VOICE FOR SALESFORCE

BY ENREACH

Smart Routing configuration guide

Version 1.50

Contents

Introduction

Smart Routing functionality enables customer to dynamically route incoming phone calls based on Salesforce data.
Smart Routing is typically activated on voice queues level and requires activation also at Enreach cloud configuration side.
This configuration is optional and only used when Smart Routing is needed.

Native Salesforce Flows and Flow Builder are used to configure actual routing logic. Managed package includes two predefined Flow templates, which can be used and modified for specific customer scenario, when activating Smart Routing. Customer can freely create their own Scenario flows too.

The below diagram describes the overall design.
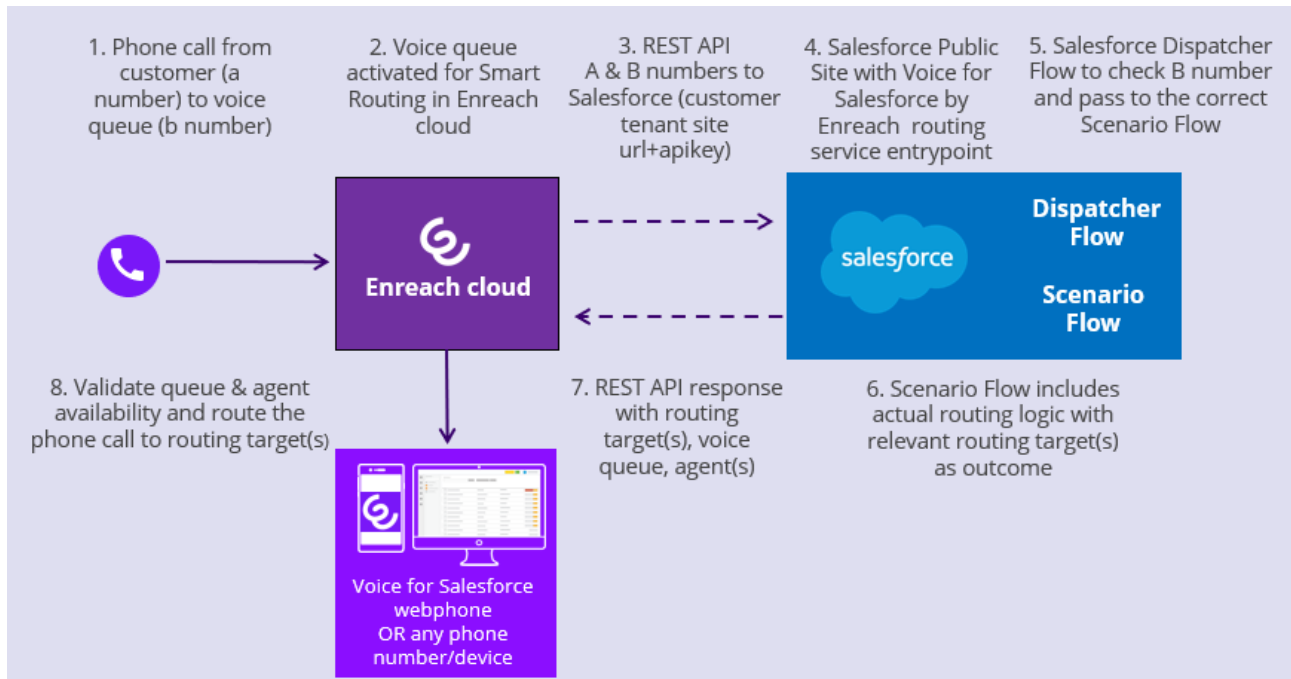
Configuration

Components description

Smart routing implementation consists of the following components in the Salesforce side:

a) Site:
- o Public site is used for accessing to Smart Routing endpoint on Salesforce side.

b) Classes:
- o RoutingInfoService – this class incorporates POST endpoint handling. It receives POST request with anum (caller A phone number), bnum (called B phone number, typically voice queue phone number) and apikey, verifies the apikey and pass anum and bnum to a flow. Once the flow is finished RequiredTarget, InitialTarget, FailTarget, AllocationTargets and Priority are got from the flow and placed to the response.

c) Flows:
- o SmartRouting_Dispatcher – this flow should be created or cloned and updated from the provided template by customer's admin. The goal of this flow is receiving of anum and bnum and passing it to the correct scenario flow depending on hardcoded bnums inside SmartRouting_Dispatcher flow. There aren't restrictions about name of this flow, the name can be any.
- o SmartRouting_Scenario - this flow should be created or cloned and updated from the provided template by customer's admin. It incorporates the actual business logic for call routing. This flow gets anum and bnum from the Dispatcher and process the flow logic. There can be several flows

for different processes and queue numbers (bnums). There aren't restrictions about name of this flow, the name can be any.

d) Fields:
- o Enreach Configuration.SmartRouting REST API Key – this field stores apikey on Salesforce side. Customer can set any apikey.
- o Enreach Configuration.SmartRouting Flow - this field stores name of the initial flow - dispatcher, it is entry point for the flows, usually it is 'SmartRouting_Dispatcher' (or the name of the dispatcher flow which was created by customer's admin).
- o These fields are configured via Voice for Salesforce lightning app -> Smart Routing tab

Configure Public Site

There are two options creating public site.

- The recommended option is to create dedicated public site, which is used only for Smart Routing purposes
- Some of the existing sites can be used for Smart Routing Service, but Site limits should be considered.
- Site Limits, each request to Salesforce endpoint will consume the following limits of the site:
  - o Service request time (per rolling 24-hour period per site) - time which a request took will be counted to this limit
  - o Page views - each request is treated as 1 page view

1) Go to Setup > User Interface > Sites and Domains > Sites and click New.

2) Set the name and fill the required fields, then save it.

3) Click Activate
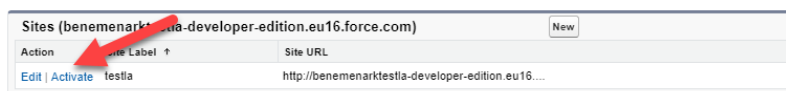
4) Copy the created site url and provide it to Enreach. This site url is used by Enreach at Enreach cloud side as REST endpoint URL when activating Smart Routing.
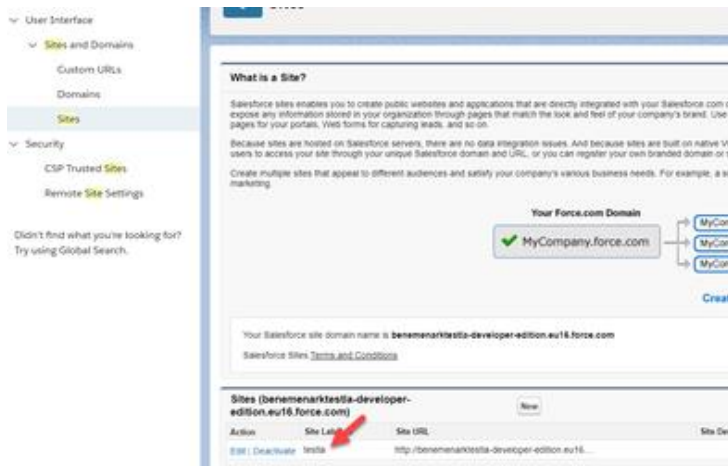
Note: Overall REST endpoint URL format is:

https://<configured site url>/services/apexrest/BenemenPhone/routinginfo

5) The next action is creation of a sharing rule for the Guest Site user.

