Genesys Cloud

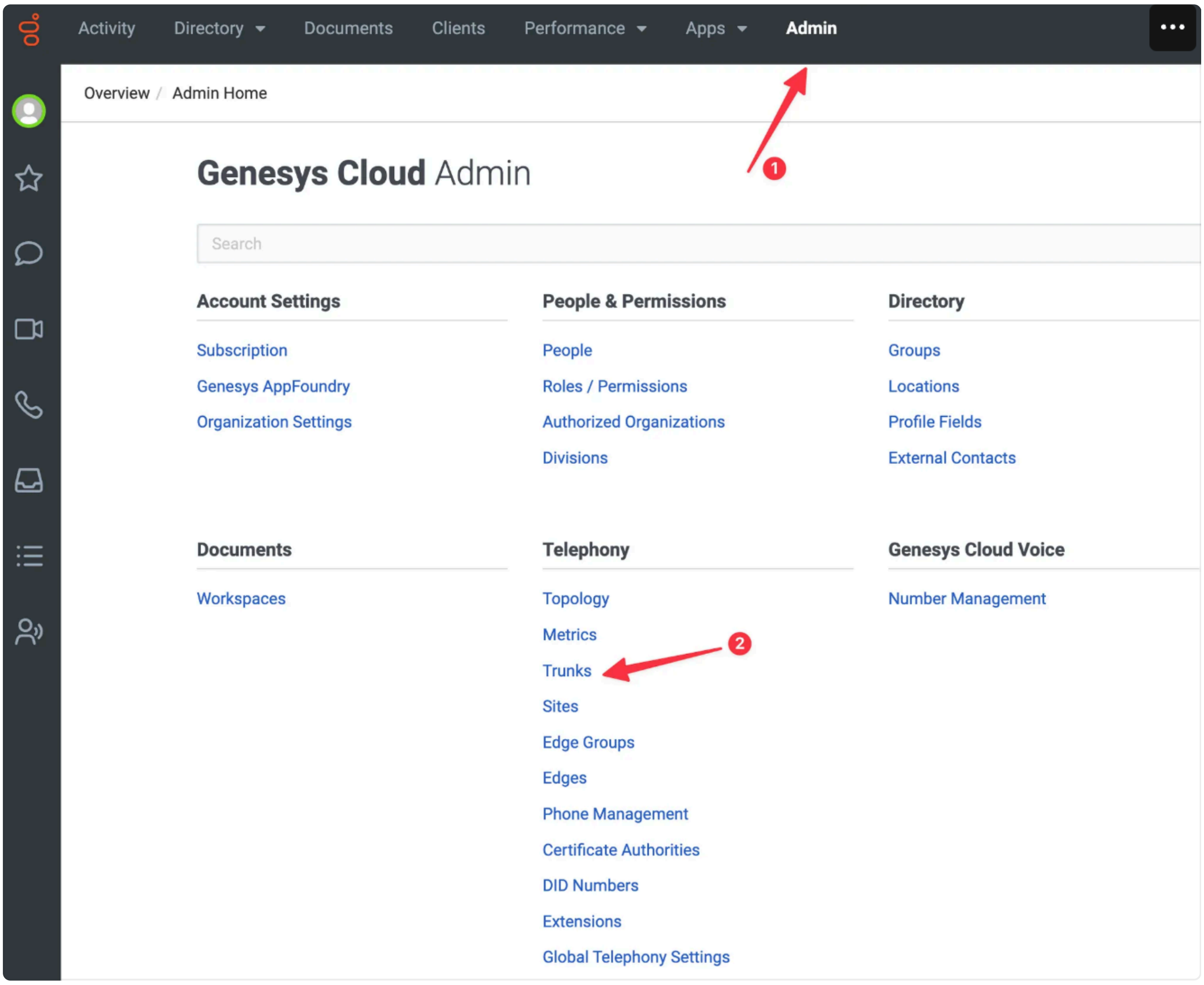
The how to guide on setting up the Genesys Cloud integration with RTT

Pre-requisites

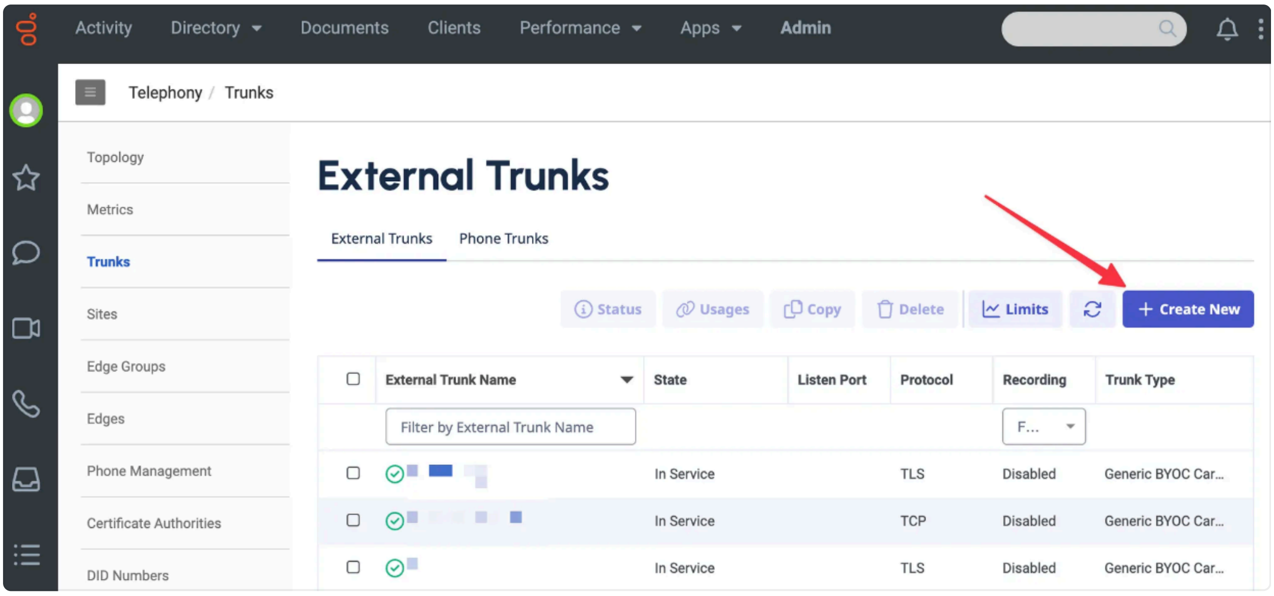
- **Genesys Cloud** account with Admin rights
- **Parloa** account with Admin rights on your tenant

Configuring the SIP Trunk

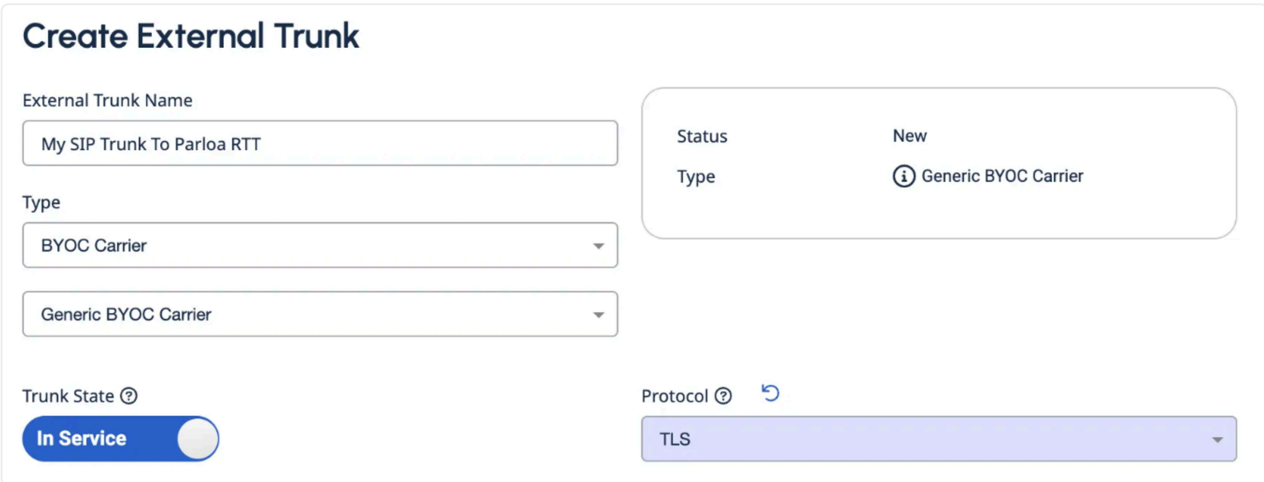
1. Start by navigating to Genesys Cloud's Admin panel, then Telephony → Trunks.



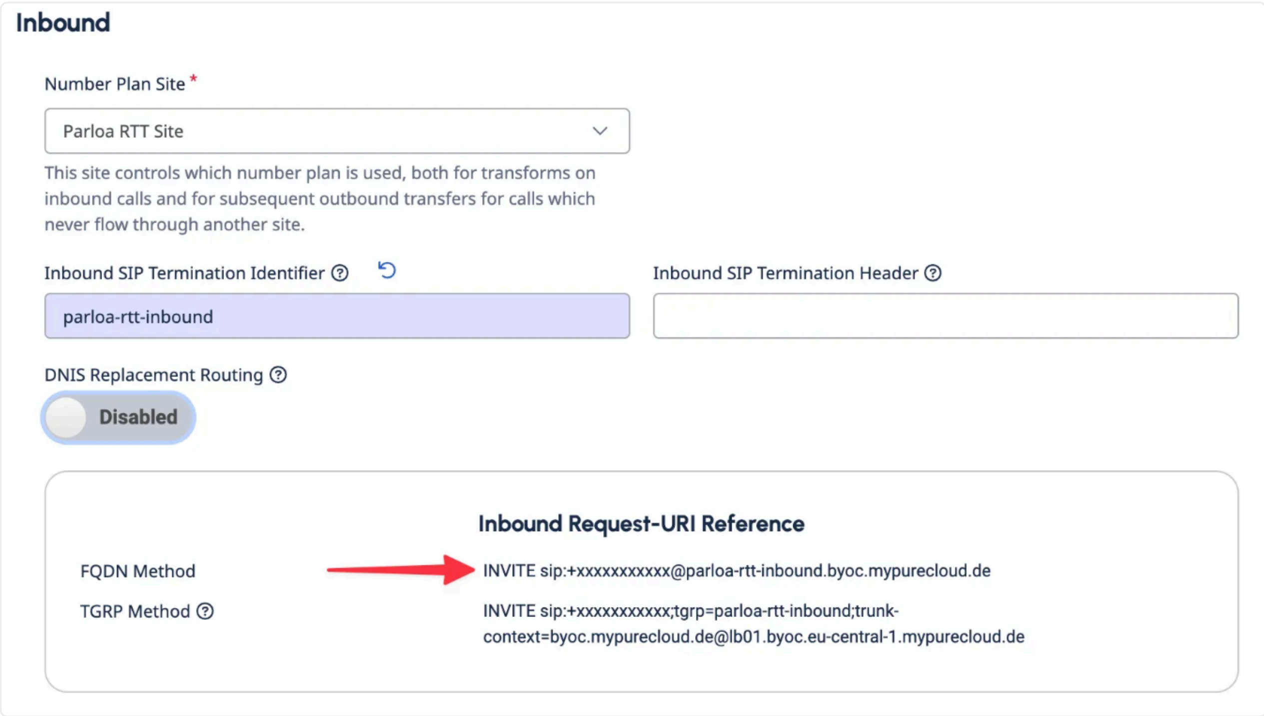
2. Click Create New to create a new SIP Trunk



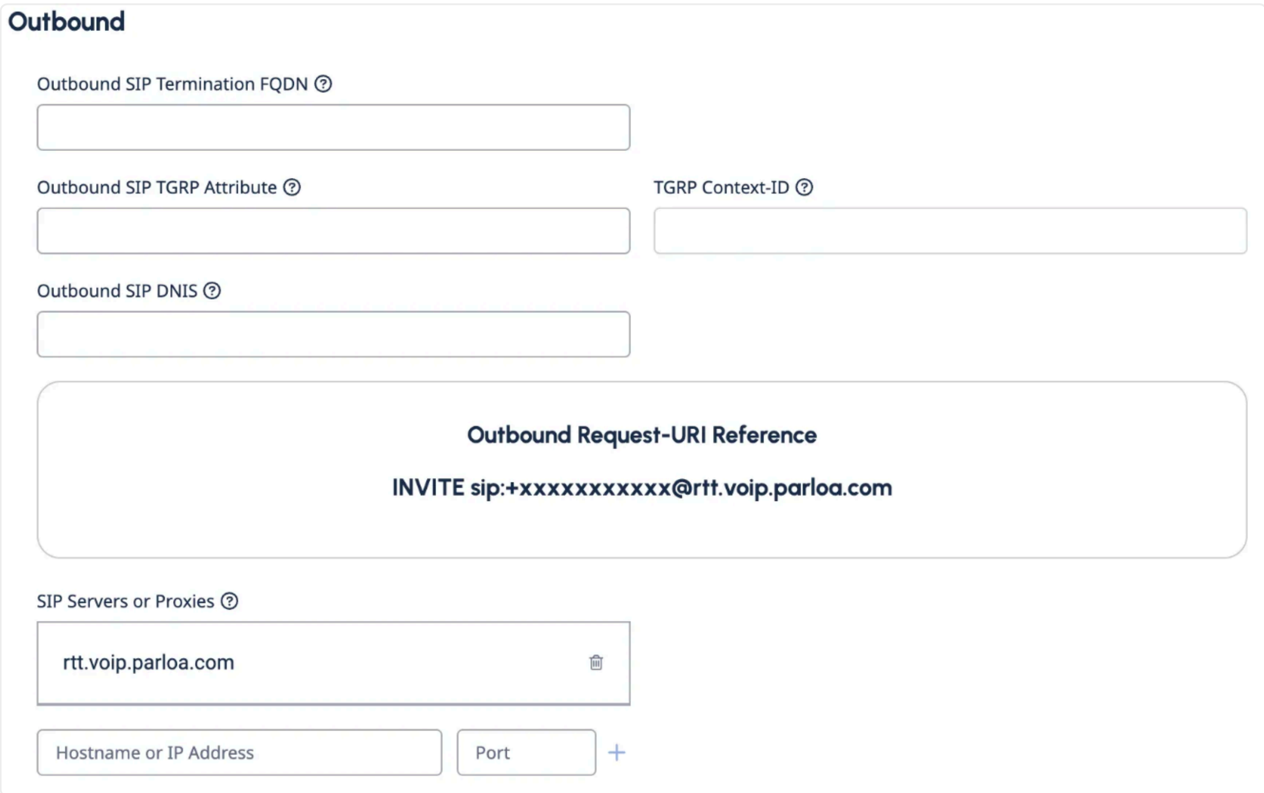
3. Give it a descriptive name, and under **Type** select **BYOC Carrier**, and **Generic BYOC Carrier**. Under **Protocol**, we strongly suggest **TLS**. Lastly, make sure the **Trunk State** is **In Service**.



4. Under **Inbound**, select your preferred Site, and type your desired **Inbound SIP Termination Identifier**. This Identifier will be part of the FQDN that will be used for inbound calls into Genesys, so keep that in mind.

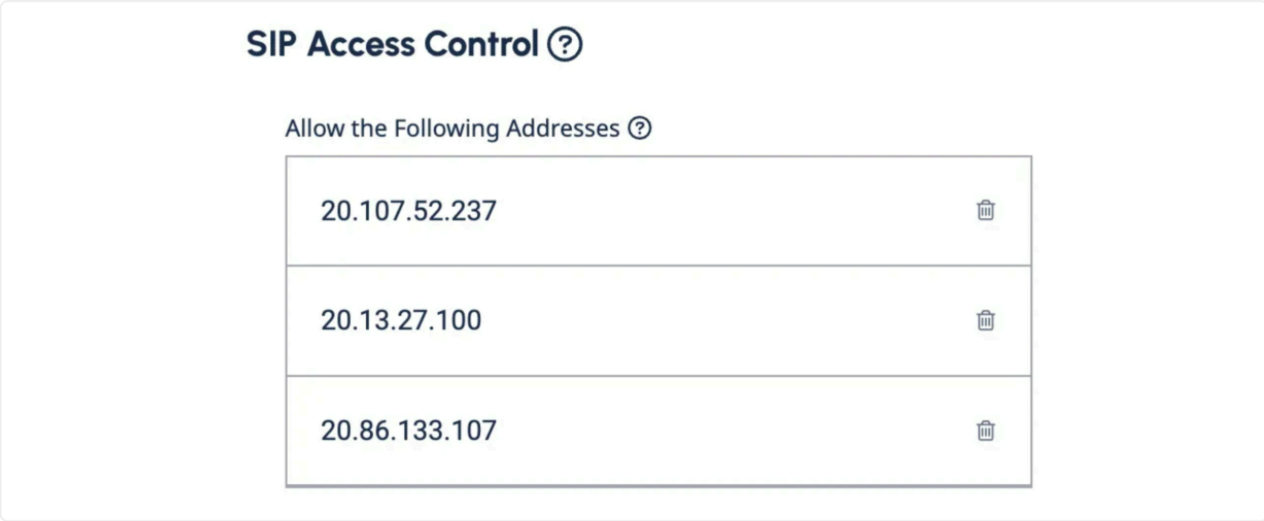


5. Under Outbound, the only thing you will need to add is a **SIP Server or Proxy**. For RTT, we always use `rtt.voip.parloa.com`, and there is no need to specify the port. The rest can be left blank.

