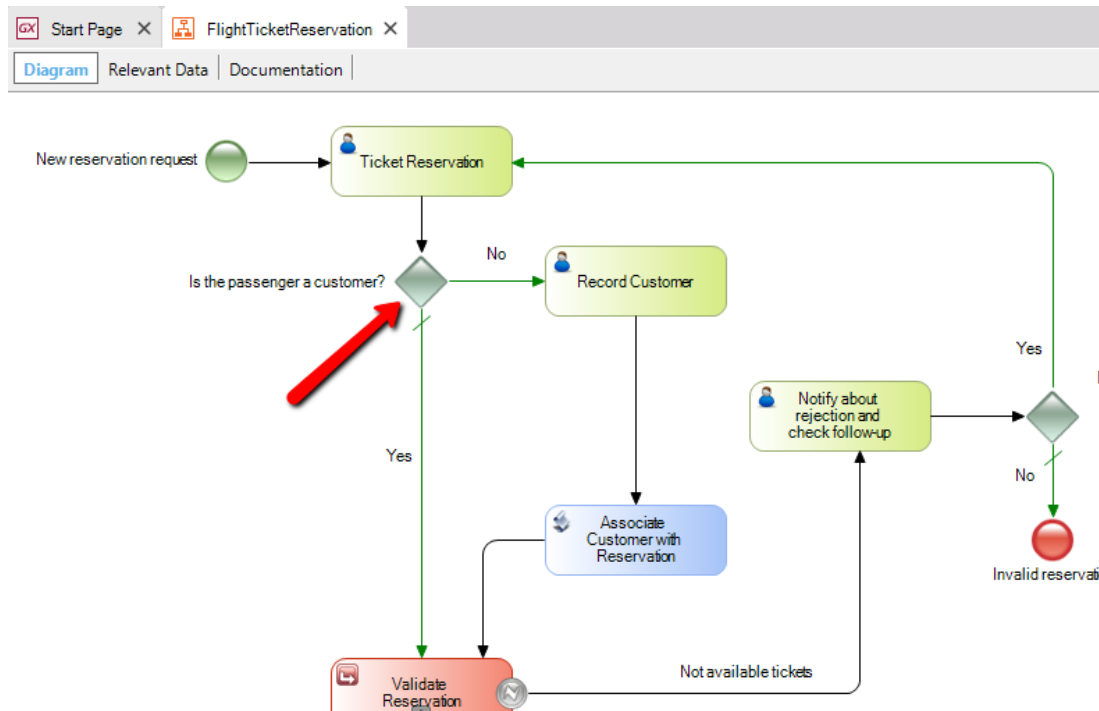


First execution of the process diagram.

In the video entitled Automation Part 1, we saw how to associate a transaction object to the task relative to recording travel ticket reservations.

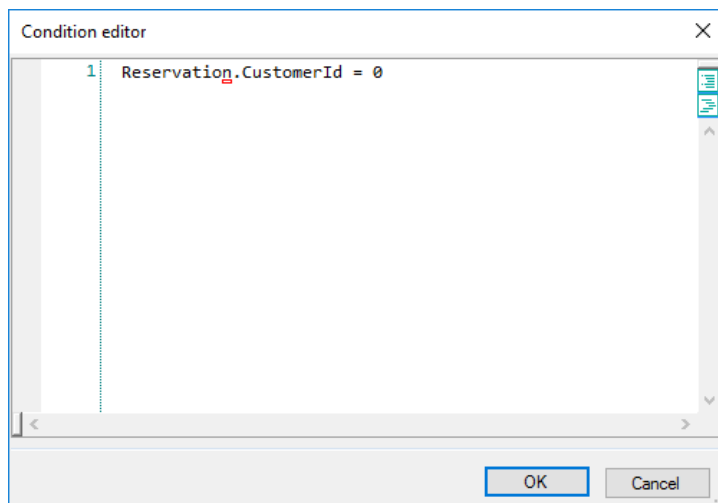
As we continue with our process diagram, once the reservation has been recorded, we must determine whether the passenger is one of the company's customers or not.