6) First we need to know the name of autocreated Site Guest user for our site.

a) Go to Setup > User Interface > Sites and Domains > Sites and select the
    appropriate site:

b) Click Public Access Settings
c) Click Assigned Users. You will see the name of the autocreated user (the name
    can be different, it will depend on the name of the site):
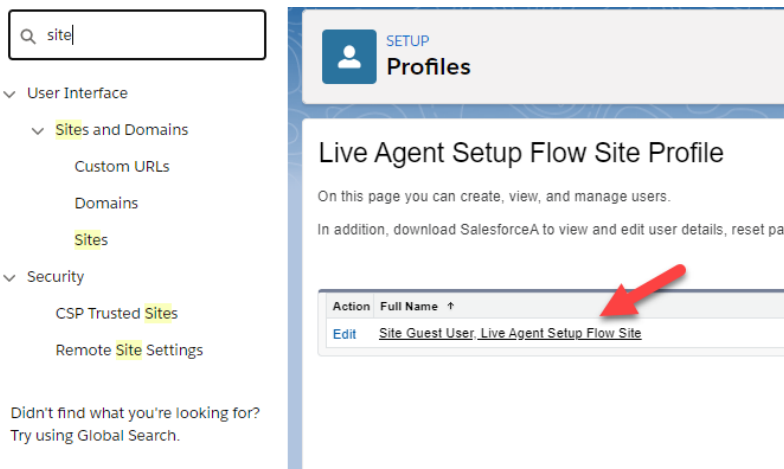
7) Create Sharing rule for sharing Enreach Configuration record.

a) Go to Setup > Security > Sharing Settings.
b) Scroll down to "Enreach Configuration Sharing Rules" under Sharing Rules section
    and click New.
c) Set Label (it can have any name).
d) Select 'Guest user access, based on criteria' option in Step 2: Select your rule
    type section.
e) Set the following criteria in Step 3: Select which records to be shared:
    Configuration Name equals Enreach
f) Select the name of our Site Guest user in Step 4: Select the users to share with.
g) Select Read Only in Step 5: Select the level of access for the users.
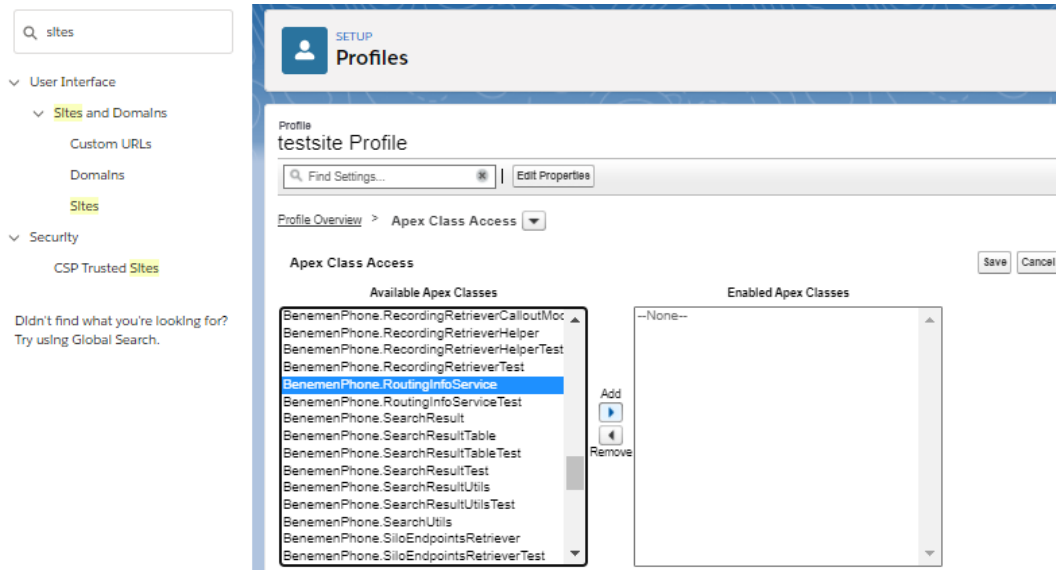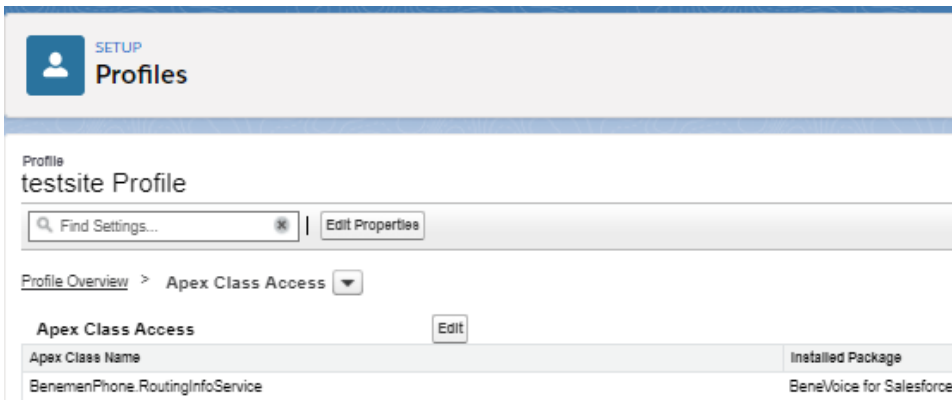
h) Save it.

8) The last action is to add BenemenPhone.RoutingInfoService Apex class to Site Guest User Profile

a) Go to Setup > User Interface > Sites and Domains > Sites and select the appropriate site
b) Click Public Access Settings
c) Click Apex Class Access on the site profile page
d) On the Apex Class Access page, Add and Save BenemenPhone.RoutingInfoService to Enabled Apex Classes

Optional - Configuring Login IP Ranges restriction

It is possible to activate Login IP Ranges restriction for the Public Site guest user. It allows to accept and process requests from the client in the configured IP address range only. Please follow the instructions below to configure it.

a. IP address subnets can be found in Endpoint requirements here;
https://benemen.atlassian.net/wiki/spaces/PD/pages/1215791782/SmartRouting+HTTP+Integrator+Guide#Endpoint-requirements

b. Go to Setup > User Interface > Sites and Domains > Sites and select the appropriate site. Important point: the site should be used for Smart Routing functionality only, to not impact functionality of other sites.

c. Click 'Public Access Settings'.

d. Click 'Login IP Ranges'.

e. Click 'new' button.

f. Set the IP address range received from Enreach, and click 'Save'.

Configure Flows

Managed package includes two Smart Routing flow templates

- SmartRouting_Dispatcher_Template – it is foundation of the future scaling; this flow checks the bnum (called phone number, typically voice queue number) and routes it to Scenario (sub)flow which describes exact routing logic. The default template includes single bnum check and sub(flow), but more rules can be freely added to the dispatcher flow for the other bnum's.
- SmartRouting_Scenario_Template – it is actual (sub)flow template which incorporates logic of routing. This is predefined example, which can be freely used and modified for a specific customer routing scenario. Customer can freely create other scenario flows and invoke them from Smart Routing Dispatcher flow.

The following steps describes configuration steps to use the provided flow templates as baseline for your own smart routing logic.

1) Go to Setup -> Flows and click on the SmartRouting_Scenario_Template flow template name, the flow will be opened.