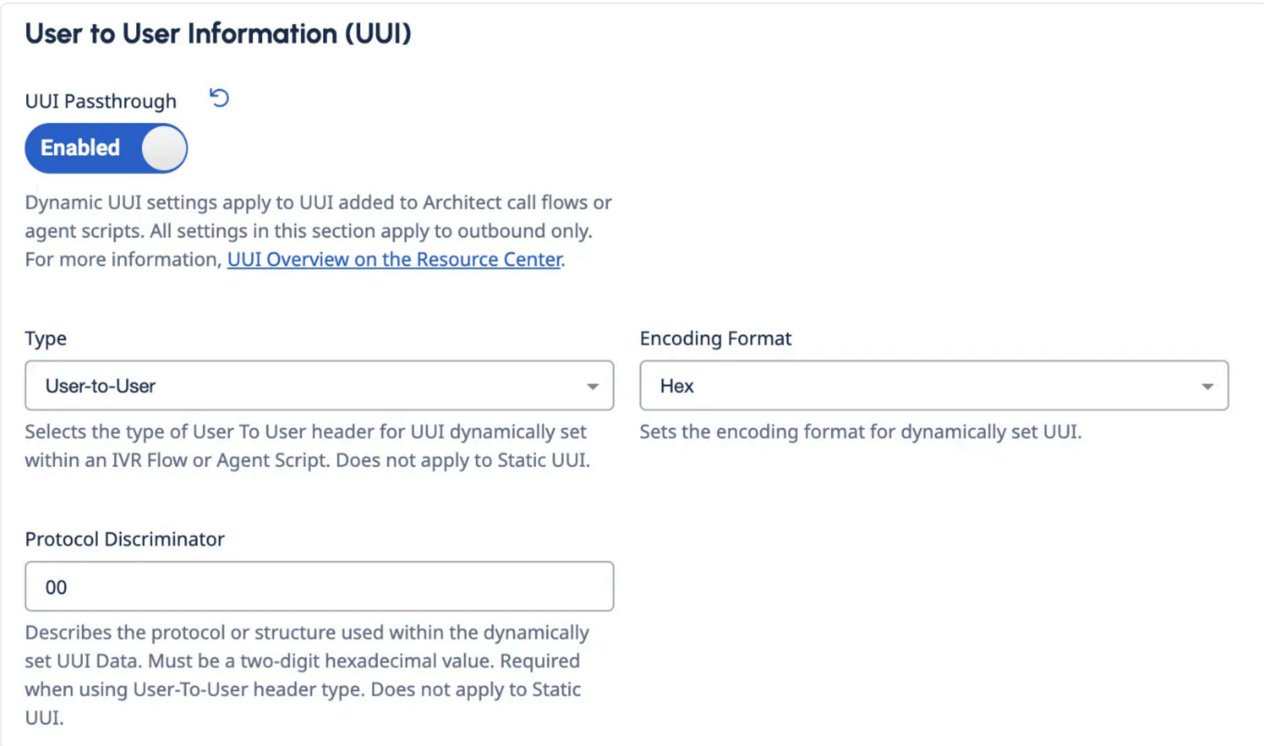


6. Scroll down to **SIP Access Control**. Here you have to whitelist Parloa's Signalling IPs. You can find these in our [Public Documentation](). Make sure you

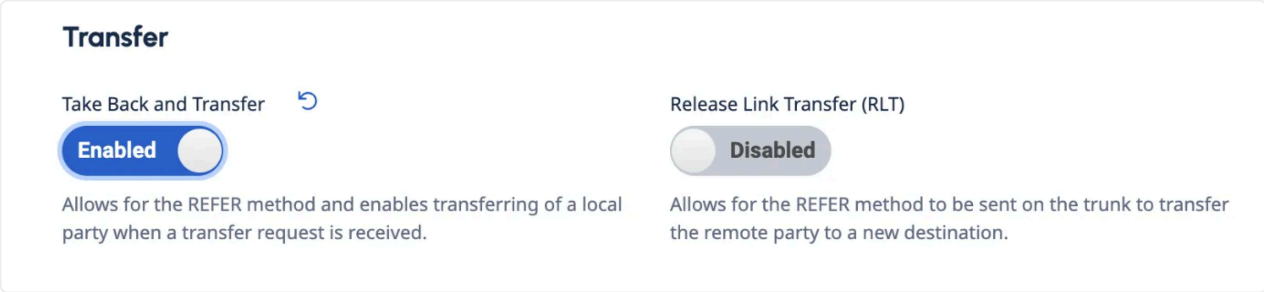
add the ones tagged as **VoIP Traffic**.



7. Scroll down and expand the **Protocol** section. Under **User to User Information (UUI)**, we suggest you enable **UUI Passthrough**.



8. Still under **Protocol**, enable **Take Back** and **Transfer**.



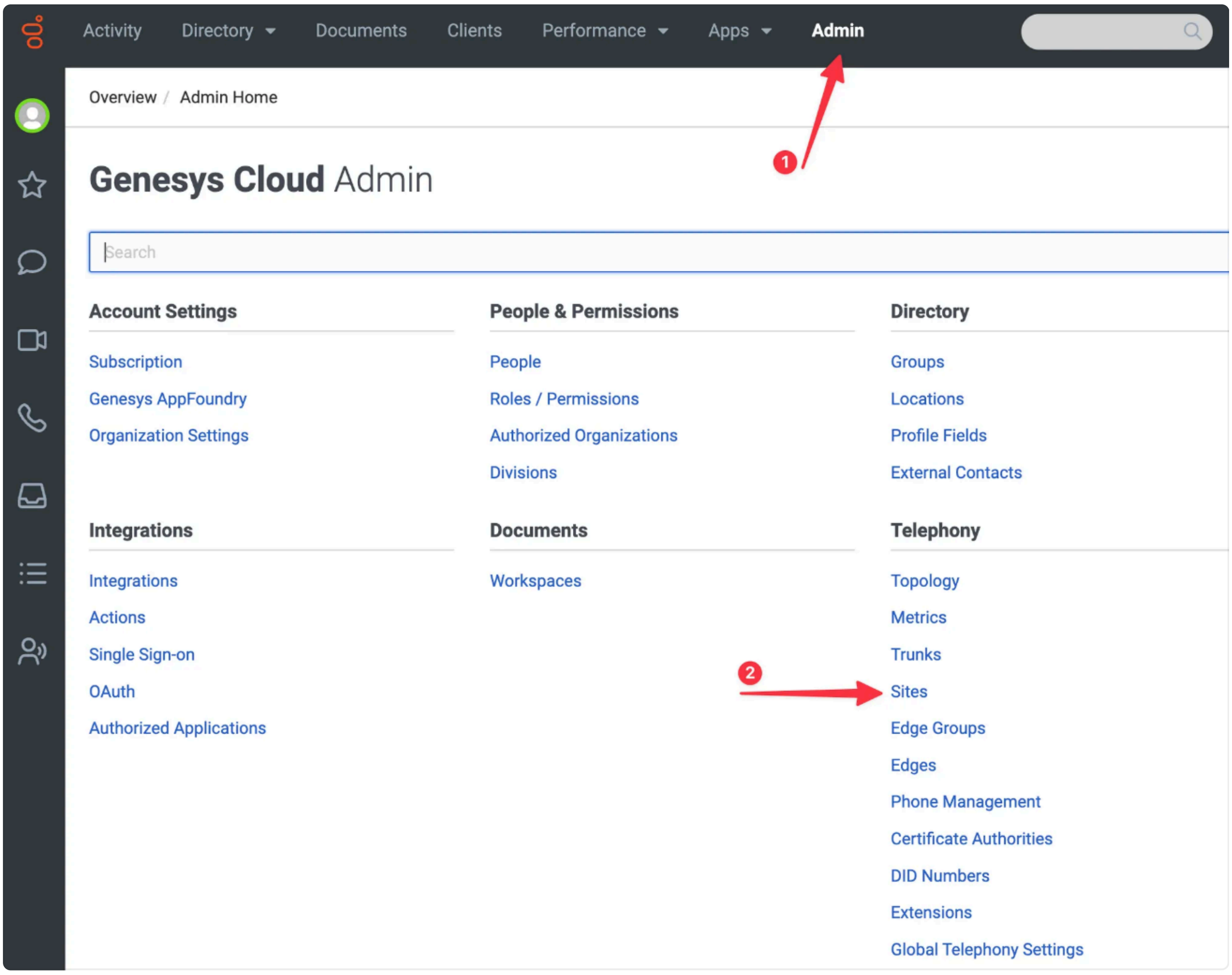
9. Click Save External Trunk.

This settings should be enough to allow calls from Genesys to Parloa, and vice versa.

Numbering Plan & Outbound Routes

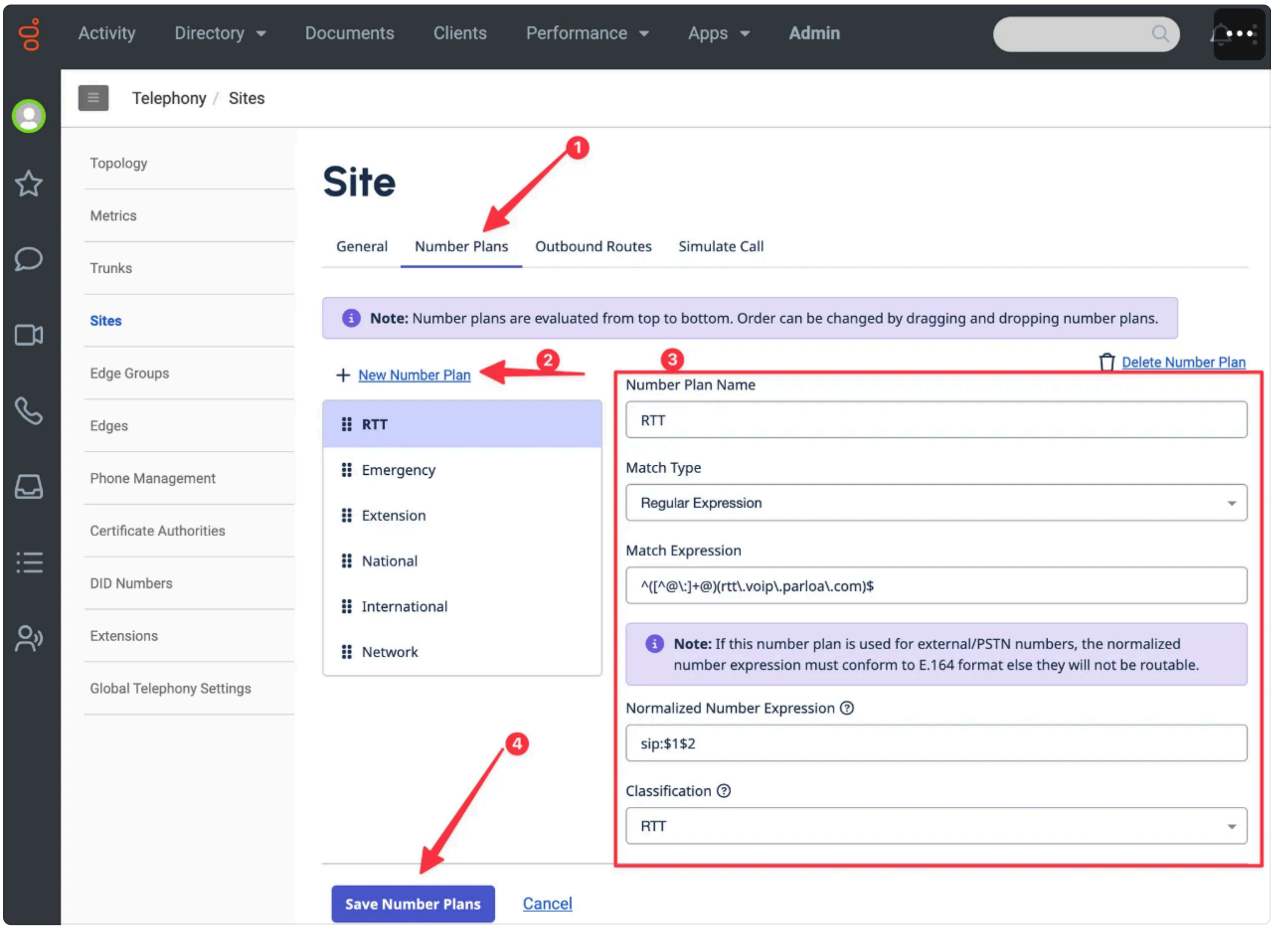
For every caller language in RTT, you will have a unique URI you need to call from Genesys Cloud. In this section we'll configure Genesys Cloud to route these URIs to the newly created SIP trunk.

1. Navigate to the Admin panel, then **Telephony → Sites**

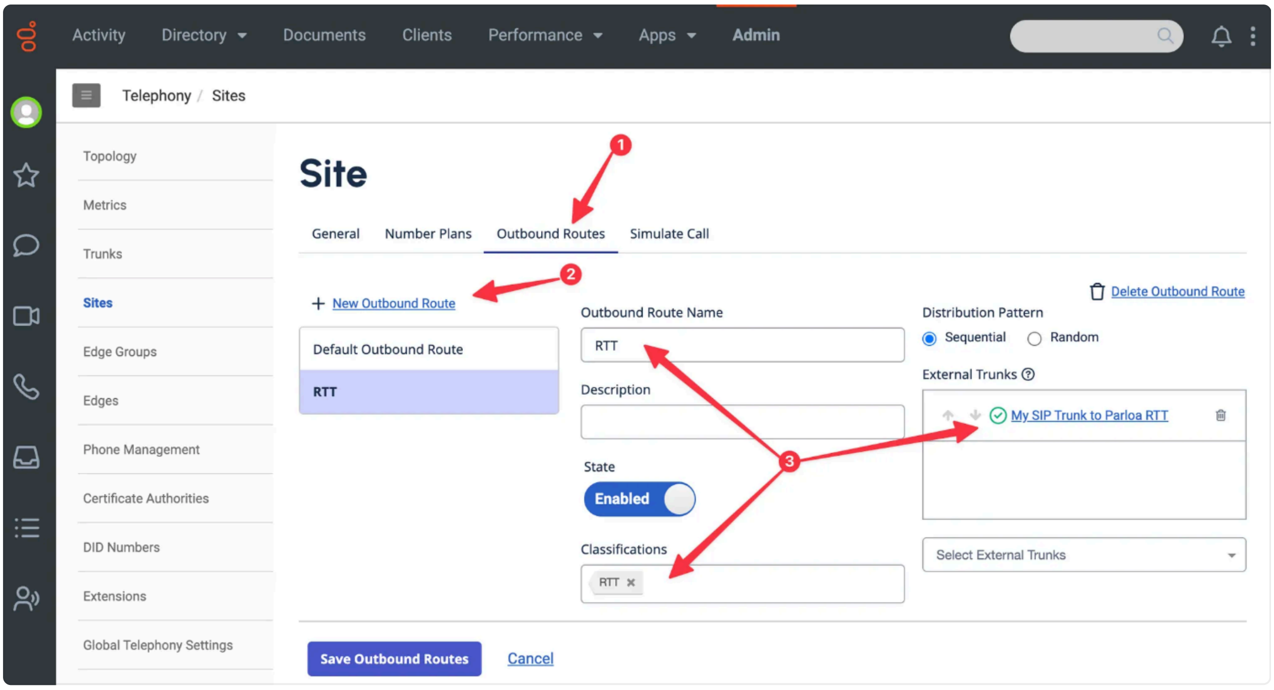


2. After clicking on your Site (this has to be the same Site the trunk was set up on step #4 of Configuring SIP Trunk), navigate to **Number Plans**, create a New Number Plan for RTT URIs and do the following:

- **Number Plan Name & Classification**: Can be whatever you want; in this case we name both `RTT`.
- **Match Type**: **Regular Expression**
- **Match Expression**: `^([^@\\:]+@)(rtt\\.voip\\.parloa\\.com)$`
- **Normalized Number Expression**: `sip:$1$2`



3. After Clicking **Save Number Plans**, navigate to **Outbound Routes**. Here is where we decide which trunk we want to send the calls that match the RTT regex. Click **New Outbound Route**. Give it a name, select the RTT classification you created in the previous step, and select the external trunk you created it the first section of this guide.



We've now told Genesys how to identify RTT URIs, and which trunk should be used whenever we need to call said URIs.