This implies that, as we enter the reservation data into the screen of the Reservation transaction, when the individual booking is not a customer we will not enter a value in the CustomerId attribute.

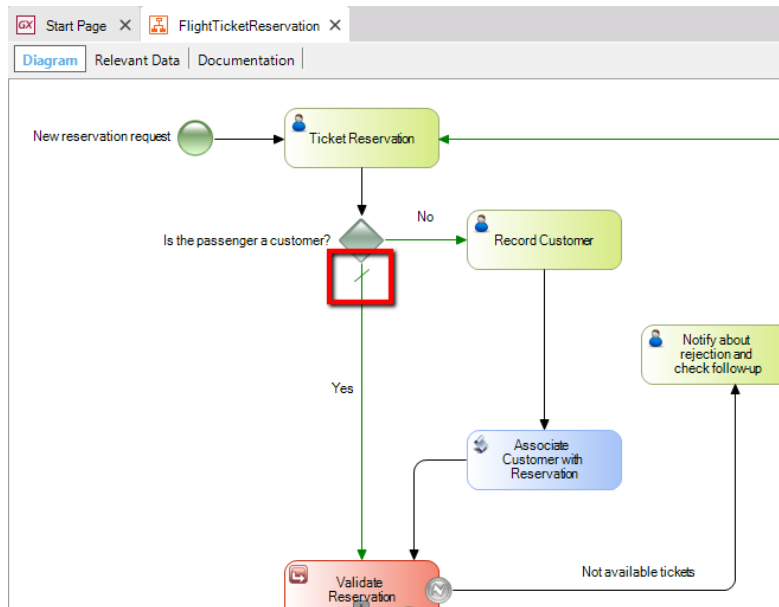
To assess the customer we double click on the connector that links the exclusive Gateway "Is the passenger a customer?" with the Record Customer task, and we will be opening a window of the **conditions editor**.

To determine whether we enter a value or not, we type `Reservation.CustomerId = 0`



Note that we **use the transaction's name to identify the CustomerId attribute, in order to determine the attribute that is foreign key in the Reservation transaction**, corresponding to the CustomerId attribute that is primary key of the Customer transaction.

Let's now press OK and get back to the diagram. We have already entered a condition in the connector labeled No. However, it is not necessary to enter a condition opposite to the previous one in the connector labeled with Yes, because we had already indicated it as Default.



In this case, if the condition entered in the No is not accomplished, then the program flow will automatically continue through the Yes connection.

Let's now click on the Record Customer task and its Application property and associate it with the Customer transaction. We will see that the relevant data CustomerId is created.