2) Click 'Save As'
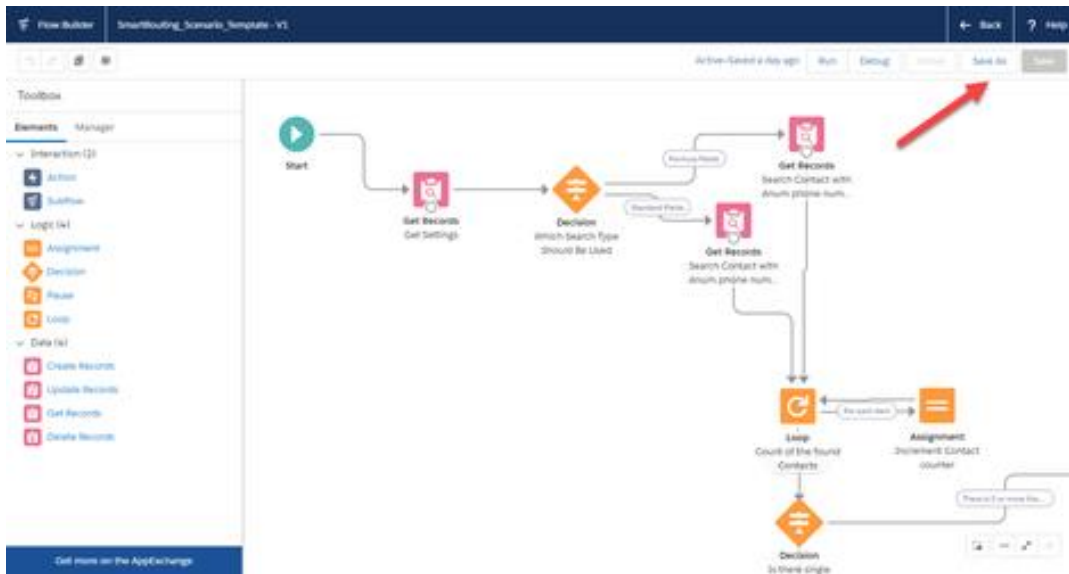
3) Set Flow Label (Flow API Name will be populated automatically) and save it as a new flow. Name e.g. SmartRouting_Scenario1

4) Next open SmartRouting_Dispatcher_Template flow and click 'Save As' to save it a new flow. Name e.g. SmartRouting_Dispatcher

5) Still on newly saved SmartRouting_Dispatcher flow

- Remove original Subflow element, by going over it and click recycle bin icon
- Drag & drop a new Subflow element to the page

6) On the new page select SmartRouting_Scenario1, which you saved in step 3)

7) On the new page

- Components anum and bnum must be set and included under '**Set Input Values'** while setting the subflow.
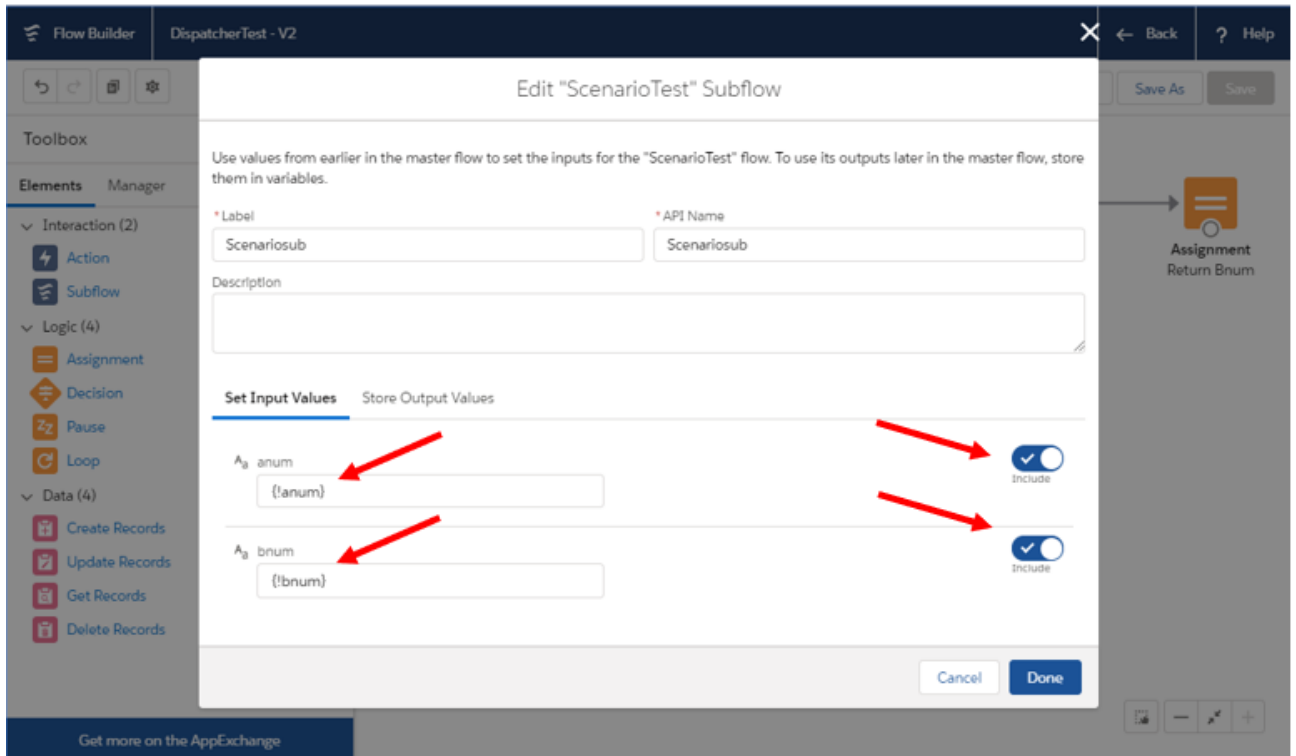
- Check 'Manually assign vabiables (advanced) and the following components must also be set under '**Store Output Values'**; allocationTargets , FailTarget, InitialTarget, Priority and RequiredTarget. (anum and bnum are left empty here)
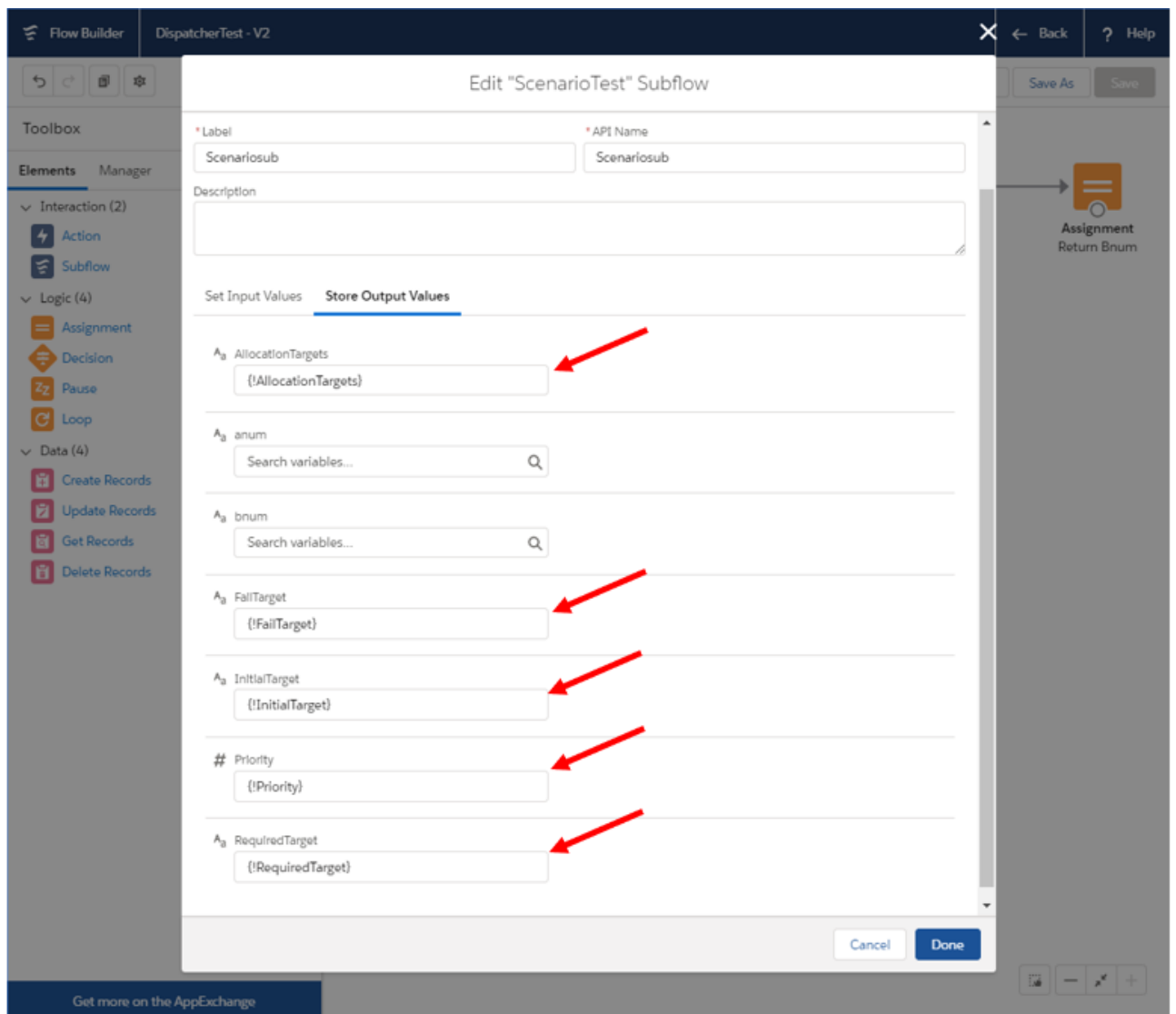
- After that click 'Done'.