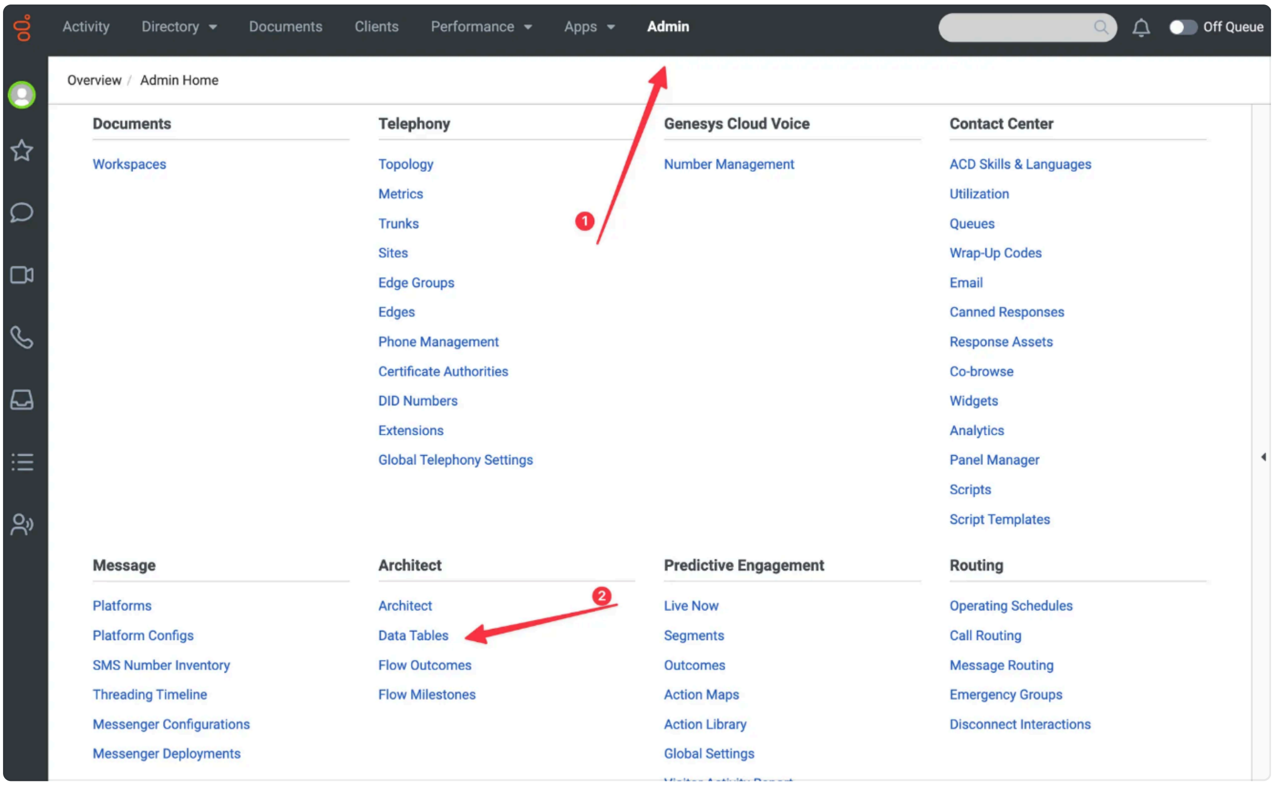
Forwarding calls to RTT

In this section, we'll explain how to forward calls from Genesys to specific languages in RTT. There are of course may ways you can do this, but if you have no idea where to begin, this is a good starting point. This may or may not apply to your setup, but we recommend reviewing it briefly. In this scenario, we have to different hotlines come into Genesys, and they will be forwarded to different languages using Data Tables in the Architect.

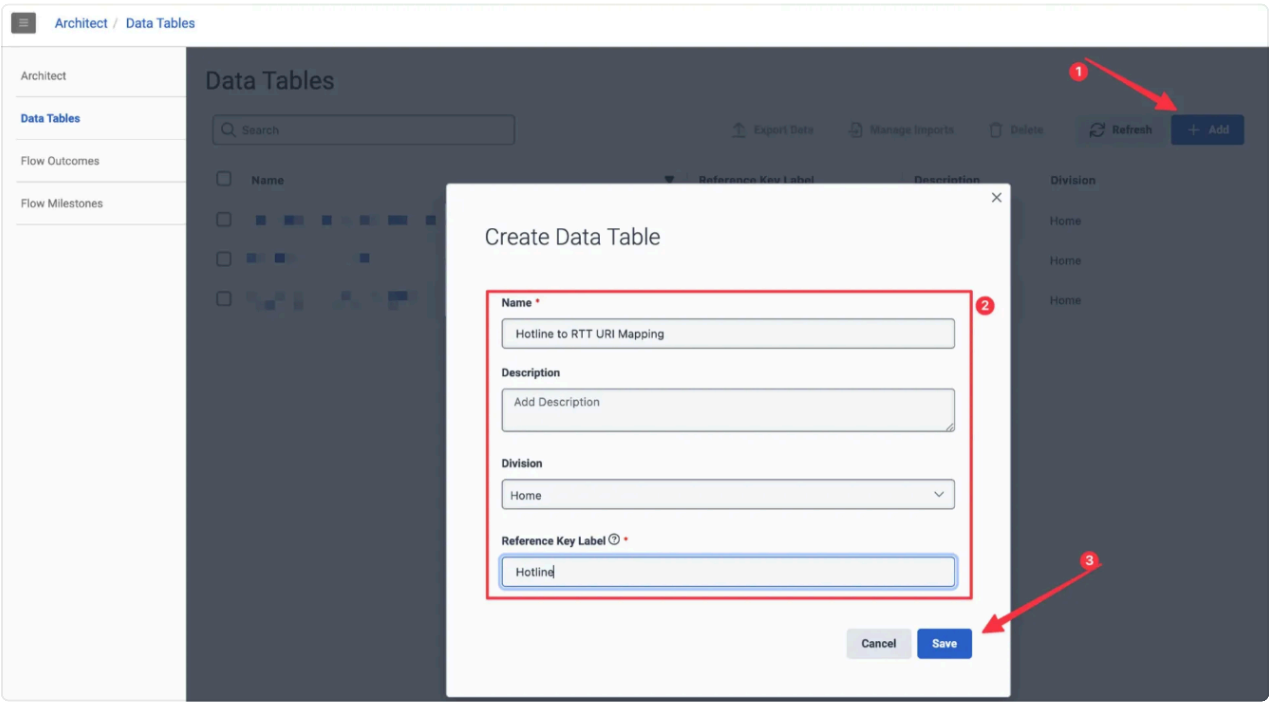
Data Tables

We will use Data Tables to map incoming calls to different hotlines, to different languages in RTT.

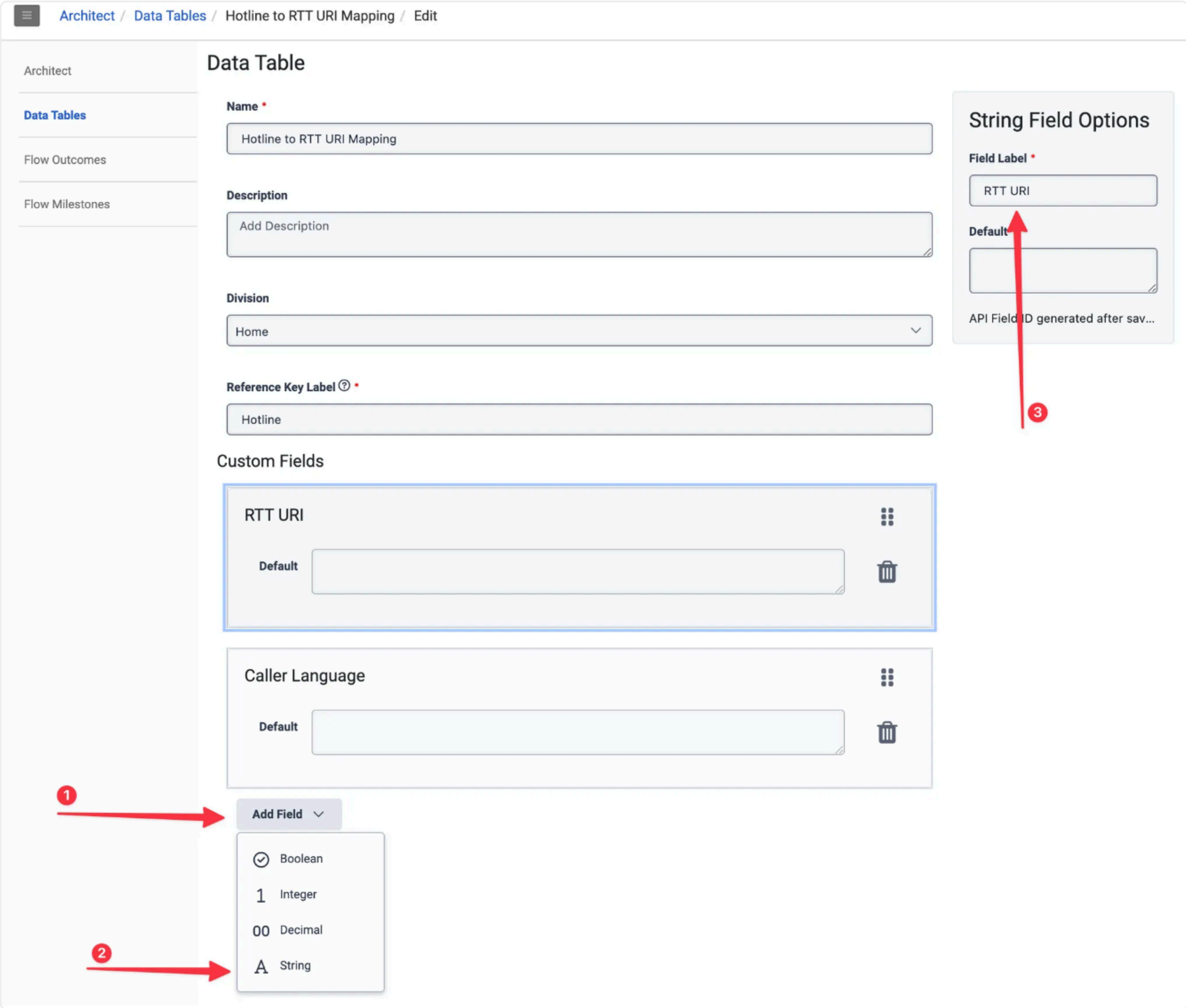
1. The first thing we need to do is to create a Data Table. Navigate to the Genesys Admin panel, then under Architect → Data Tables.



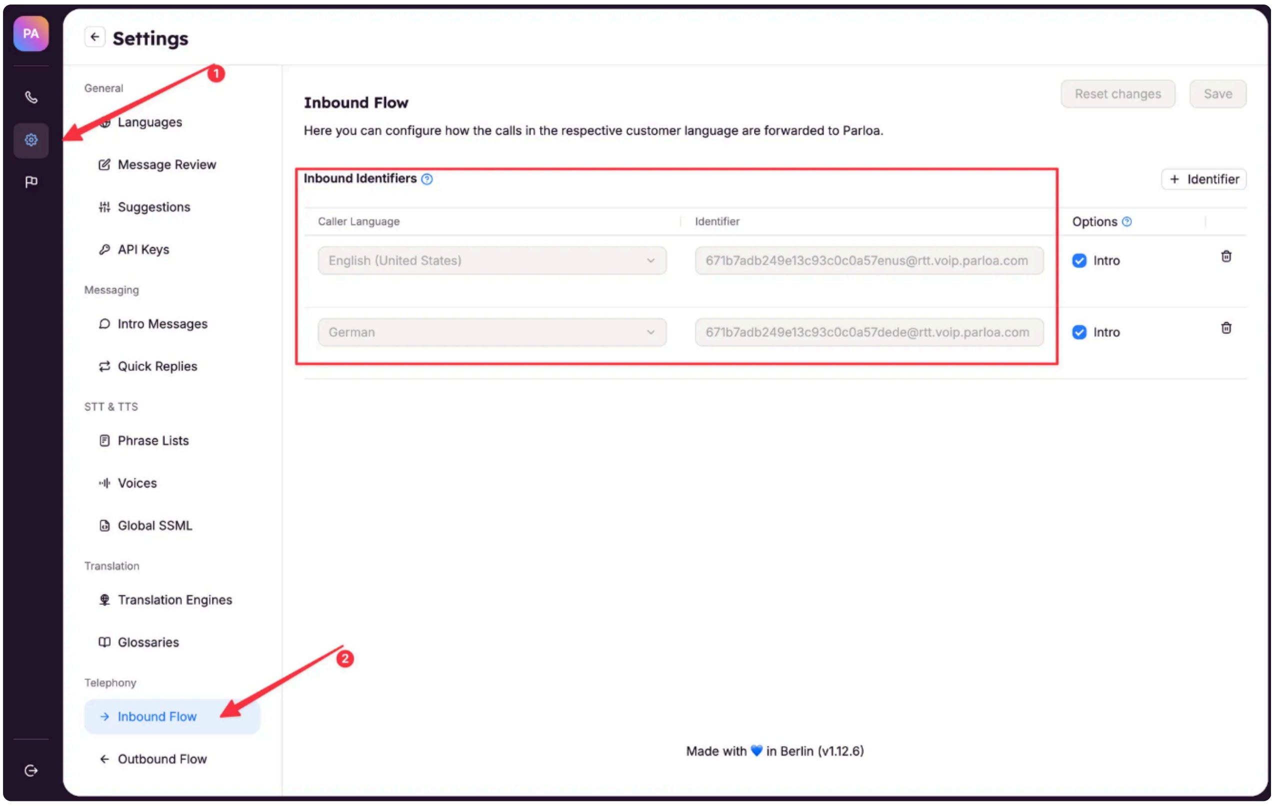
2. Click **+ Add** to create a new table. Give it a name, and a **Reference Key Label**, which in our case will be the Hotline number (the number that the caller dialed). Click **Save**.



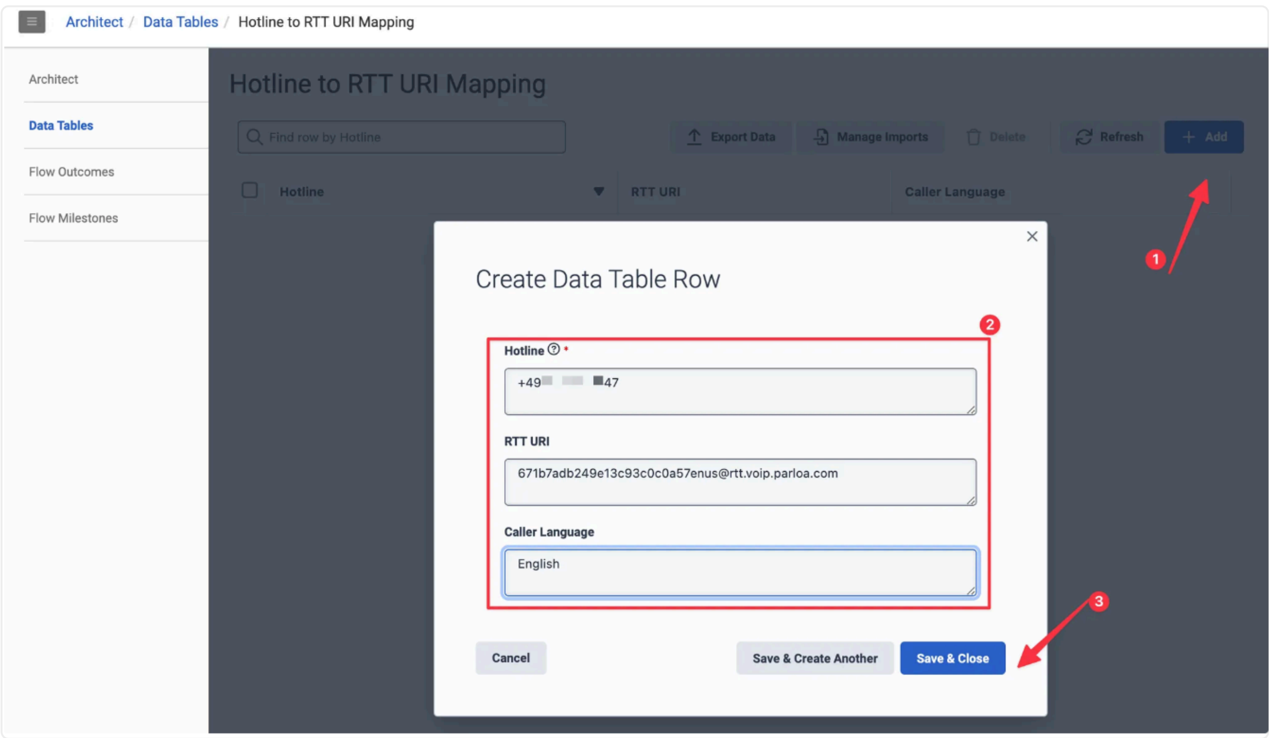
3. Click **Add Field**, then String and name it something like **RTT URI**. Optionally, you can add a third field with the caller language. When you're done, click **Save**.



4. Before we add rows to our table, we must first get our RTT SIP URIs. Log into https://augment.parloa.com, go to **Settings → Telephony → Inbound Flow**. Here you will find your Inbound (or caller) languages, each with its own URI. In our example, we will use English and German languages. Copy these URIs.



5. Go back to your Data Table in Genesys and click **+ Add** to add a new row. Input your hotline number with the corresponding URI. Click **Save & Close** and repeat as necessary.