Application Declaration

Application: Customer

Data Mappings

Primary key attribute	Relevant data
CustomerId	CustomerId

OK Cancel

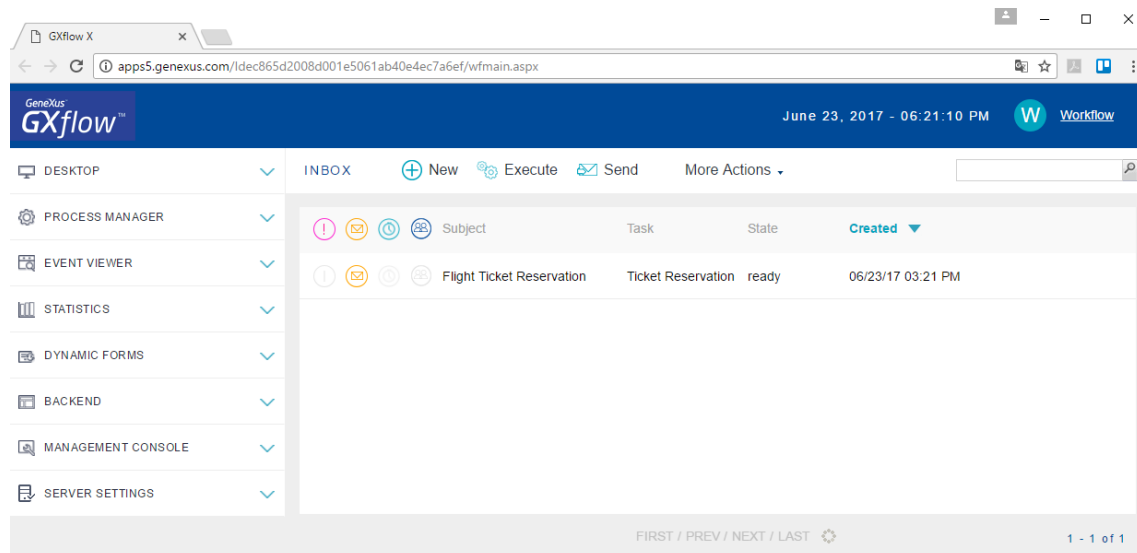
Let's now press OK and execute our process diagram for the first time.

But before we do that, let's complete the conditions of the other gateways in the diagram so we don't get an error in the execution. We only have one more, so we double click on the connection that links the exclusive Gateway "Do you make another reservation?" with the TicketReservation transaction and, for the time being, we set its value as True. Further ahead we will set the corresponding condition for it.

To execute our diagram we right click on the diagram name and select "Run".

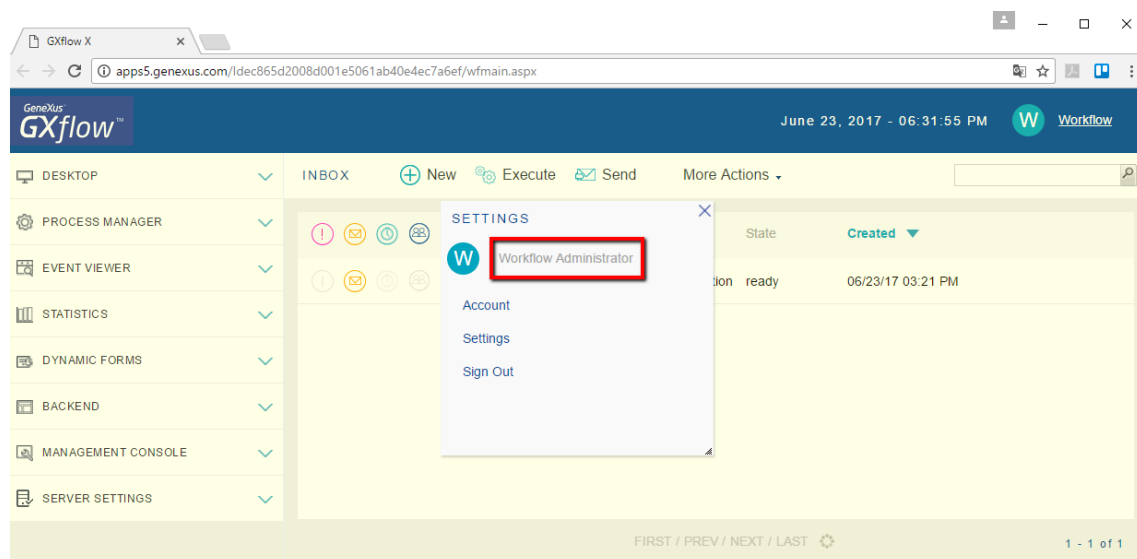
Since this is the first time we execute the diagram, the database will be created, as well as the tables that will store the definitions of the process we model. This information is called metadata and is managed by the workflow engine for executing and administering processes.

After the specification, generation and compilation of the KB's objects are complete, we will see the start of a window in the browser showing a main window of **Gxflow Client**.



This application enables us to execute, monitor and manage the execution of processes.