8) Connect Decision element to new Subflow element by connector arrow from Decision element

9) Still on SmartRouting_Dispatcher flow, click Manager and then BNUM_CONST_1 constant

- Modify Value by the approriate service pool queue phone number, which you would like to use in your smart routing scenario
- Click Done

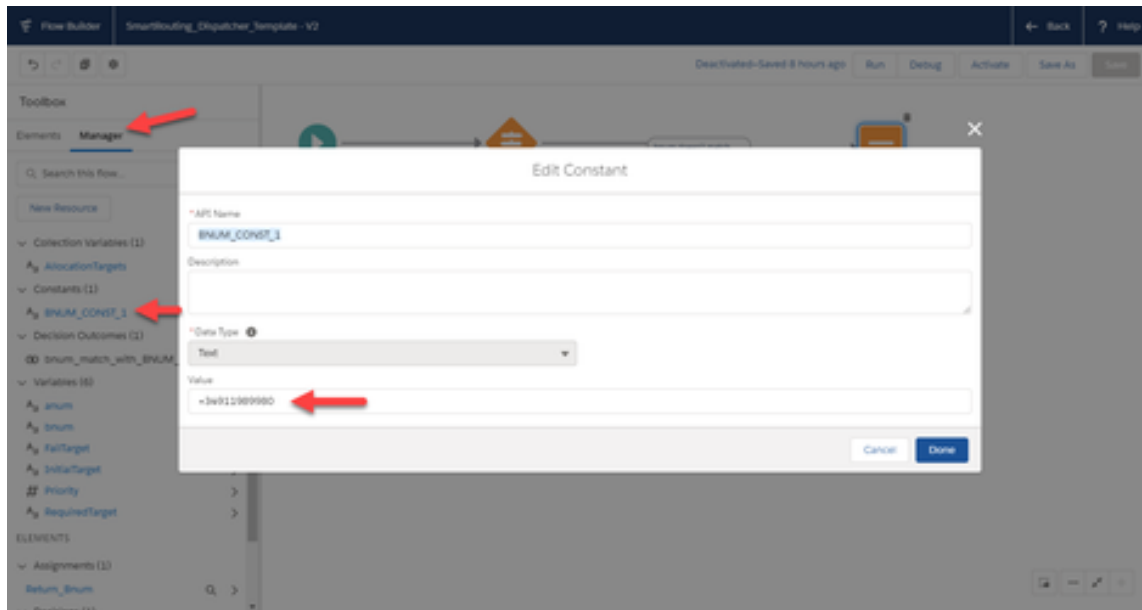9) You have now completed the needed flow configuration to use predefined smart routing logic 'as is' with the service pool queue you defined in BNUM_CONST_1 Value

10) It's almost certain that you need to modify saved SmartRouting_Scenario1 flow with your own business logic and routing targets. Please see Scenario (sub)flow detailed description for more details.

11) When finished, make sure you have activated your dispatcher and scenario flows. You may deactivate the original flow templates installed with the managed package

12) Please proceed to the next chapter to finalize basline configuration on the Salesforce side

## Voice for Salesforce configurator app – Smart Routing activation

Complete the following steps to activate Smart Routing at Salesforce side. Please note that this will not have any affect for actual call routing until also Enreach has completed smart routing activation at Enreach cloud side for the wanted voice queues.

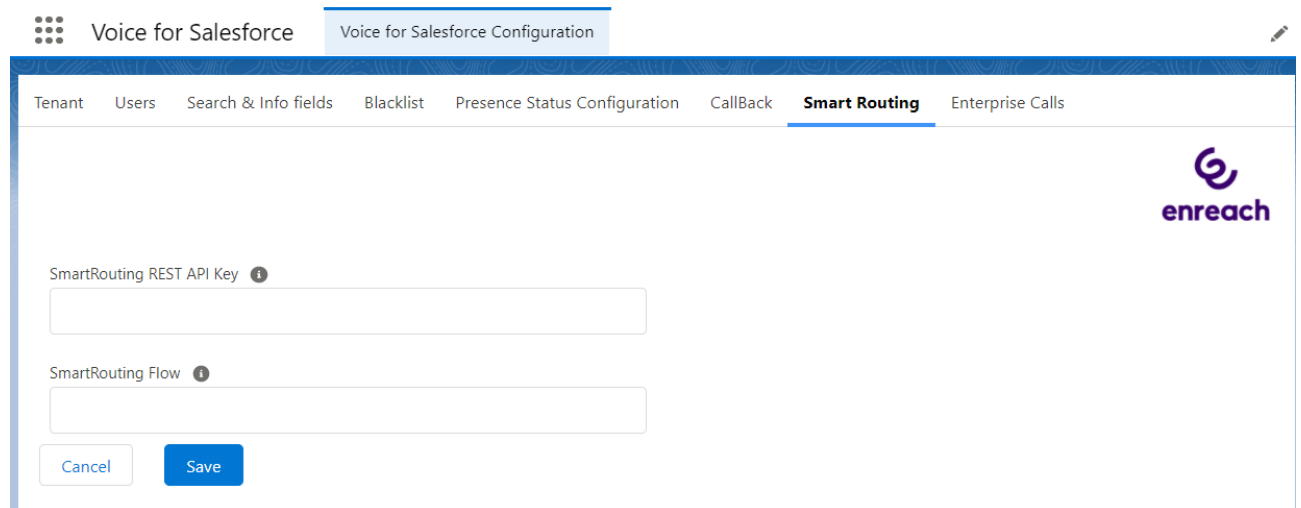1) Go to App Launcher and select 'Voice for Salesforce' app
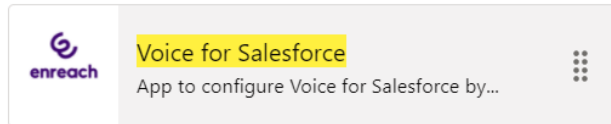
2) Select 'Smart Routing' tab, click 'Edit'

3) SmartRouting REST API Key: Set any API Key and provide this to Enreach. This is used as API key secret for REST endpoint.
4) SmartRouting Flow: Name of the dispatcher flow e.g. SmartRouting_Dispatcher
5) Click 'Save'

Retrieve Queues

Smart Routing configuration page includes Retrieve Queues button.

