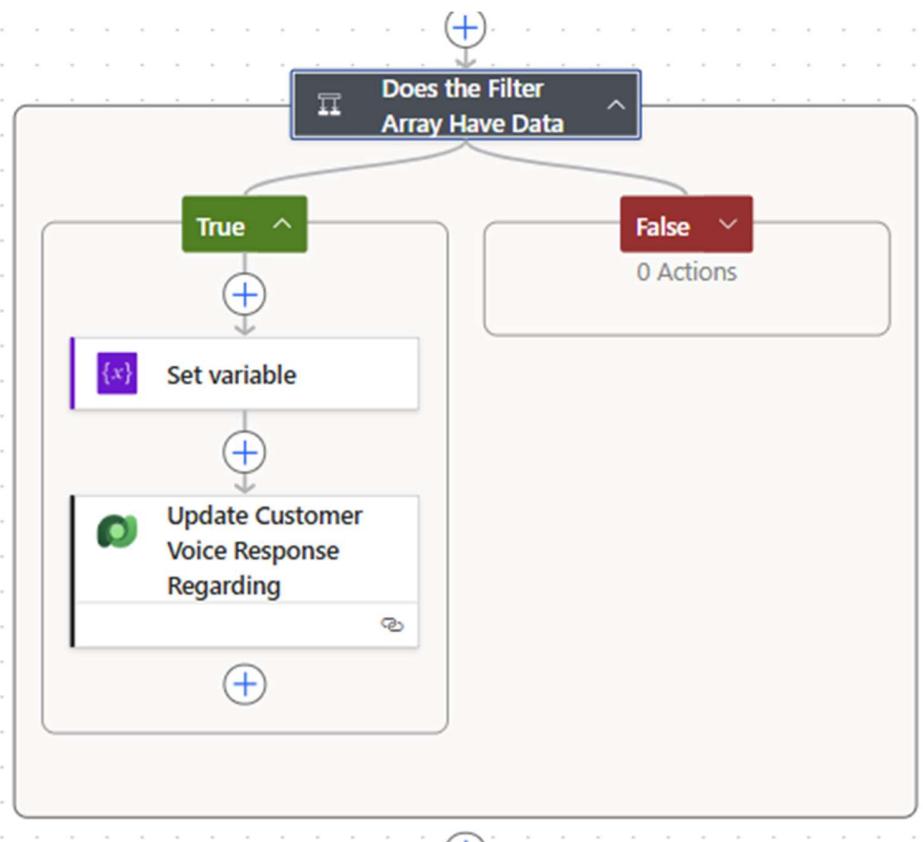
From *

body/value ✕

Filter Query *

_partyid_value@... ✕   is equal to   ⌄   contact

Edit in advanced mode

Like many polymorphic fields this field could hold data from another entity than a Contact and therefore I am using **a condition** to check if it returns contact data when listing the activityparty or not. If not than the condition is False.

Looking closer at the **Filter Array** it grabs PARTYID Value = CONTACT
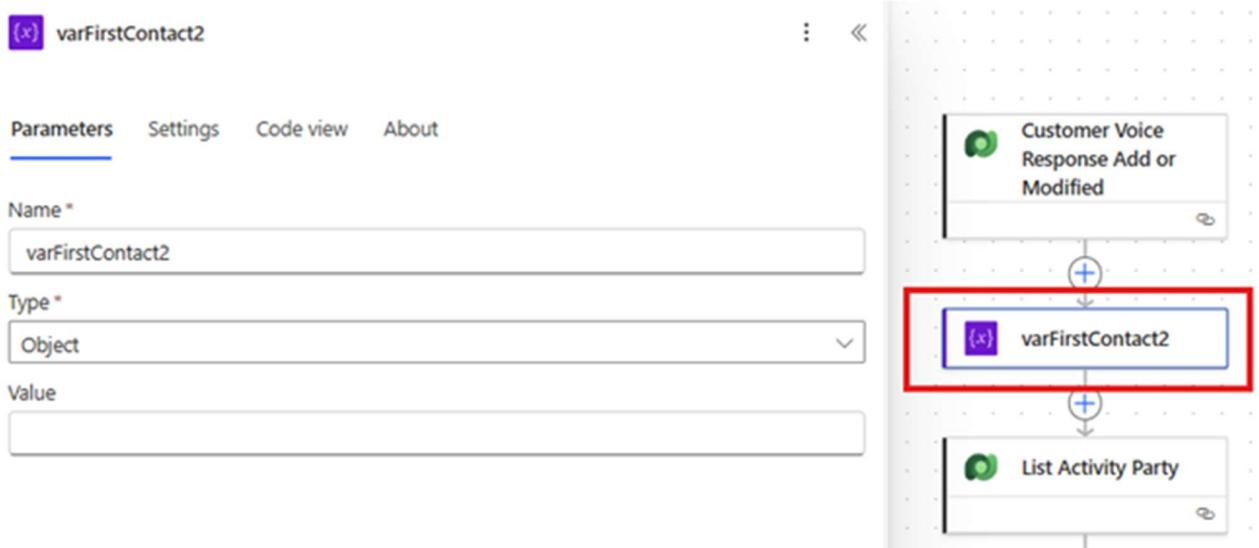
"_partyid_value@Microsoft.Dynamics.CRM.associatednavigationproperty": "partyid_contact",