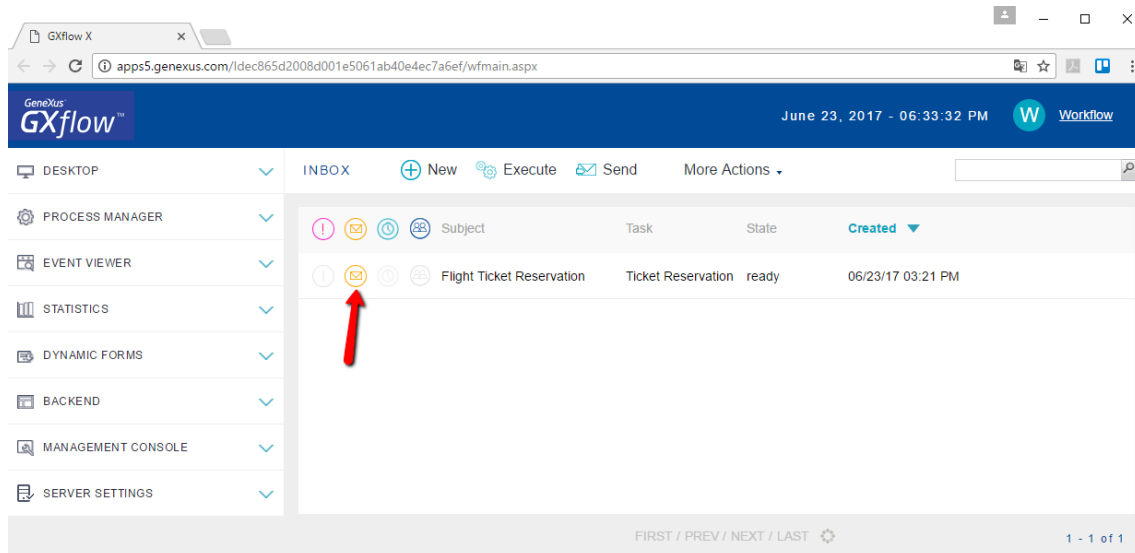
It is divided into 3 sections: at the top is a **header** with the date, time and logged-in user, which in this case is the Workflow administrator, and to the left is an **options menu**. The section in the middle is an **Inbox** where we can view the tasks that are pending and have been assigned to the logged-in user.



Since we haven't defined roles or users yet, we can see that the user assigned by default is the Workflow Administrator, which will be capable of viewing and executing all the processes defined.

At the top, in the incoming box, we can see several buttons for executing actions on the tasks.

When we concentrate on the contents of the incoming box we find a closed envelope, a Subject column that shows: FlightTicketReservation, the name of our process and, in the Task column is the TicketReservation task. The closed envelope indicates that this task is pending to be executed.



To execute the TicketReservation task we select it and double click on it, or otherwise press the Execute button ...

As we expected, we can see that the Reservation transaction opens up for us to enter the reservation.

We leave the identifier with no value because it is autonumbered, and select today's date, then type value 1 to indicate that the reservation is for a passenger who is not one of the agency's customers, so we do not know the passenger's customer identifier, and that's why we leave it at zero.

We will also leave empty the check box indicating whether the reservation is available or not because this will be evaluated further ahead.

We press confirm and will see that the record was added correctly. Then we press "X" to close the form.

In the inbox we will see that the closed envelope icon has disappeared and the task status has changed to "in process".