Using retrieve queues button is optional, but can be useful.

Clicking the button will;

- retrieve all the Enreach cloud side voice queue information
- store the information in Enreach Queue custom object
- display the details on the Smart Routing page.

There are 3 queues in Salesforce. Please find the details below:

BENECLOUD QUEUE NAME	BENECLOUD QUEUE ID	SMART ROUTING PHONE NUMBER	QUEUE PHONE NUMBERS
Platinum queue	ecd7b17f-e482-ea11-b81a-0050569e6df2	+358293092528	+358293092528
Open Case Queue	8f914b51-db61-ea11-b812-0050569edc3b	+358293092529	+358293092529
Customer Service Queue	574e1fc5-da61-ea11-b812-0050569edc3b	+358293092520	+358293092520

Retrieving Enreach voice queue information has two purposes:

- Help admins as they can see the Enreach cloud voice queue details in Salesforce side. E.g. if they need to double check some voice queue phone numbers used in smart routing flows
- As these are stored as Salesforce custom object, admins could even utilize this object and fields directly in smart routing flows instead of phone number text inputs. Note: If using Enreach Queue custom object in smart routing flows, admin must create sharing rule also for the Enreach Queue object. (See chapter 2.2 Step 7).

Enreach cloud – Smart Routing activation

As a final step smart routing shall be activated at Enreach cloud side.

1) Provide configured Public site URL and REST API key values to Enreach
2) Provide voice queue numbers / names, where smart routing should be activated to. In this baseline config BNUM_CONST_1 you configured in the flow setup.
3) Agree on possible additional logic on Enreach cloud side which should be applied
    a. Maximum waiting time for preferred agent
    b. Possible caching timeouts. E.g. if customer calls in again within 5 minutes, should Enreach cloud send smart routing query to Salesforce or use previously used routing logic
    c. Etc.
4) Enreach to activate smart routing on agreed time
5) Once smart routing is activated on Enreach cloud side for the given voice queue, customer may freely modify routing logic via the scenario flow
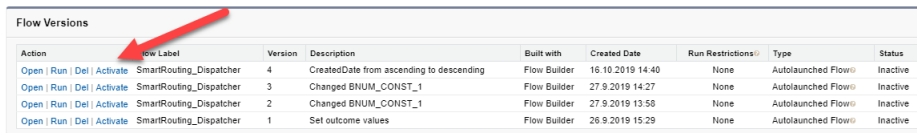
Detailed information about predefined flows

Flows are used for relatively simple configuration of business processes, in our case - routing scenarios related to a customer caller number and called voice queue number.

We can find flows here: Setup > Process Automation > Flows. Each flow can have many versions, but only one version can be active, so this version will be invoked when the flow is started.

If you want to activate a flow you can go to Setup > Process Automation > Flows, click arrow on the right side of the flow and select 'View Details and Versions'

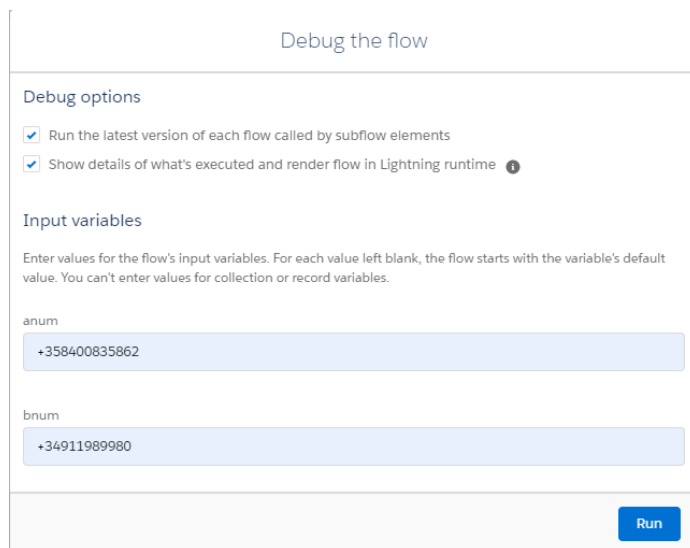After that you can click 'Activate' on the appropriate version:

If you want to change logic in a flow, you need to click 'Open' under actions.

The flow builder will be opened (the version which is currently activate will be opened), you can do your changes (for example change phone number of the target queue), after that you can click 'Save As' button and save it as a new version, you can set some description for this version. After that you can activate the new version.

While the flow builder is open you can easily debug your flow by clicking 'Debug' in top right corner.

This will open debugger where you can test the current flow by providing wanted caller phone number (anum) and called voice queue number (bnum). Click then 'Run' and you will see the results.
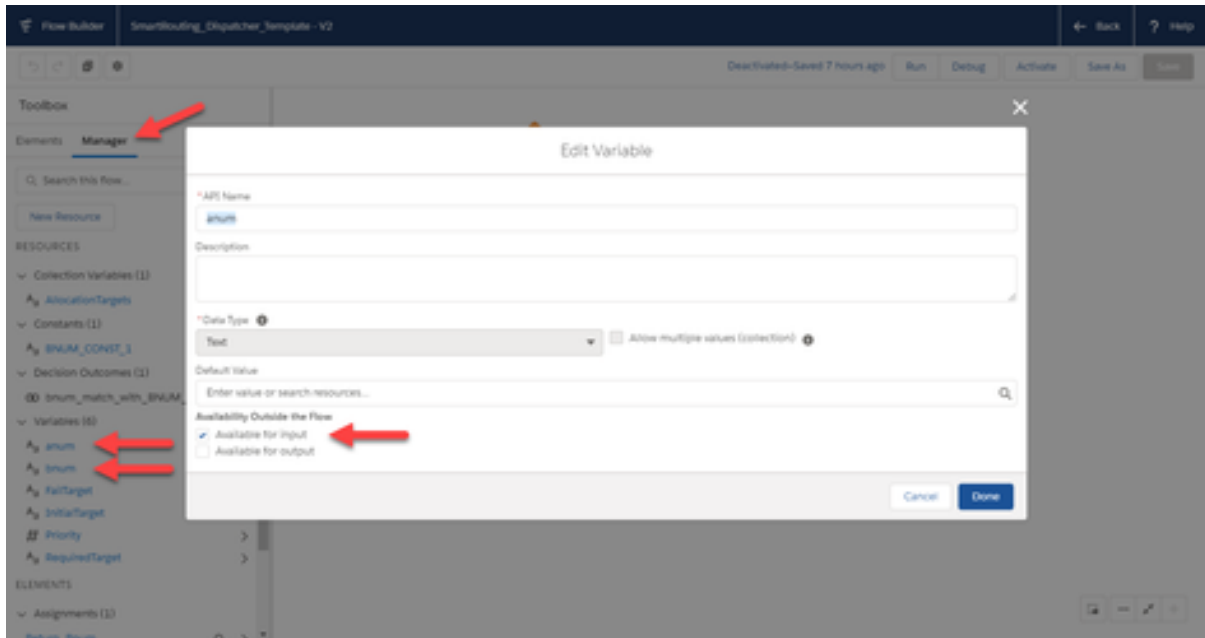