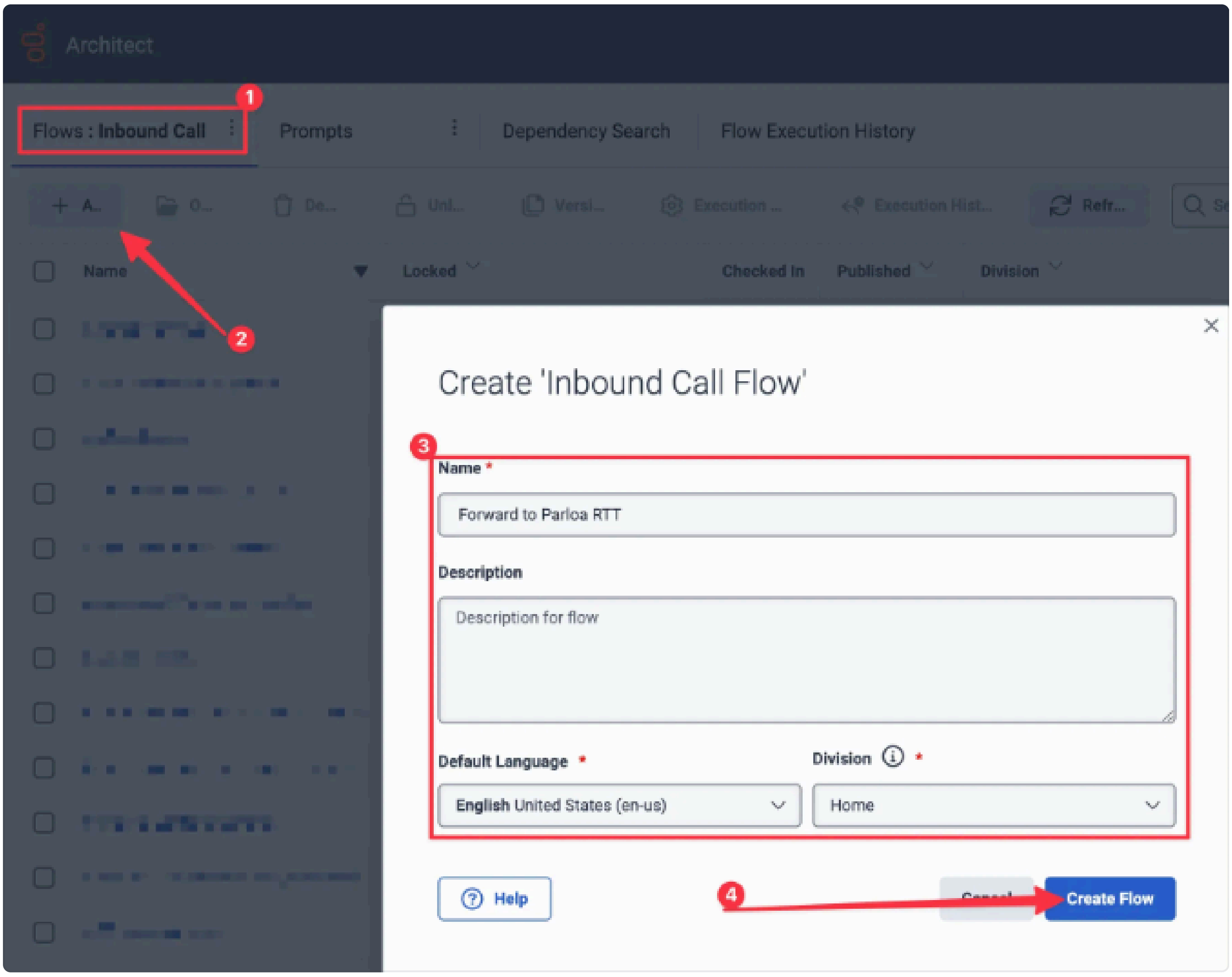


6. Our end result with two languages looks like this:

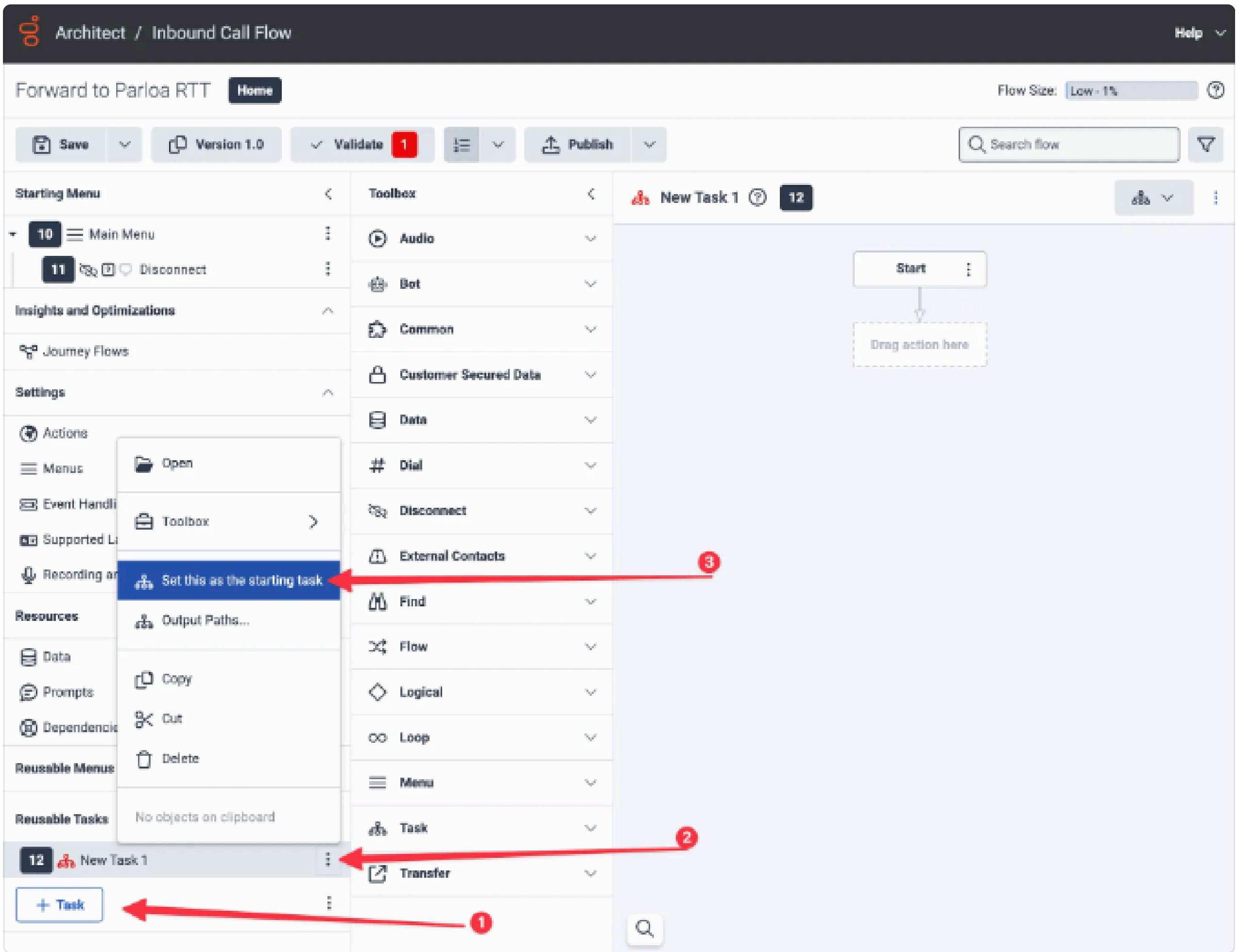
Inbound Call Flow

Now that we have our Data Table, we have to create a new Inbound Call Flow to use said Data Table and forward the call to Parloa.

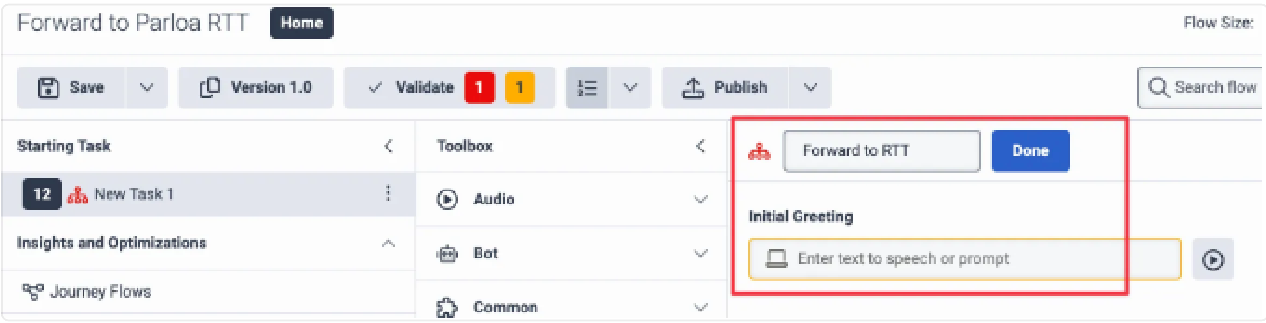
1. Navigate to **Admin** → **Architect** → **Architect**. A new tab will open. Make sure **Inbound Call** is selected under **Flows**. Click **+ Add**, give it a name and click **Create Flow**.



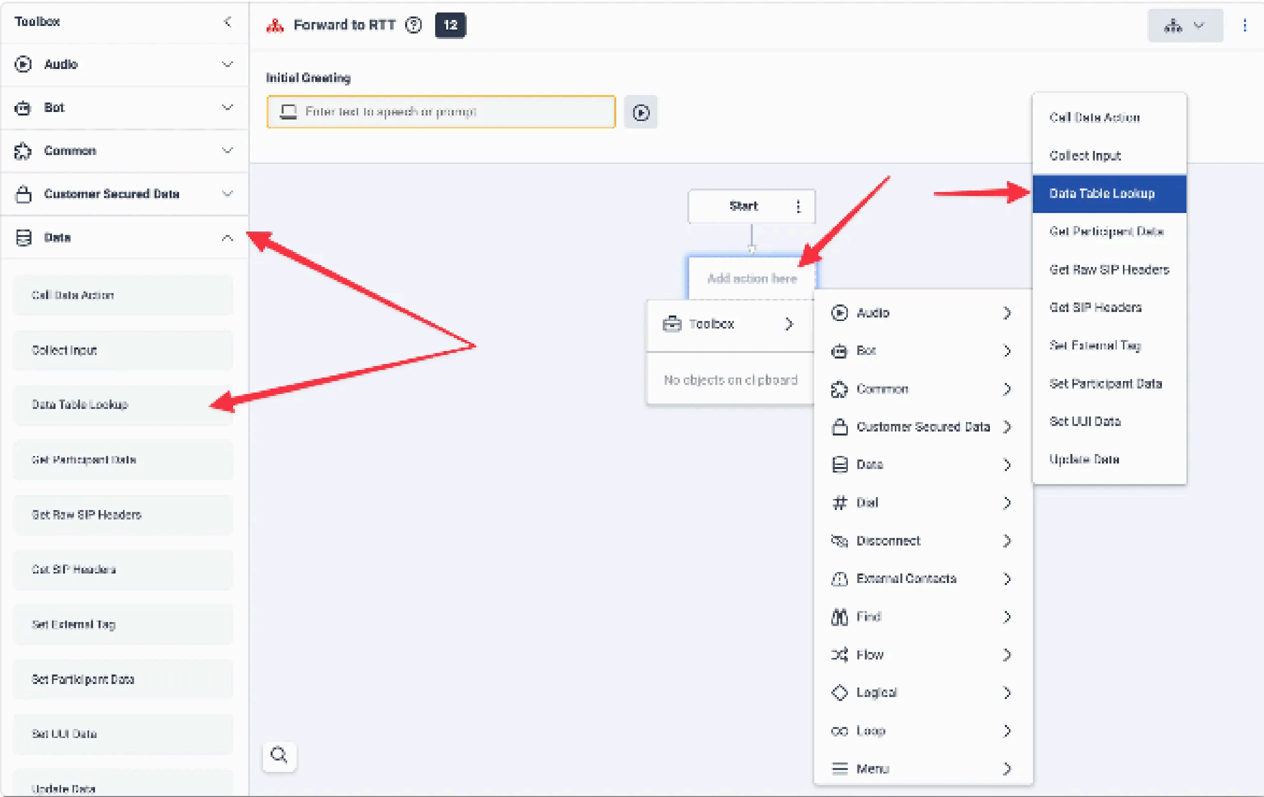
2. First thing we need to do: Add a new **Reusable Task**, and set it as the starting task.



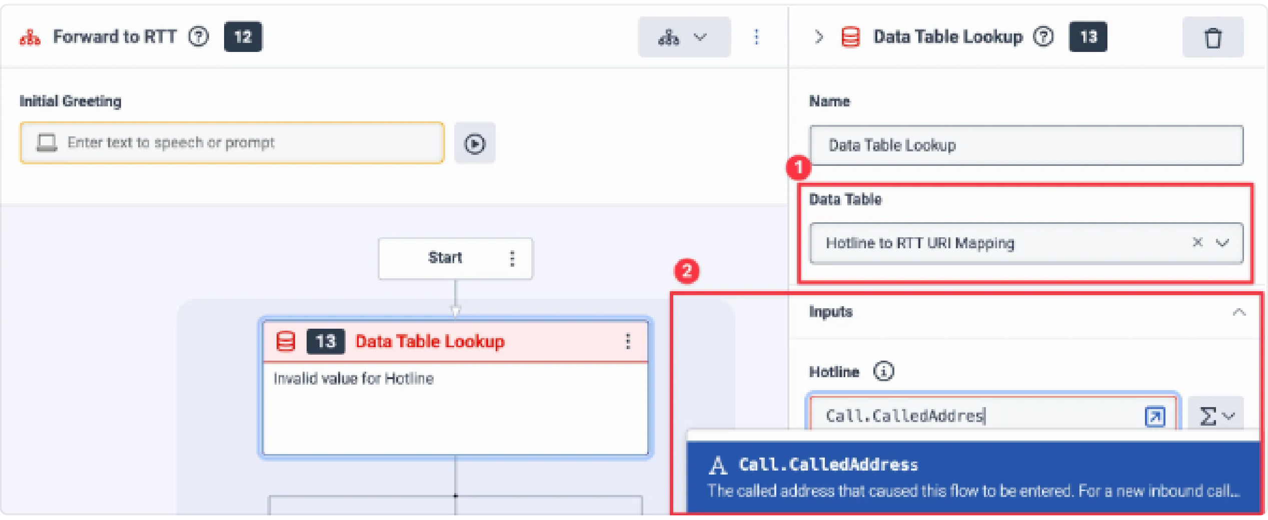
3. Delete the **Initial Greeting**, and feel free to rename the task. This is so that no announcements will be played. This is mandatory to avoid that sentences that are not spoken by the CSA are recognised by RTT and sent to the caller.



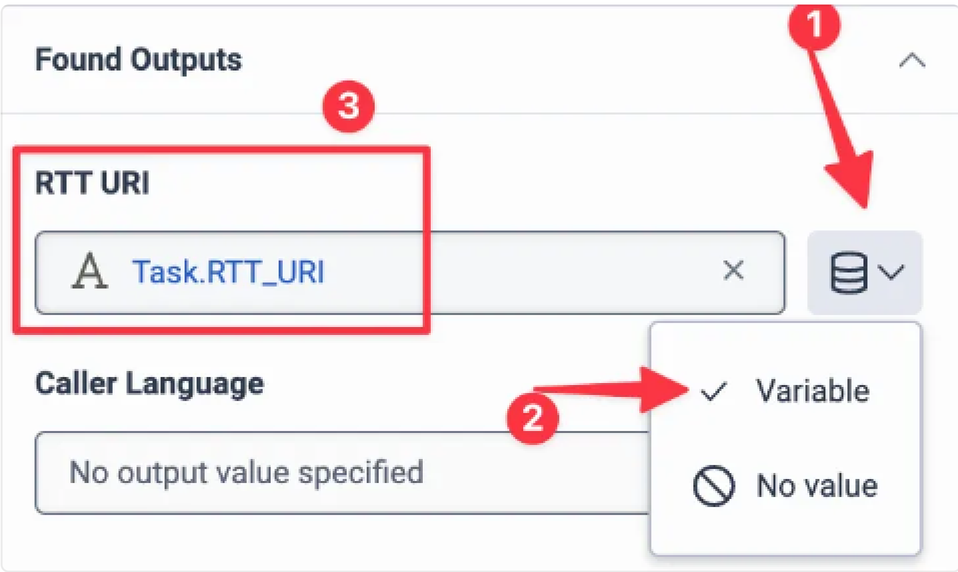
4. Now we'll focus on the flowchart. Let's add a **Data Table Lookup** action. You can add it by either dragging from the toolbox to the left, or by clicking the "Drag action here" box.



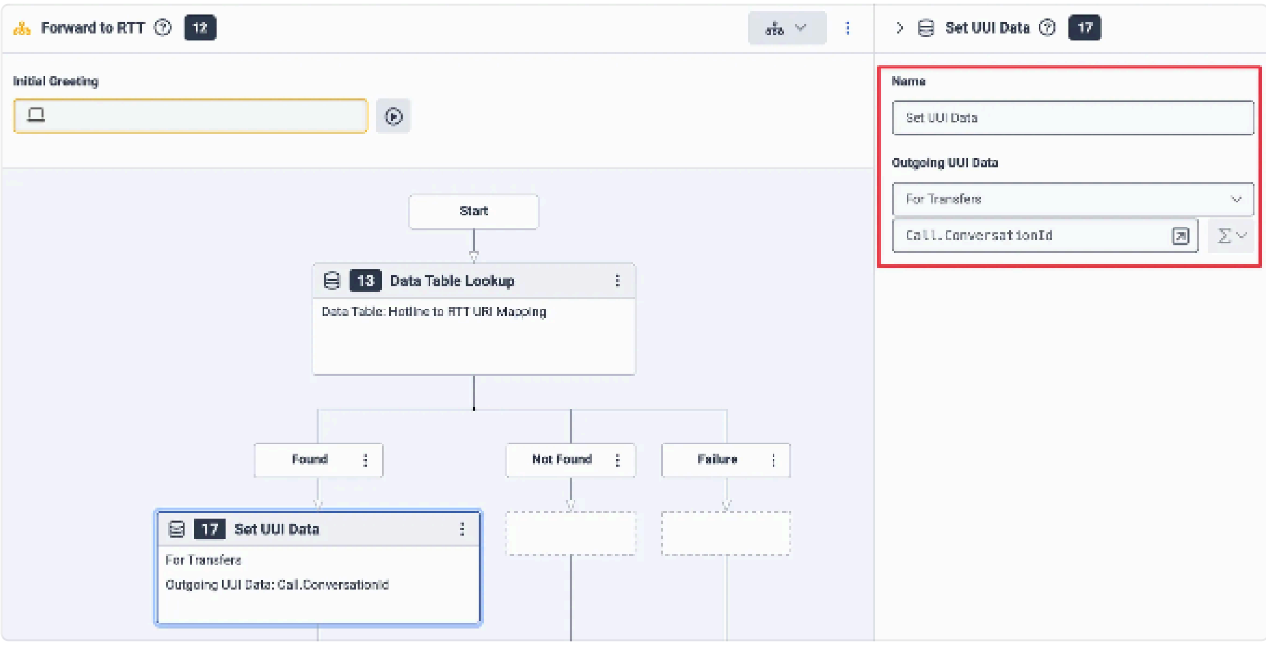
5. On the right hand side, choose the Data Table we created. Under Inputs, there should be a field for the Hotline. Here we need to pass the hotline number that was dialed to reach this Inbound Flow, which are also present in the Data Table. For this, you can simply input `Call.CalledAddress` (Genesys will autocomplete if done correctly)



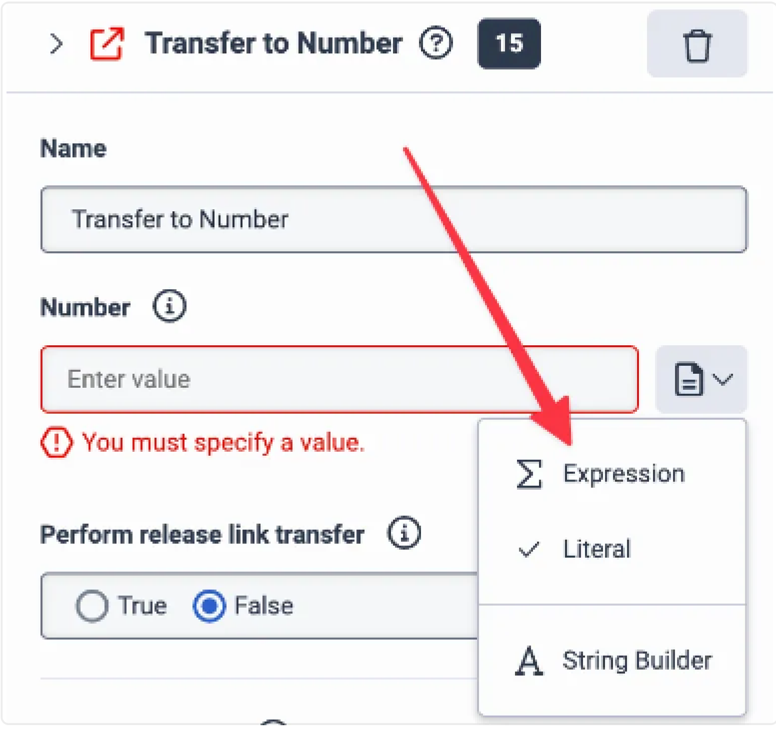
6. Genesys will do a lookup in the table for the hotline number, and will return the rest of the data. In this example, we only care about the RTT URI. Under Found Outputs, create a new variable to store the returned value. Note: there is no need to add `Task`. It will be done automatically by Genesys.