"_partyid_value@Microsoft.Dynamics.CRM.lookuplogicalname": "contact"
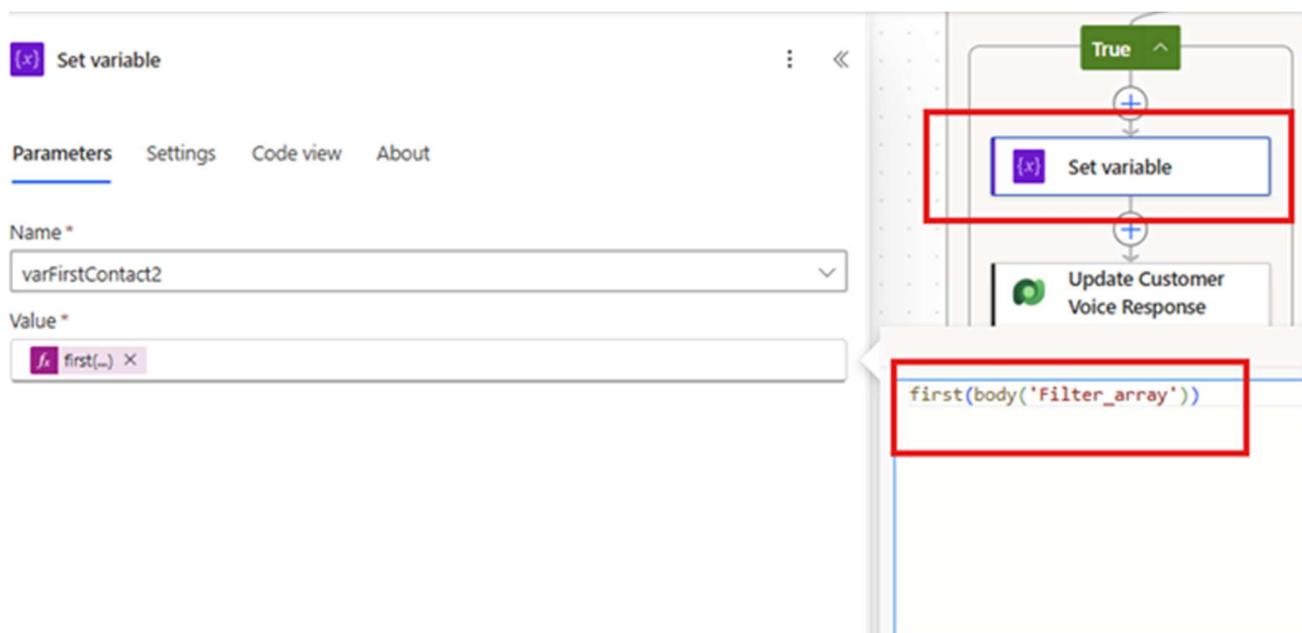
If the Filter Array body doesn't contain this than it is False in my Power Automate.

VarFirstContact2

Earlier in the Flow I created an empty object variable that will be populated with GUID of the from PARTY VALUE that we capture in the Filter Array.
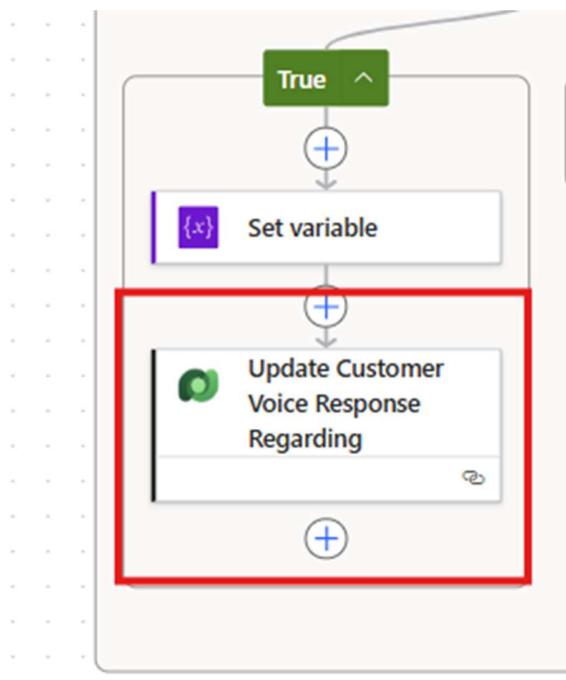

If the condition is true in that the Filter Array has PartyID Values this it sets the GUID in this variable

Set Regarding

The last step is to update the Customer Voice Response Regarding with the following:'

/contacts(@{variables('varFirstContact2')?['_partyid_value']})

Final thoughts

This a huge work around for something that I was hoping was more simple, less complex. The good thing about the power automate is that it can be used on all activities with minor modifications