Dispatcher Flow detailed description

The flow has the following **variables available for input**:

1. **anum** - text variable which receives anum. This is a caller number, typically end customer's phone number. This is used for identifying caller and making routing decision based on the described logic in the scenario flow.
2. **bnum** - text variable which receives bnum. This is a called number, typically voice queue phone number which end customer called. This is used to find if dispatcher flow has smart routing rule for this bnum. Bnum can also be used as default outcome i.e. when we can't find any smart routing logic, we return bnum back
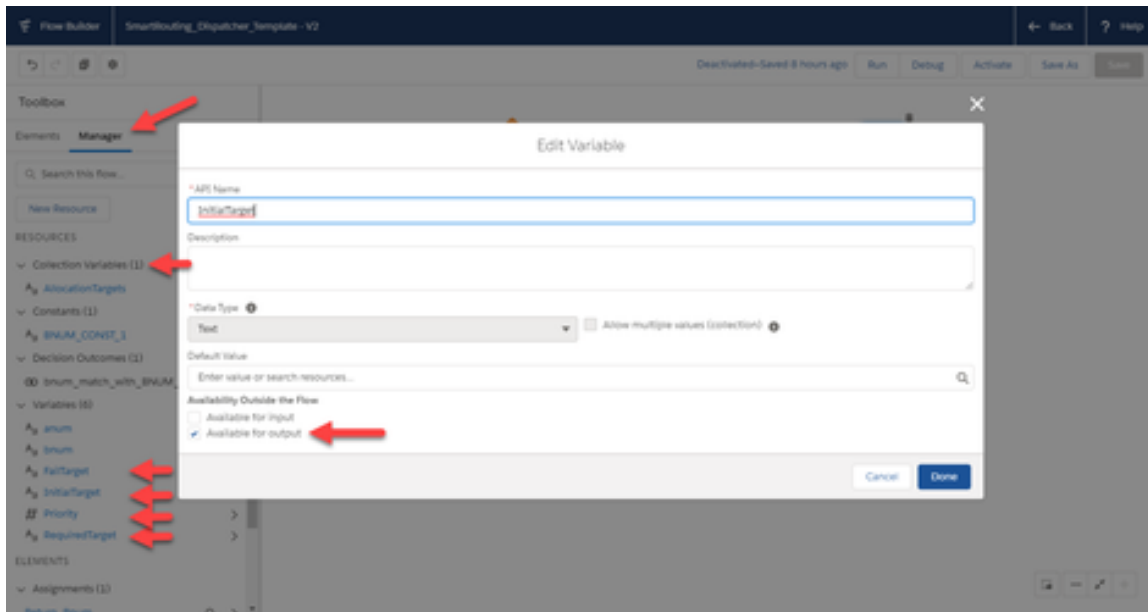
and phone call is routed by standard routing logic. Note: empty response is normally recommended option, when smart routing logic is not found.

The flow has the following **variables available for output**:

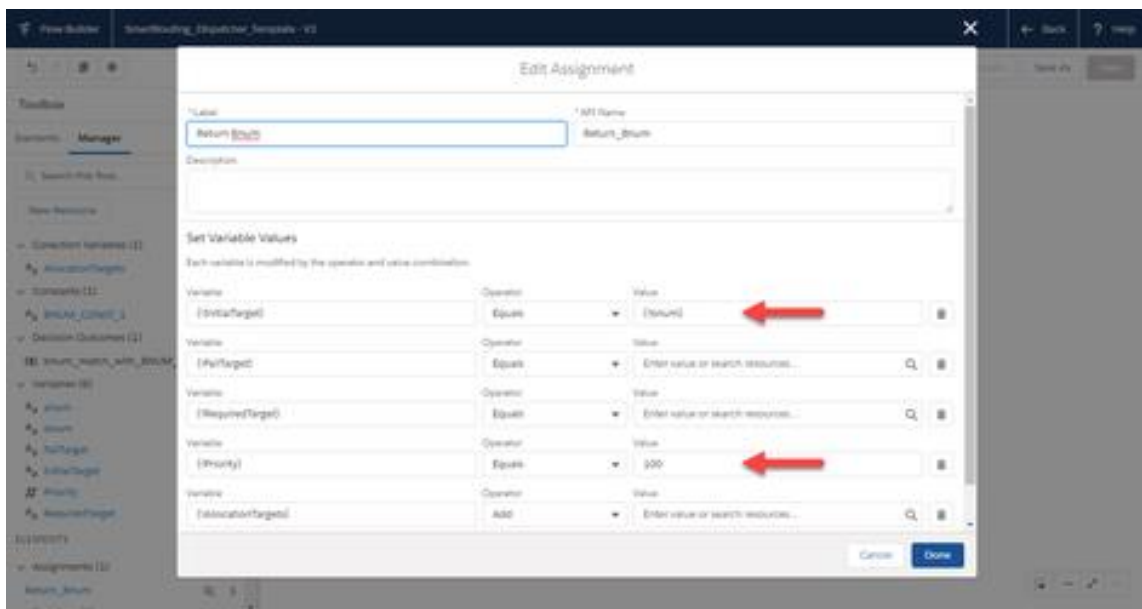| Key	Type	Values	Description	Mandatory
InitialTarget	string	o Any number (external) o Queue number o Queue GUID o Agent number o Agent email-ID o Agent GUID	Primary target. If target is Queue, preferred agents can be specified with RequiredTarget and AllocationTargets	No
FailTarget	string	o Any number (external) o Queue number o Queue GUID o Agent number o Agent email-ID o Agent GUID	Secondary target. Target if InitalTarget fails.	No
RequiredTarget	string	o Agent email-ID o Agent number o Agent GUID	Primary allocation target (agent). If agent is busy, call will be waited specified time, default (20 secs)	No
AllocationTargets	string	o Agent email-ID o Agent number o Agent GUID	List of preferred allocation targets, if primary allocation target fails or is not specified.	No
Priority	int		Priority for this call, makes call x seconds "older". Calls in queue are ordered based on how long they have been waiting, oldest call is allocated first.	No

There should be **at least 1 constant which keeps phone number** of the bnum which should have smart routing scenario flow, but there can be several constants with different bnum's:

- BNUM_CONST_1 - text constant, contains phone number which will be compared with bnum.
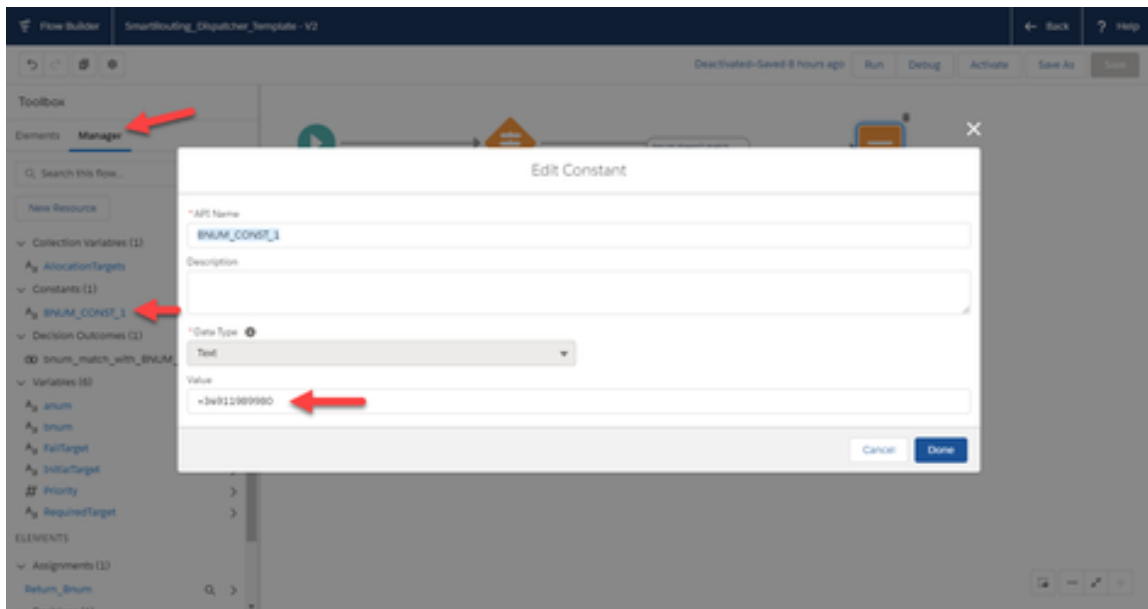
The flow should contain **default assignment** element which initialize 5 output variables. It fills InitialTarget with bnum and set Priority = 100, the rest 3 variables are leaved blank. The default assignment is used, when smart routing rule can't be found, and we just return original bnum to route the phone call by standard routing logic.
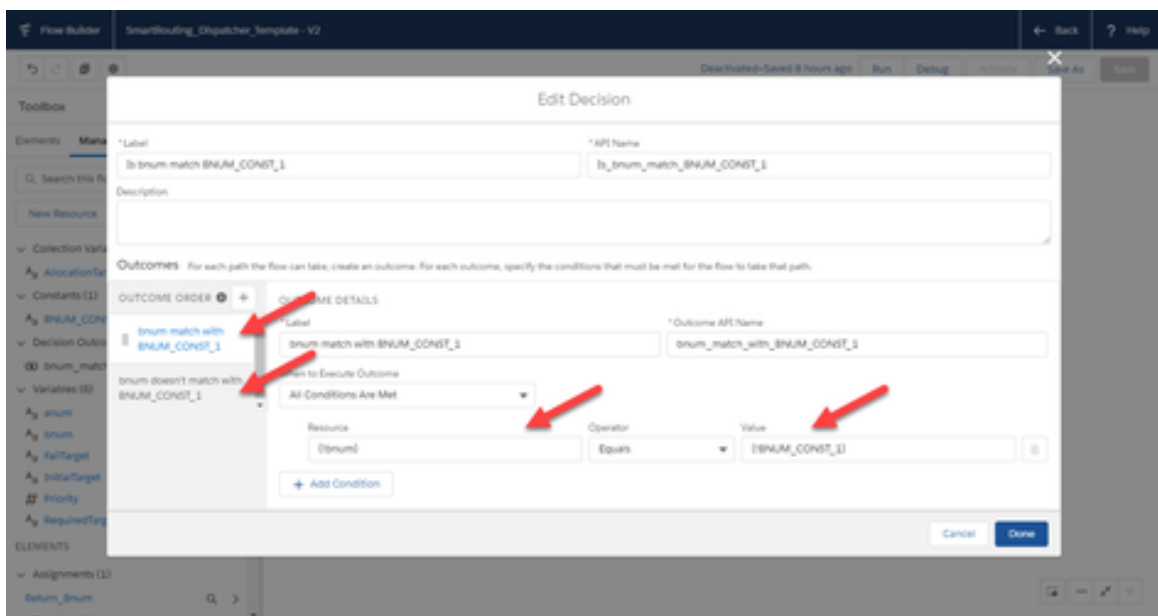
The flow should contain at least **1 decision block which checks if inbound bnum have scenario flow**. If there are more constants with other bnum's, there should be decision for each of them:
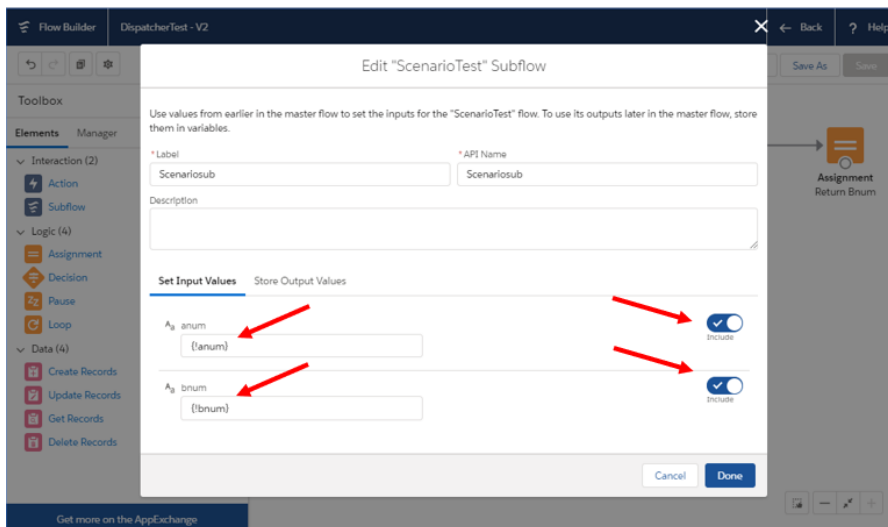
The decision have **two outcomes**:

1. The first contains condition
2. The second is just default outcome (without condition)

Finally the flow should have **Scenario** (**sub**)**flow invocation**. Subflow element can be drag and dropped from 'Elements' tab to flow area, the appropriate subflow is selected from the drop-down:

Components anum and bnum must be set and included under '**Set Input Values'** while setting the subflow.

The following components must also be set under '**Store Output Values'**; allocationTargets , FailTarget, InitialTarget, Priority and RequiredTarget. (anum and bnum are left empty here)
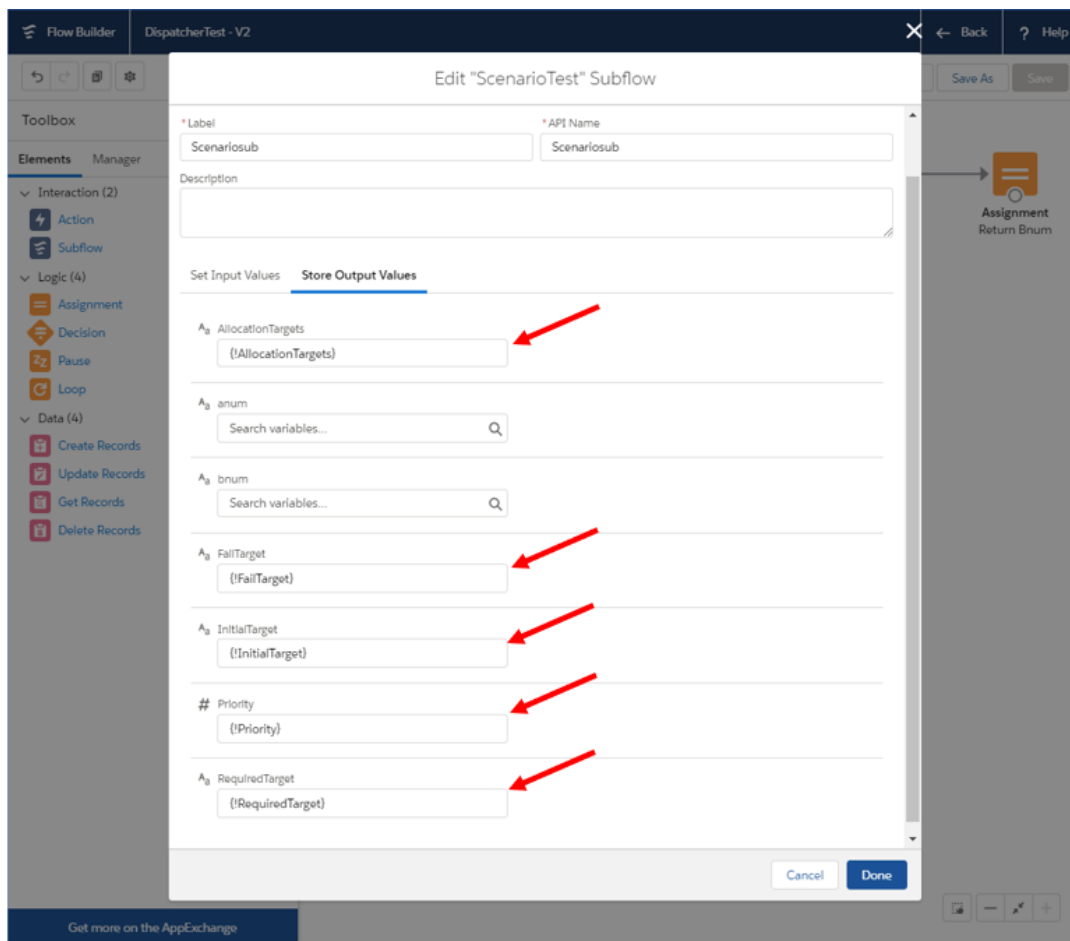
Scenario (sub)flow detailed description