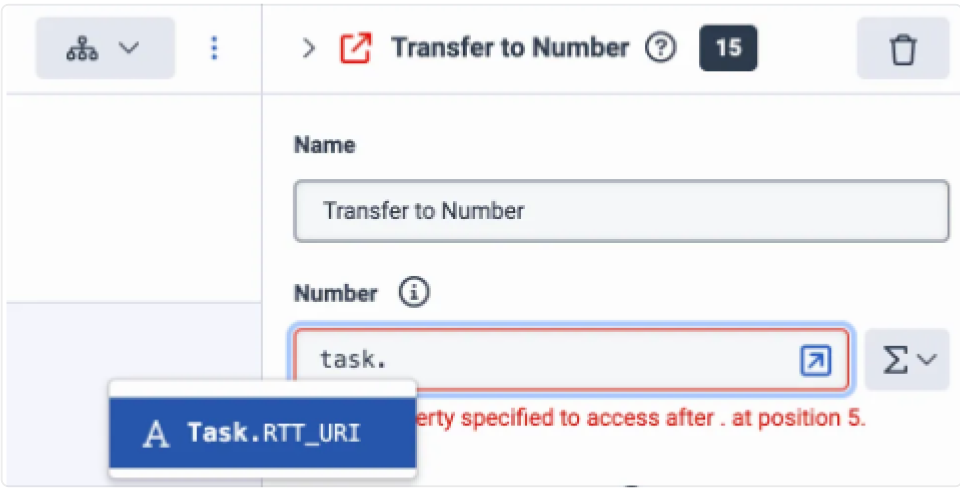
7. Now that we have our RTT URI, we need to forward the call. Before we forward the call, we need to add User-to-User info (UUI) to the call, this will help with the Agent Pop Up later on. Under the Found branch, add a **Set UUI Data block**. Select For Transfers, type in `Call.ConversationId`. This ensures we send a unique identifier on each call.



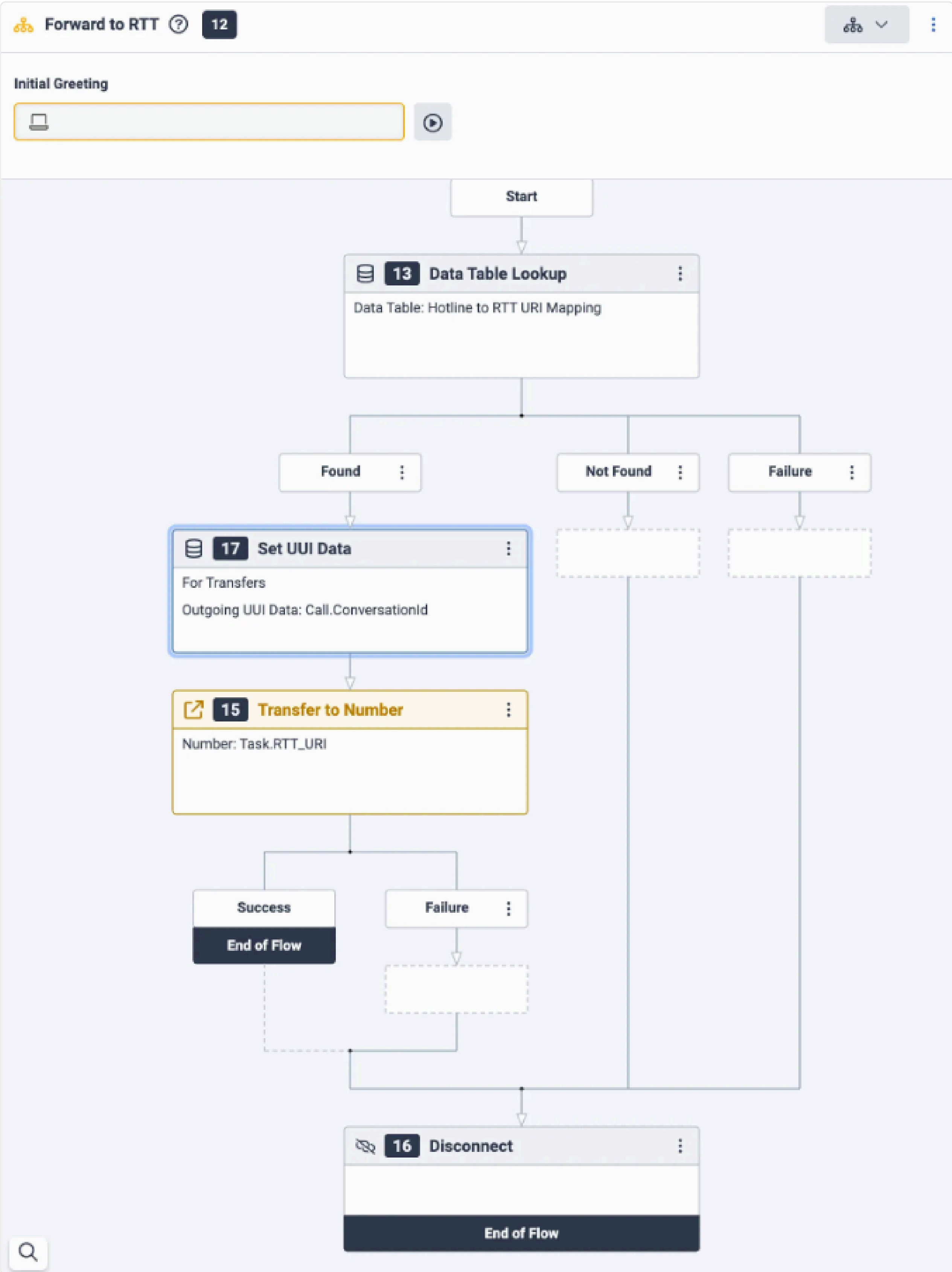
8. Under the **Set UUI Data**, add a new block: **Transfer to Number**. Change the Number field to Expression



9. Enter the variable created in step 6. Leave the rest of the settings unchanged. Number in this case means destination, so also includes URIs.



10. Lastly, add a Disconnect action at the end of the flow. (You can add additional steps and/or error handling, but we are doing the very basics.) In the end, it should look like this:

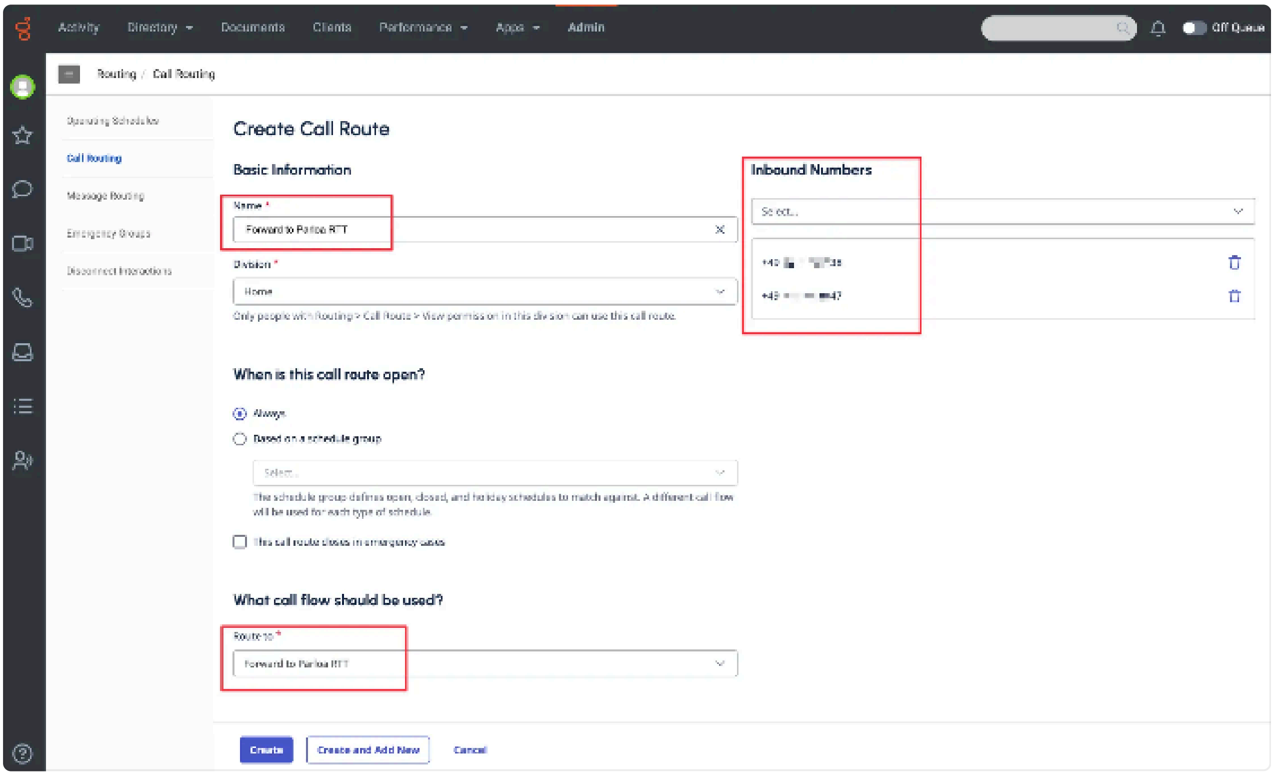


11. Click **Save** and then **Publish**.

Call Routing

Now that we have our Inbound Call Flow, let's configure a route to trigger this Inbound Flow.

Navigate to **Admin** → **Routing** → **Call Routing**, and click **+ Add** to create a new route. Give it a name, select the **Inbound Call Flow** we just created in the previous section, and select the hotline numbers (or inbound numbers) that will trigger this flow. These numbers should be the ones added in the Data Table we created earlier.

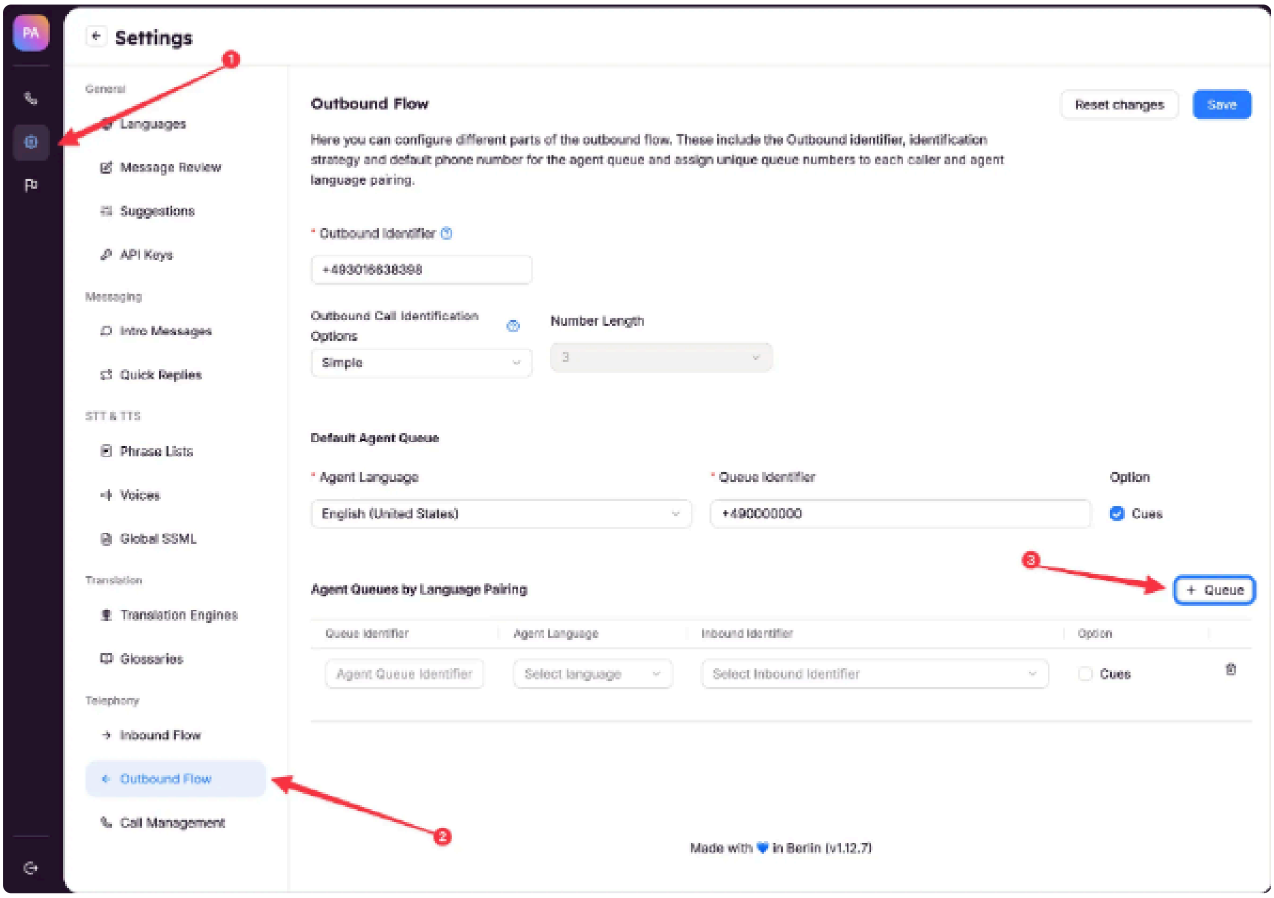


And you're done! If you followed these steps properly, you should be able to reach Parloa RTT's different languages based on the number dialed.

Configuring the Call Flow in Parloa

As described in the Call Flow diagram, RTT orchestrates two calls. Now that you configured the first call to Parloa RTT, we need to configure Parloa to generate the outbound call (the agent or the translated call) to Genesys. Before you continue, make sure you have an inbound number (can be a virtual number) that we can use.

1. First, we will pair a caller language with an agent language. Log into https://augment.parloa.com, go to Settings → Telephony → Outbound Flows. Click + Queue to add a new Agent Queue.