The flow where actual business logic for routing decision is made.

The same input and outcome variables are used as with dispatcher flow.

Input variables:

1. **anum** - text variable which receives anum. This is a caller number, typically end customer's phone number. This is used for identifying caller and making routing decision based on the described logic in the scenario flow.
2. **bnum** - text variable which receives bnum. This is a called number, typically voice queue phone number which end customer called. This is used to find if dispatcher flow has smart routing rule for this bnum. Bnum can also be used as default outcome i.e. when we can't find any smart routing logic, we return bnum back

Outcome variables (actual routing target(s)):

Key	Type	Values	Description	Mandatory
InitialTarget	string	o Any number (external) o Queue number o Queue GUID o Agent number o Agent email-ID o Agent GUID	Primary target. If target is Queue, preferred agents can be specified with RequiredTarget and AllocationTargets	No
FailTarget	string	o Any number (external) o Queue number o Queue GUID o Agent number o Agent email-ID o Agent GUID	Secondary target. Target if InitalTarget fails.	No
RequiredTarget	string	o Agent email-ID o Agent number o Agent GUID	Primary allocation target (agent). If agent is busy, call will be waited specified time, default (20 secs)	No
AllocationTargets	string	o Agent email-ID o Agent number o Agent GUID	List of preferred allocation targets, if primary allocation target fails or is not specified.	No
Priority	int		Priority for this call, makes call x seconds "older". Calls in queue are ordered based on how long they have been waiting, oldest call is allocated first.	No

Customer may build any wanted business logic, but predefined scenario flow includes the following logic:

1) Search Contact database based on caller's phone number (anum)
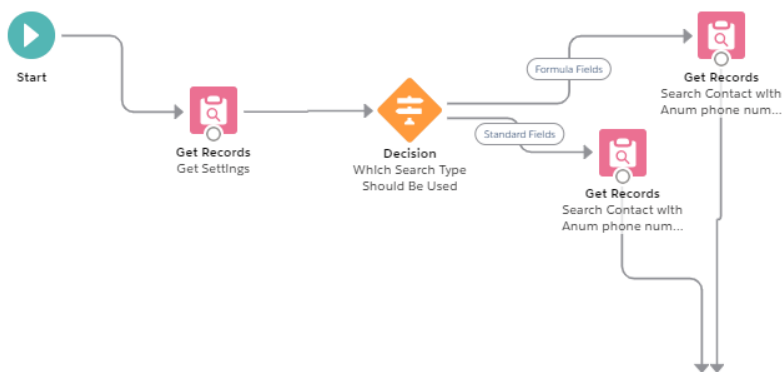
2) If single identified Contact is found, check if the contact has open Cases
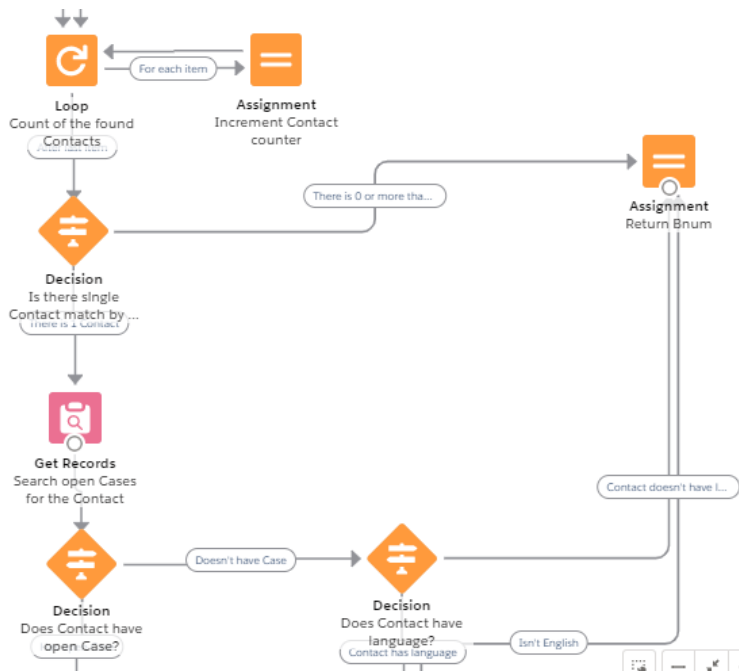
3) If yes, check who is the owner of the newest case , route the call directly to the owner based on the owner's phone number in User.Phone field

4) If no, check if the contact has preferred language identified and if it is one of the determined languages. Note: the predefined flow has Contact.Description field configured as field containing the language. Customer may naturally change this field to appropriate one. When doing so you need change the field in the following 6 resources:
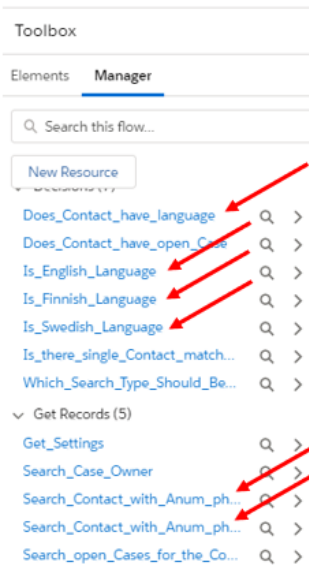
5) If yes, route the call to given language voice queue based on hard coded voice queue phone number / guid
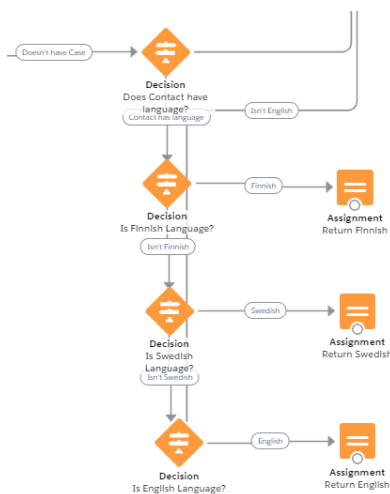
6) In general, if conditions are not met the flow sends back original bnum and the call would be routed based on standard routing logic