2. Once you add an Agent Queue:
   a. Provide a Queue Identifier. This is the endpoint where Parloa will send the agent call to. This is where the inbound identifier of Genesys comes in, the one you created in Step 4. The Queue Identifier should have the following format:

```
sip:<inbound number>@<inbound identifier>.byoc.mypurecloud.de;transport=tls
```

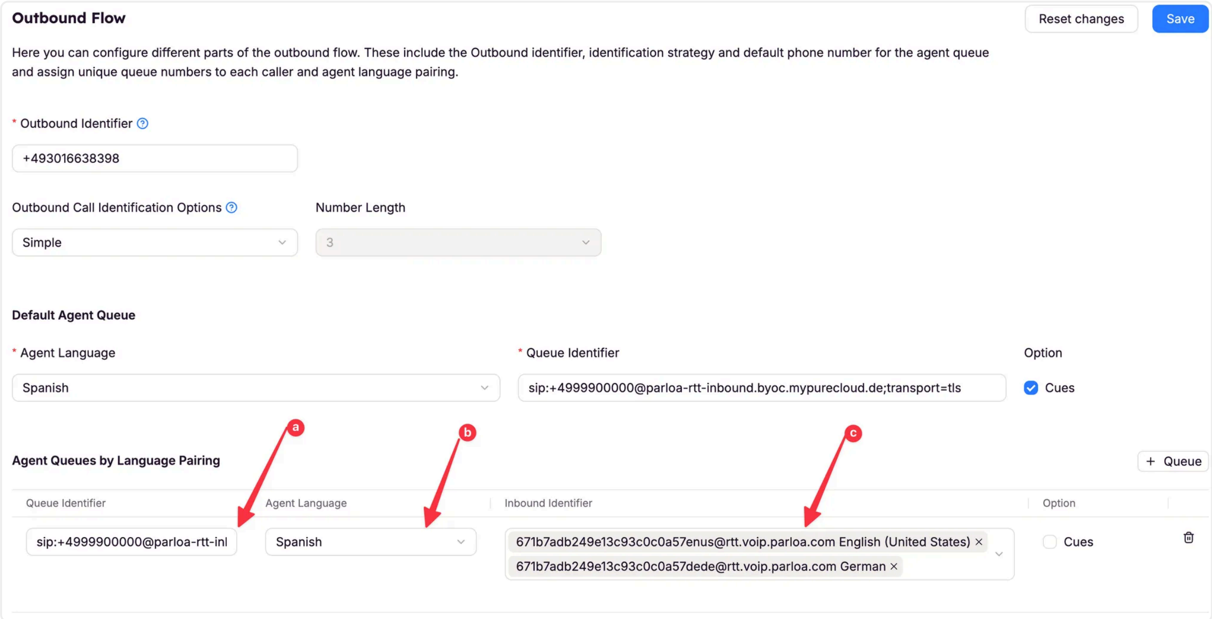
   In our example, our inbound Genesys identifier is `parloa-rtt-inbound`, and my inbound number is `+4999900000`. So this means our **Queue Identifier** in Parloa should be:

```
sip:+4999900000@parloa-rtt-
inbound.byoc.mypurecloud.de;transport=tls
```

You can also set this as your Default Agent Queue

b. Chose the **Agent Language**. This will be the language in which the Agent in **Genesys** will receive the call. In our example, our Agent will speak Spanish.

c. Add the **Inbound Identifiers**: These are the caller languages that will be routed to that particular Agent Queue. In our example, both English and German callers will be routed to a Spanish speaking agent. Don't forget to click **Save!**



That's all you have to do in Parloa! We now configured that both English and German speaking callers will be routed to Spanish speaking Agents in Genesys.

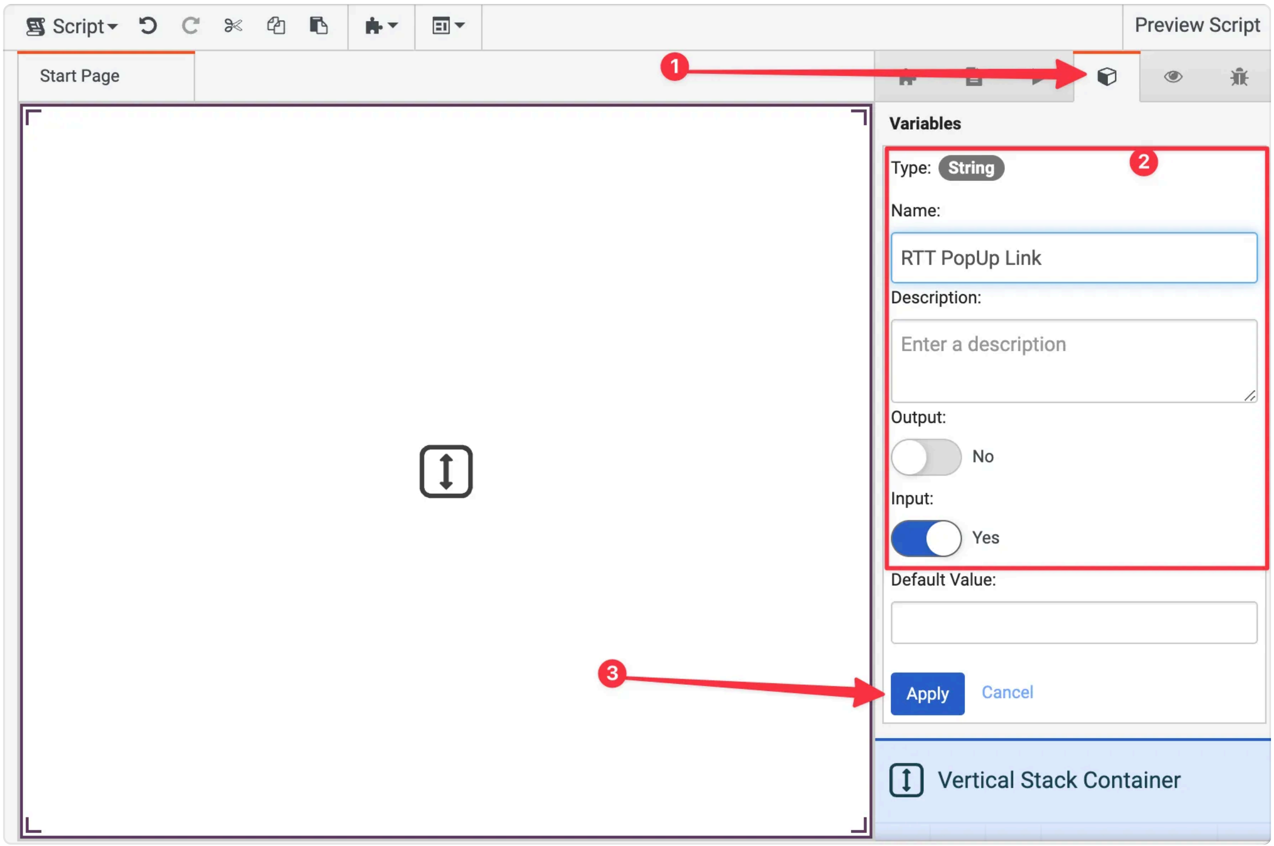
Configuring the Pop Up Screen in Genesys

One of the benefits of using Genesys Cloud and connecting via SIP to Parloa, is that we can send call metadata in the headers. In this section, we will use the metadata to automatically open the correct call when the Agent receives the call in Genesys.

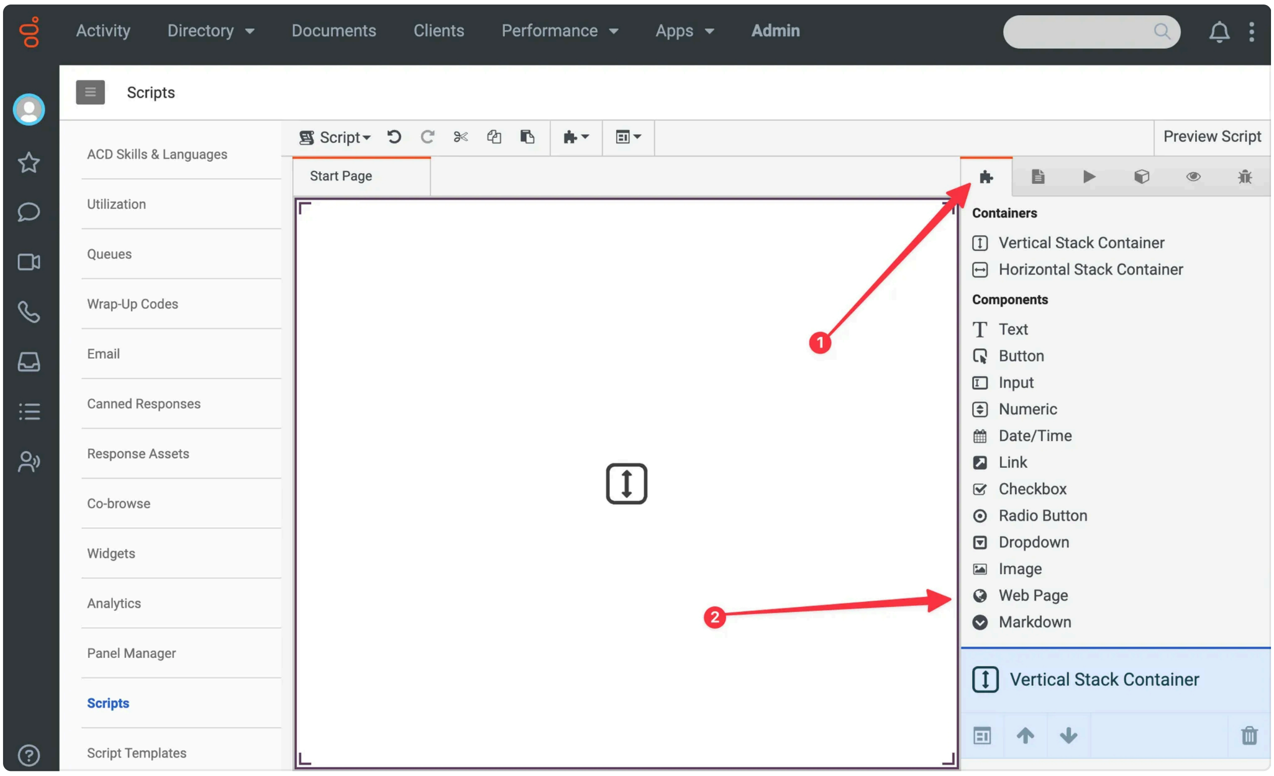
Agent Script

When the agent receives a call, we want to make it easy for the agent to open the same call in the RTT interface, so we will configure a script to automatically open the correct conversation.

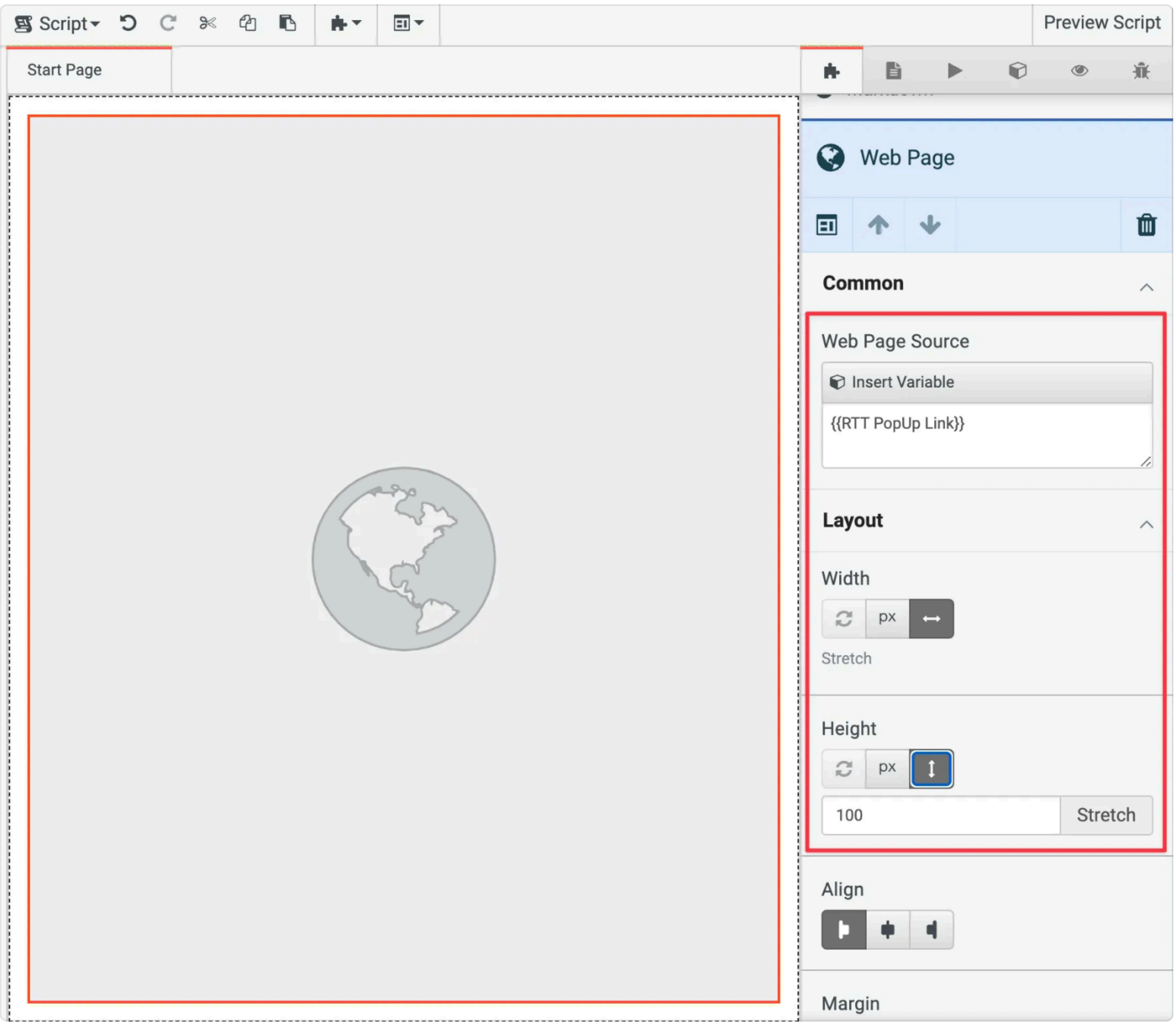
1. Navigate to **Admin → Contact Center → Scripts** and create a new blank script. First we need to create a new variable for the Pop Up link. Click the **Variables** tab and create a new **String** type **variable**, and enable it as an **Input**. Click **Apply**.



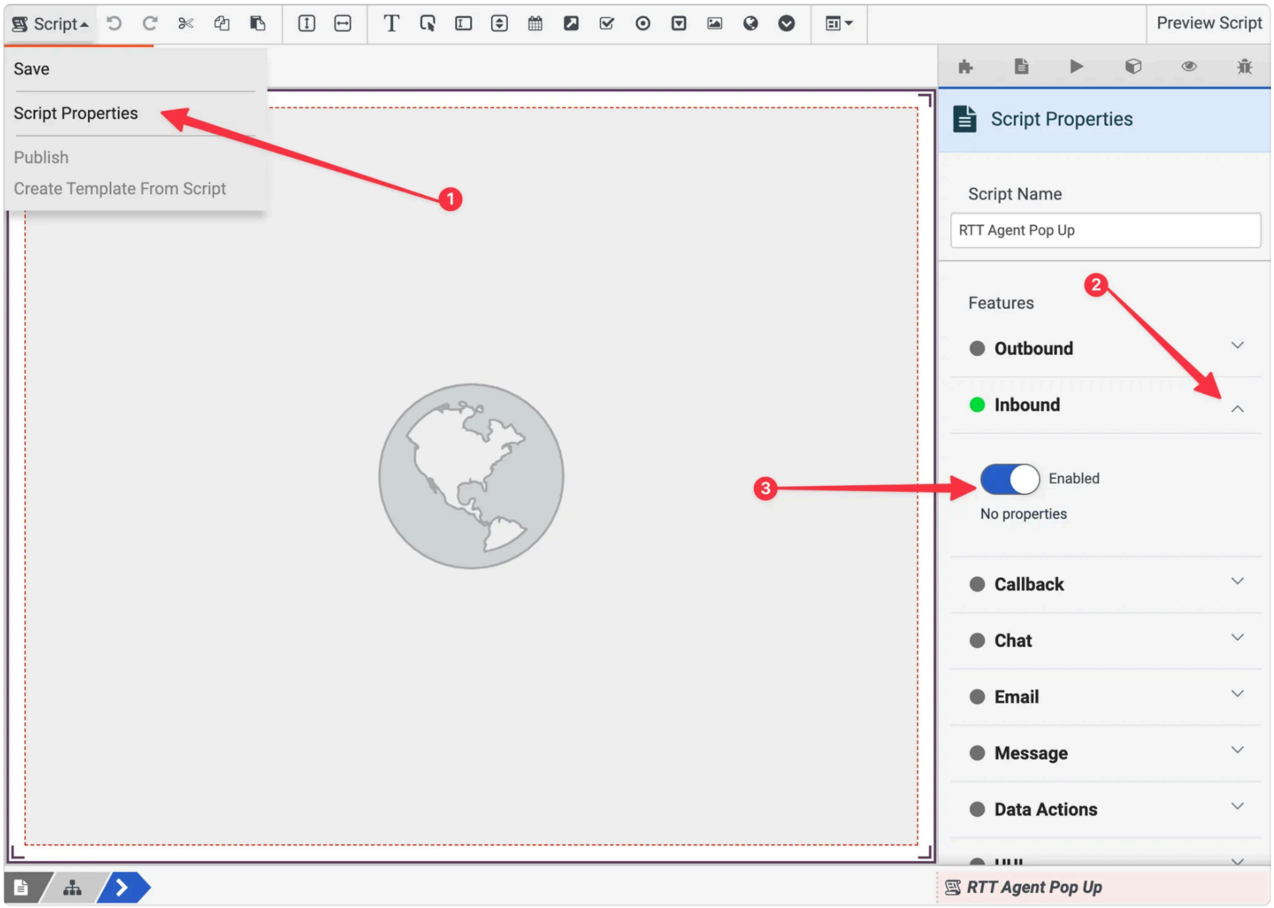
2. Go back to the containers tab, and click Web Page.



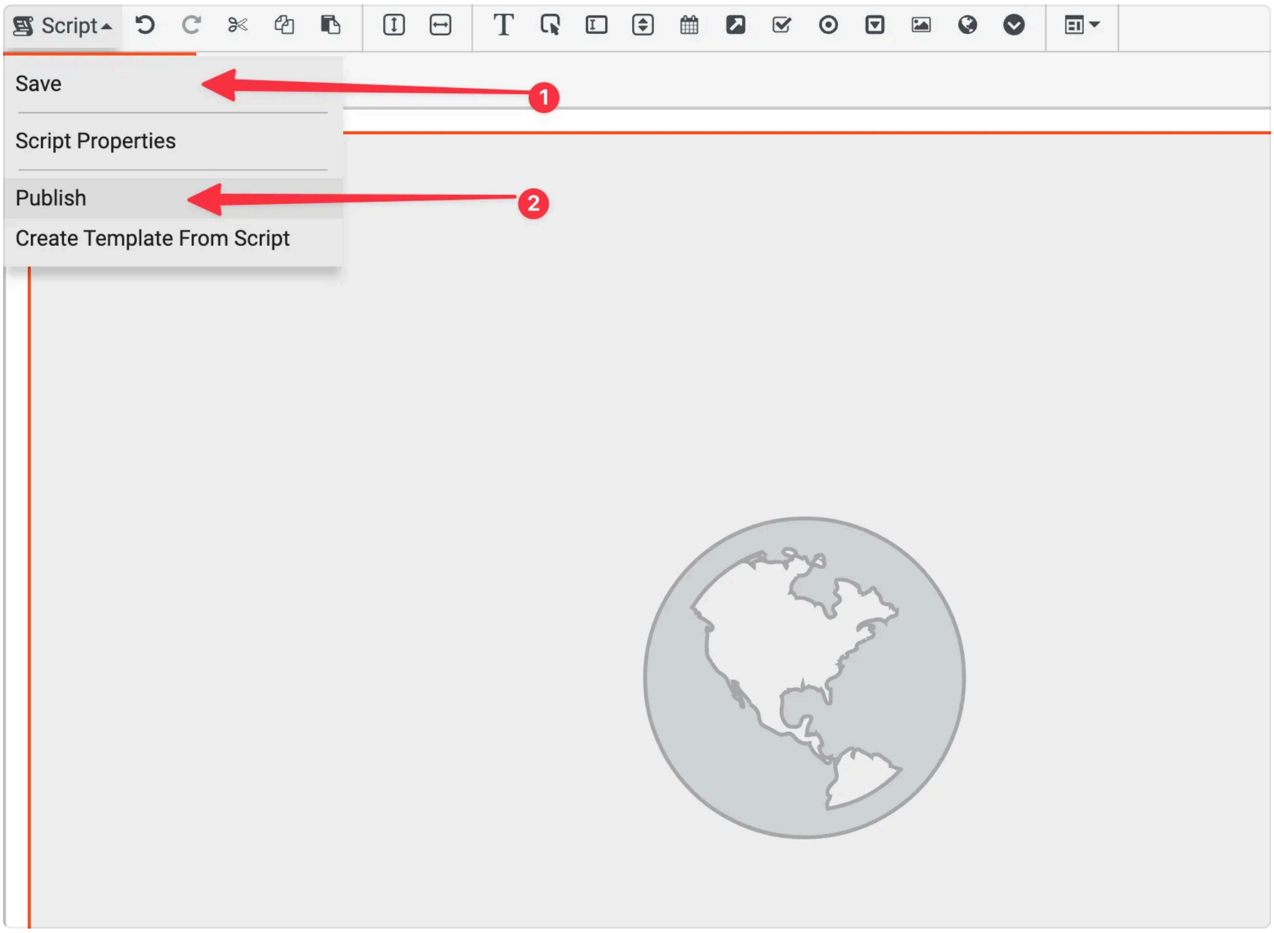
3. On the Right hand side, a few options will appear for our Web Page component.
    a. Under Common → Web Page Source, click Insert Variable and select the variable you just created in the step before.
    b. Under Layout, change both Width and Height to Stretch.



4. Lastly, we need to enable the Script for inbound use. Click on Script → Script Properties, on the right hand side expand Inbound and enable it.

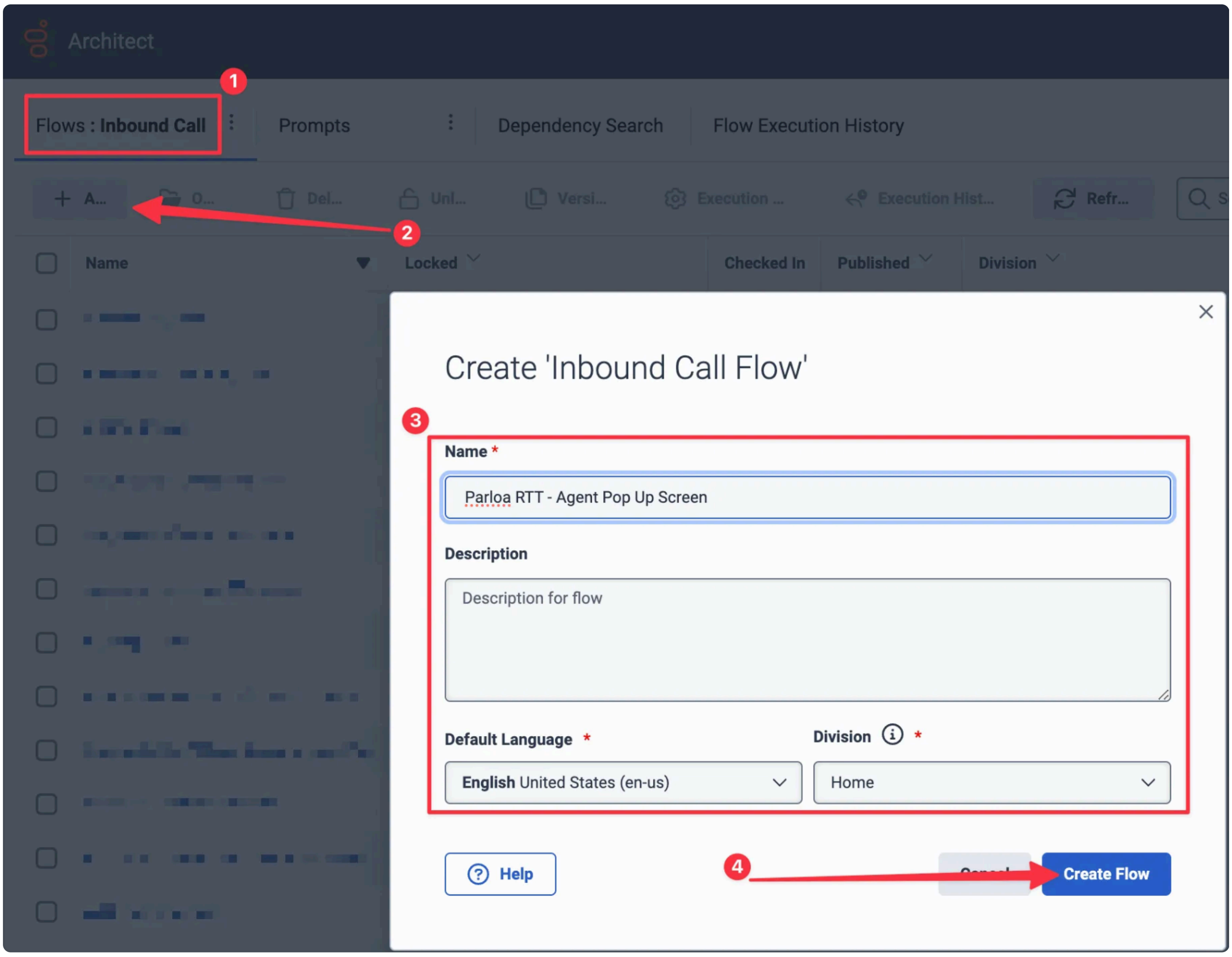


5. When you’re done, click Save and Publish.

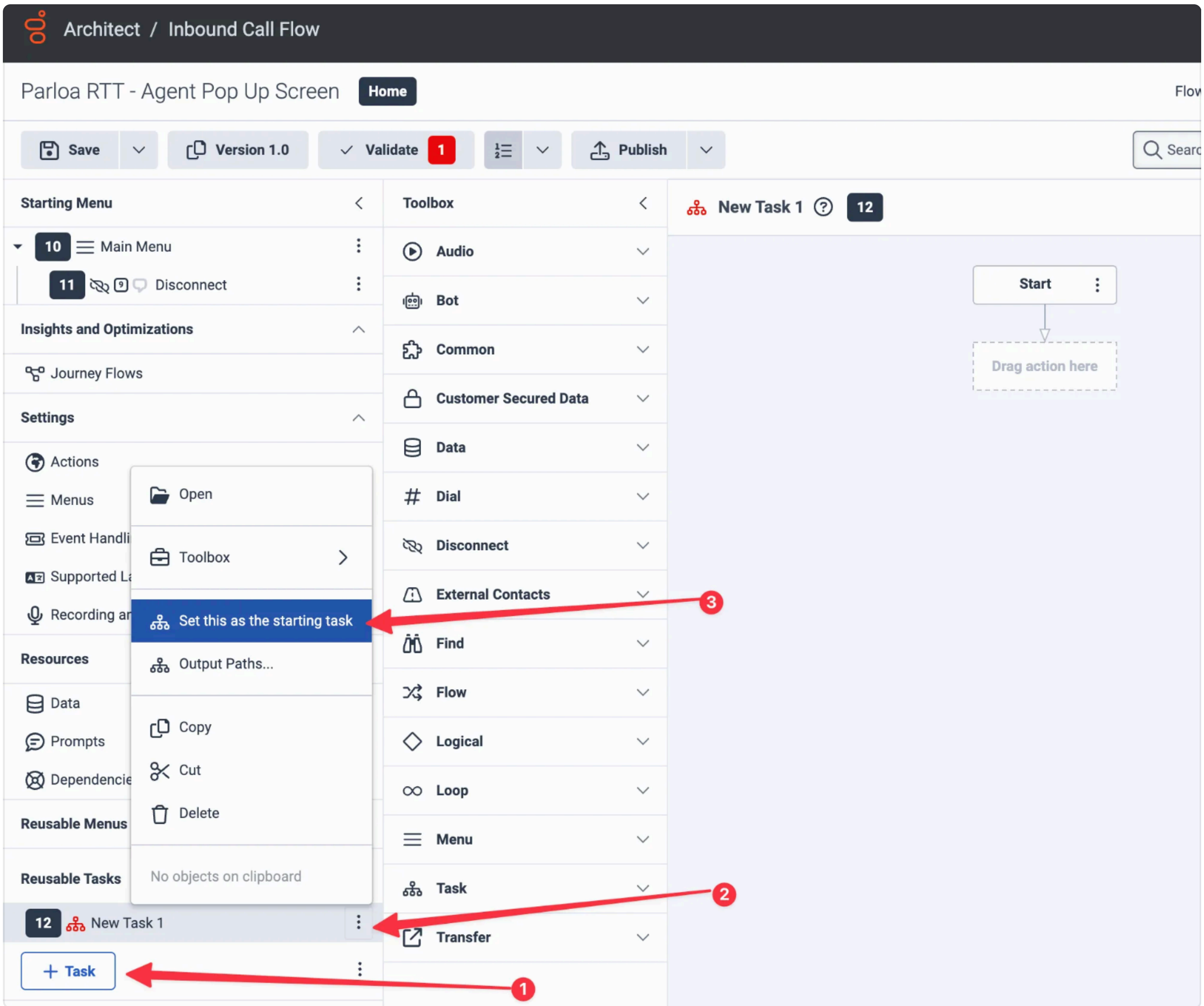


Agent Inbound Call Flow

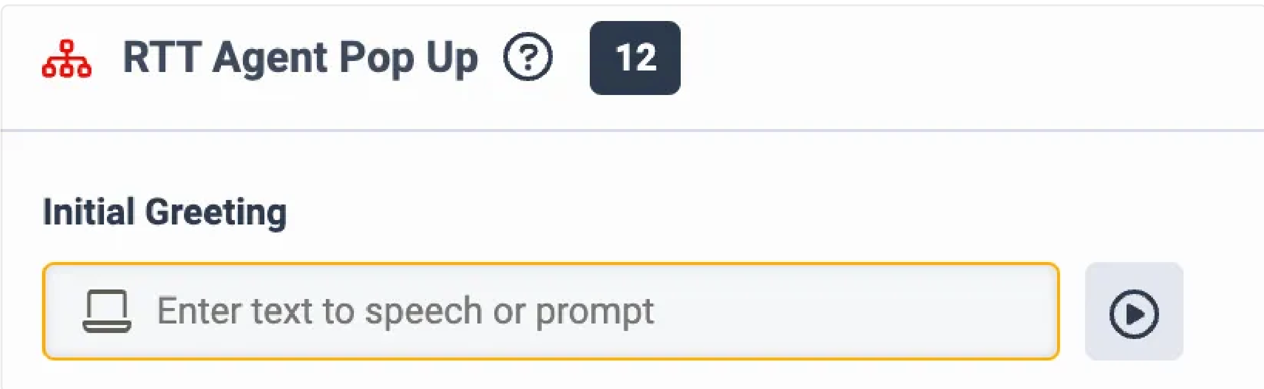
1. Now that we have our Script, let's make sure the Agents actually use it. Navigate to **Admin → Architect → Architect**. A new tab will open. Make sure **Inbound Call** is selected under **Flows**. Click **+ Add**, give it a name and click **Create Flow**.



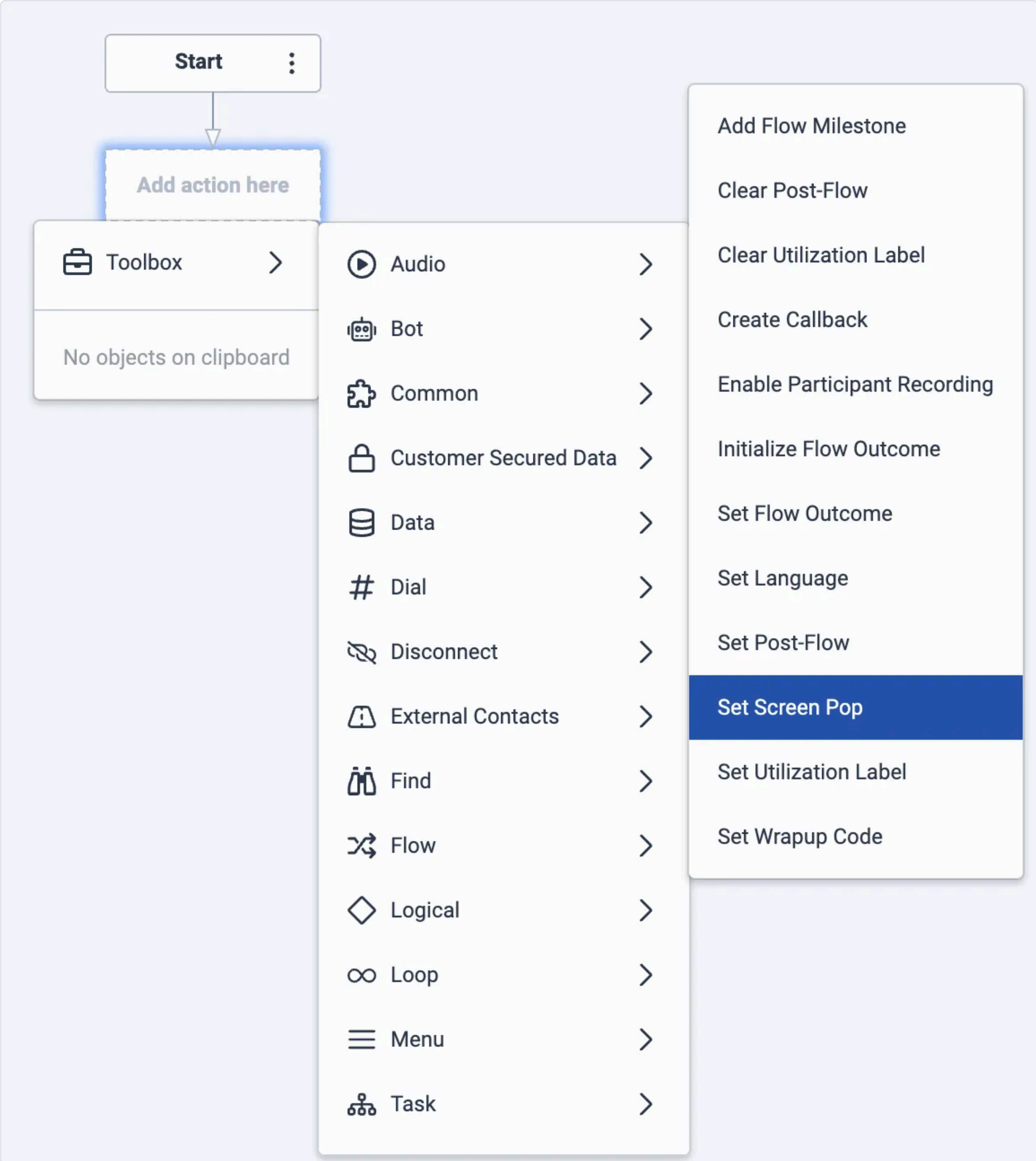
2. First thing we need to do: Add a new **Reusable Task**, and set it as the starting task.



3. Remove the Initial Greeting, and rename the task.



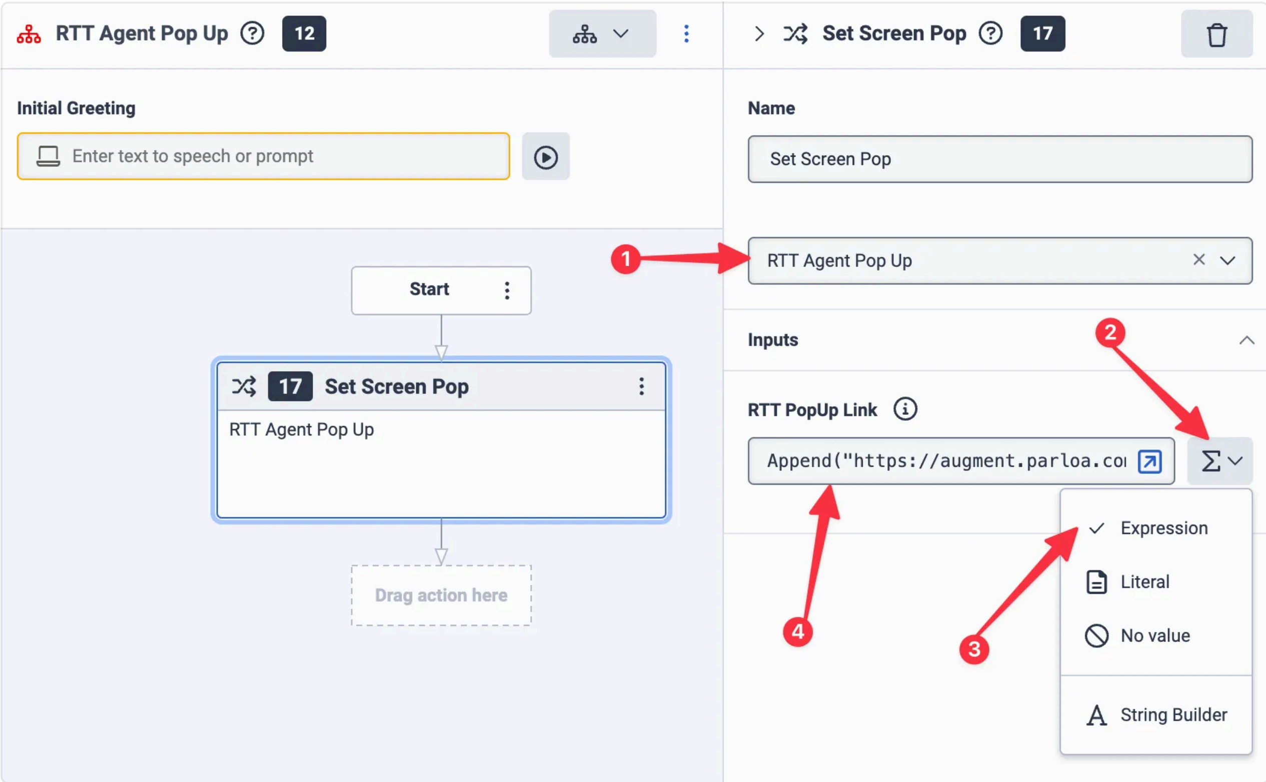
4. Now we'll focus on the flowchart. Add a Set Screen Pop block.



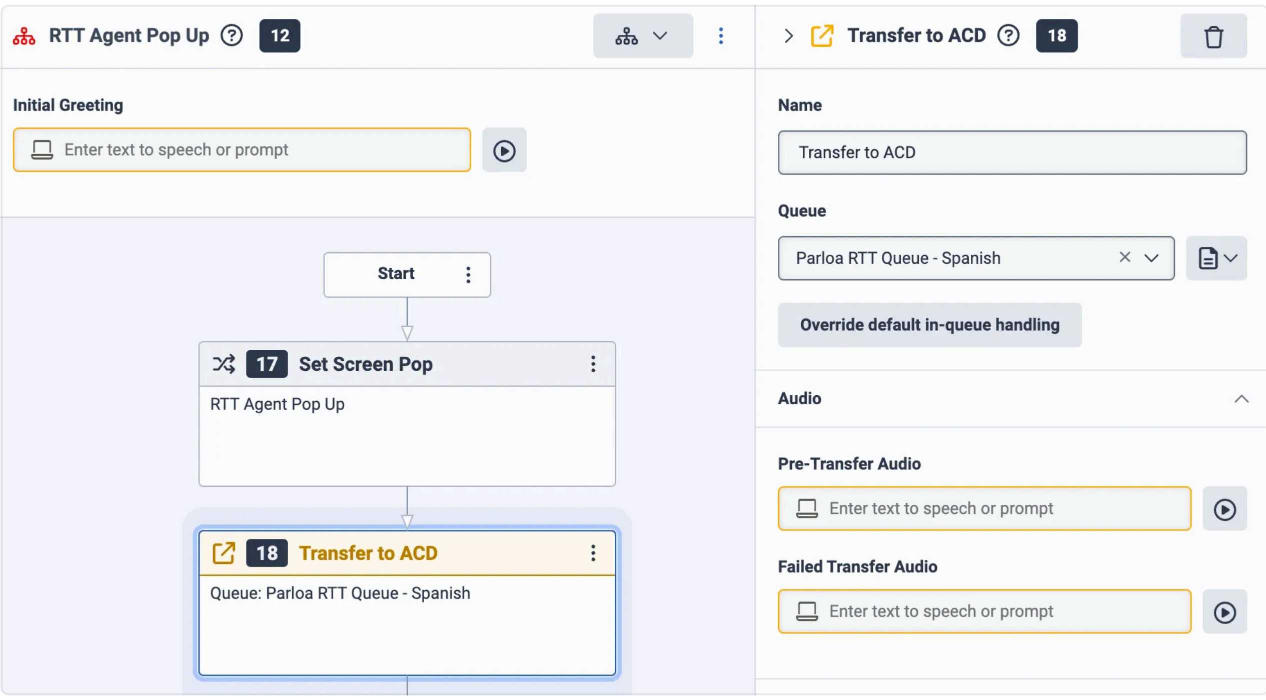
5. Select the Script we created in the previous section. In the **Inputs**, change to **Expression** and paste the following:

```
Append("https://augment.parloa.com/api/v0/conversations/", Call.UUIData,
```

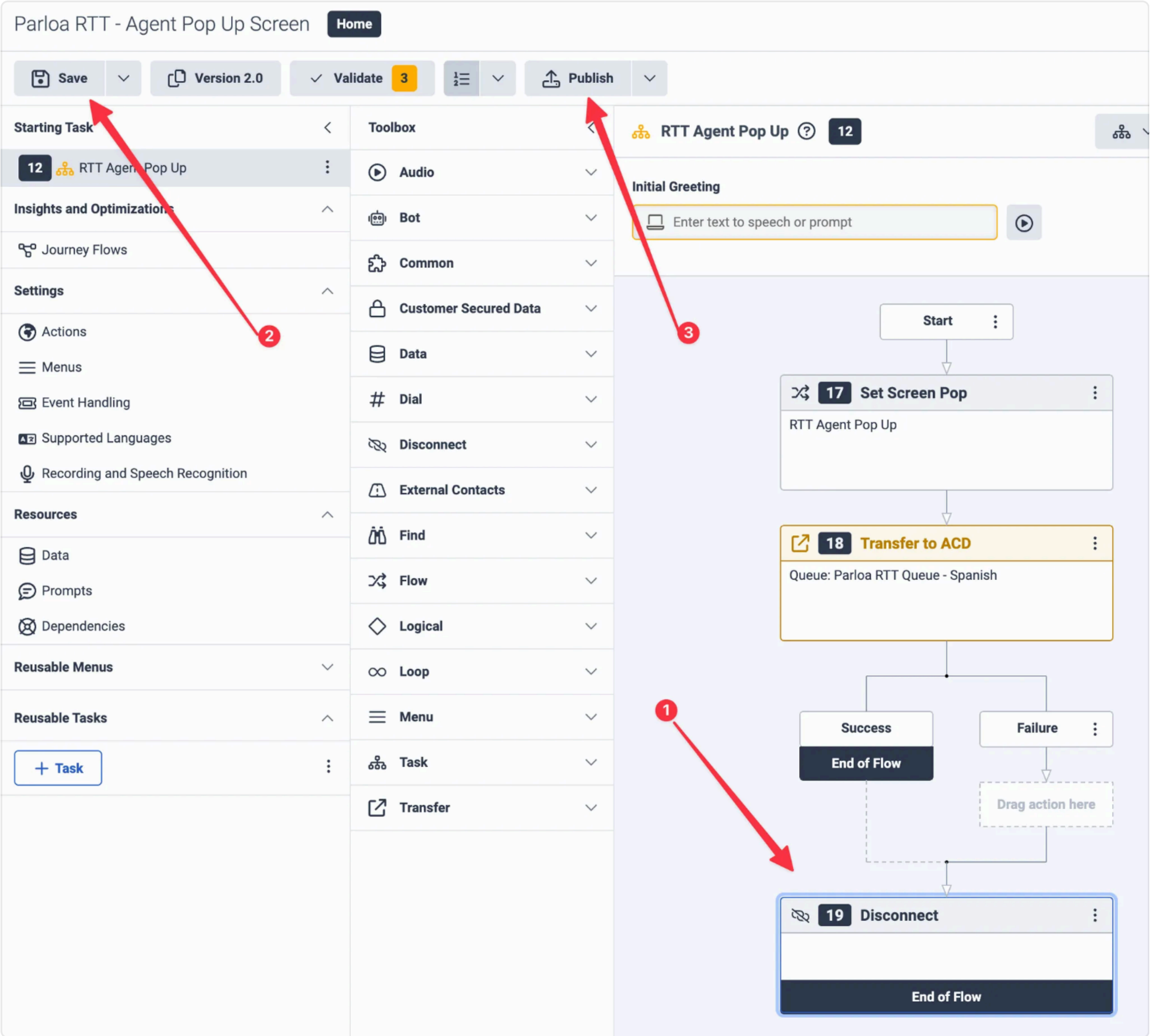
This will allow the the Script to open the correct conversation in the Augmented UI that corresponds to the call the agent received in Genesys.



6. Now that we have our script ready, we need to transfer to a queue. Add a **Transfer to ACD** block, and choose your preferred queue. In our example, our agents will speak Spanish so we are going to transfer to a Spanish speaking queue.



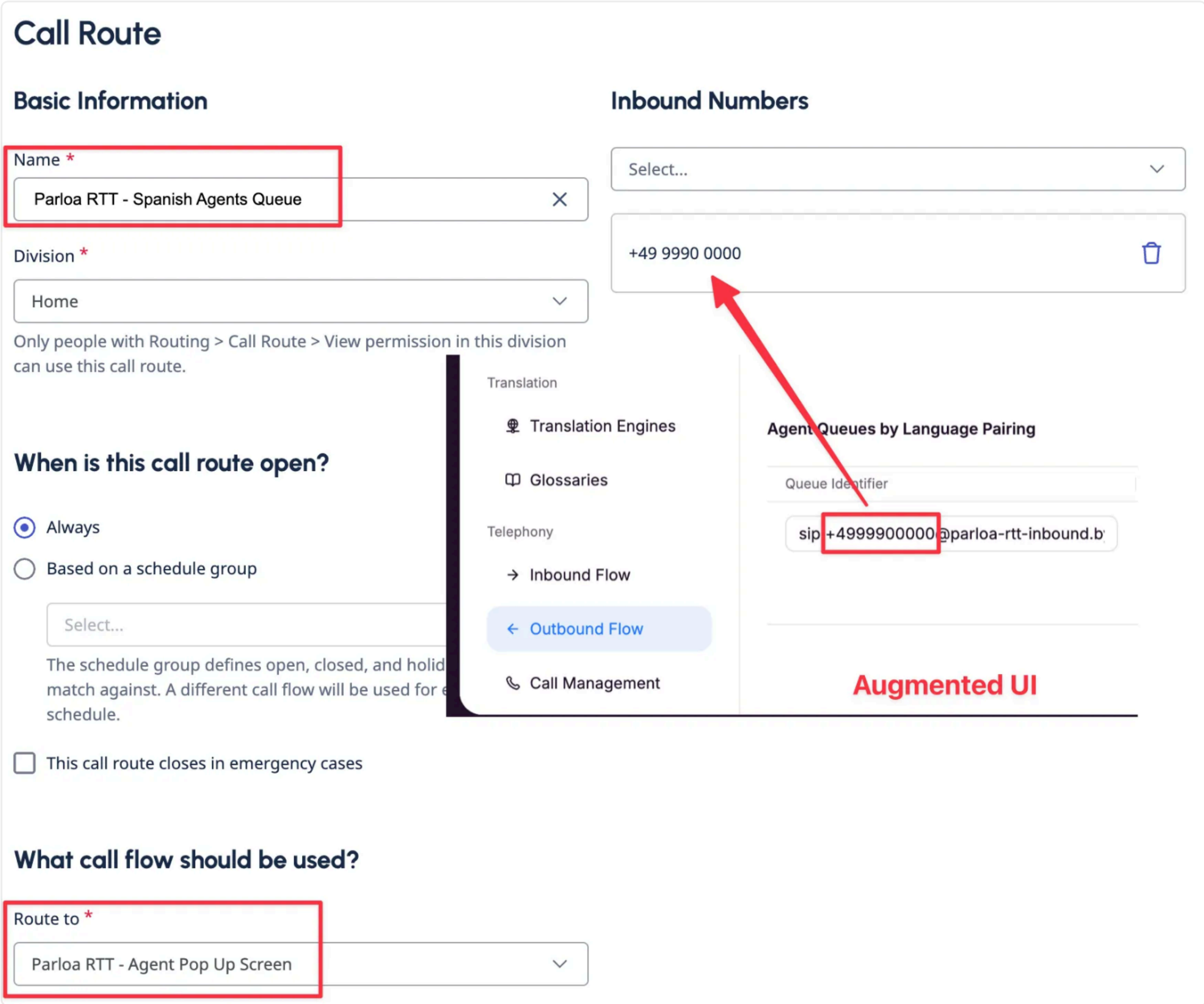
7. Lastly, add a **Disconnect** block in the end of the flowchart. After this, click **Save** and **Publish**.



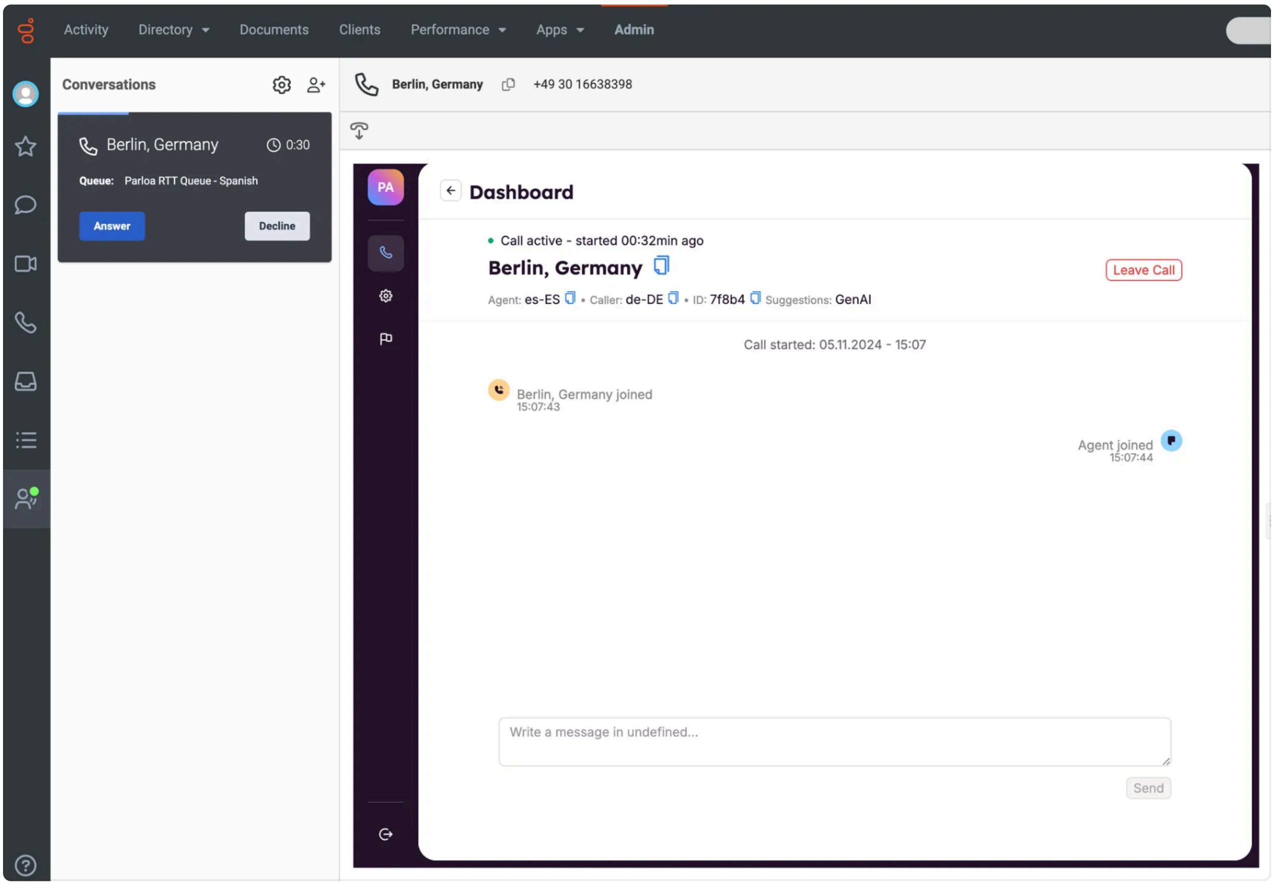
Call Routing to the Agent Queue

Now that we have our Inbound Call Flow and Script for the agents, let’s configure another route to trigger this new Inbound Flow.

Navigate to **Admin → Routing → Call Routing**, and click **+ Add** to create a new route. Give it a name, select the **Inbound Call Flow** we just created for the Agent Inbound Flow, and select the virtual number that you configured in the Augmented UI.



And you're done! If you followed these steps properly, you should be able to reach Parloa RTT platform, then your agent will receive Parloa's outbound call and should automatically open up the corresponding call in the Agent Desktop Screen


Activity
Directory
Documents
Clients
Performance
Apps
Admin
Conversations
Berlin, Germany
0:30
Queue: Parloa RTT Queue - Spanish
Answer
Decline
Berlin, Germany
+49 30 16638398
PA
Dashboard
Call active - started 00:32min ago
Berlin, Germany
Leave Call
Agent: es-ES • Caller: de-DE • ID: 7f8b4 Suggestions: GenAI
Call started: 05.11.2024 - 15:07
Berlin, Germany joined
15:07:43
Agent joined
15:07:44
Write a message in undefined...
Send