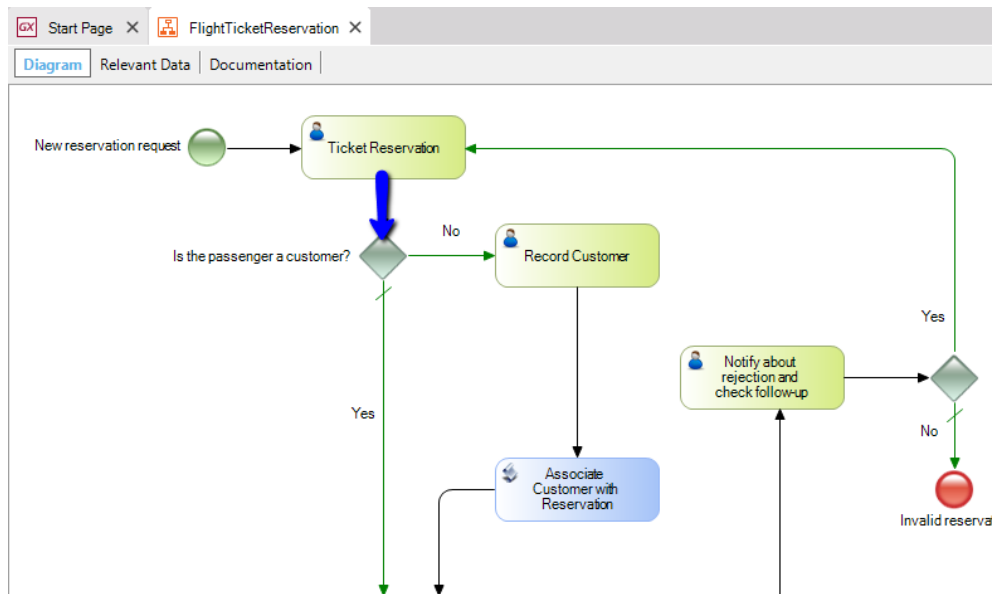
Subject	Task	State	Created
Flight Ticket Reservation	Ticket Reservation	in process	06/23/17 03:21 PM

This indicates that the task is ready to be sent, so we press Send.

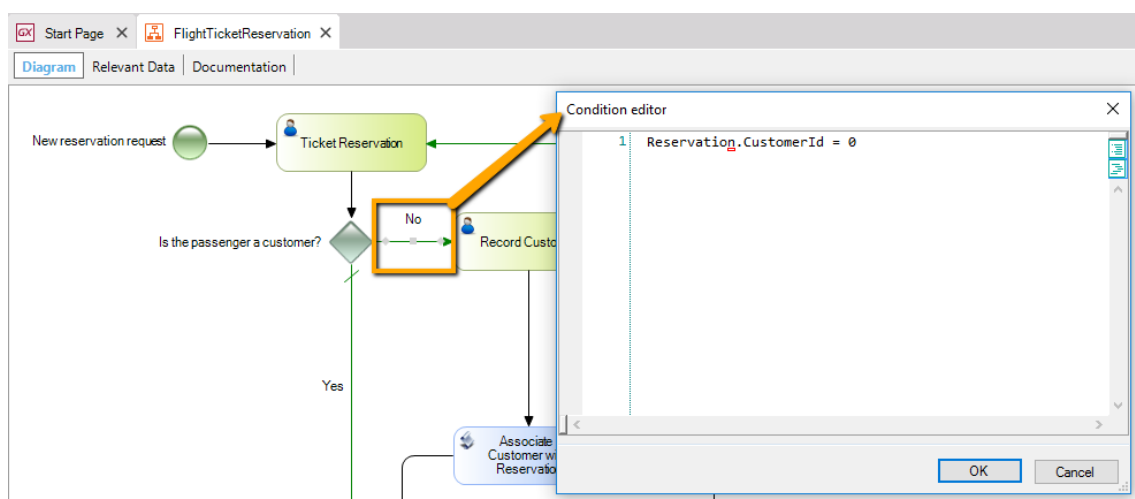
We can see that the task that is now pending for execution is RecordCustomer.

Subject	Task	State	Created
Flight Ticket Reservation	Record Customer	ready	06/23/17 03:39 PM

Looking at our process diagram we can see that, upon finalizing the TicketReservation task (where the Reservation transaction was executed), the process flowed to the exclusive Gateway "Is the passenger a customer?".



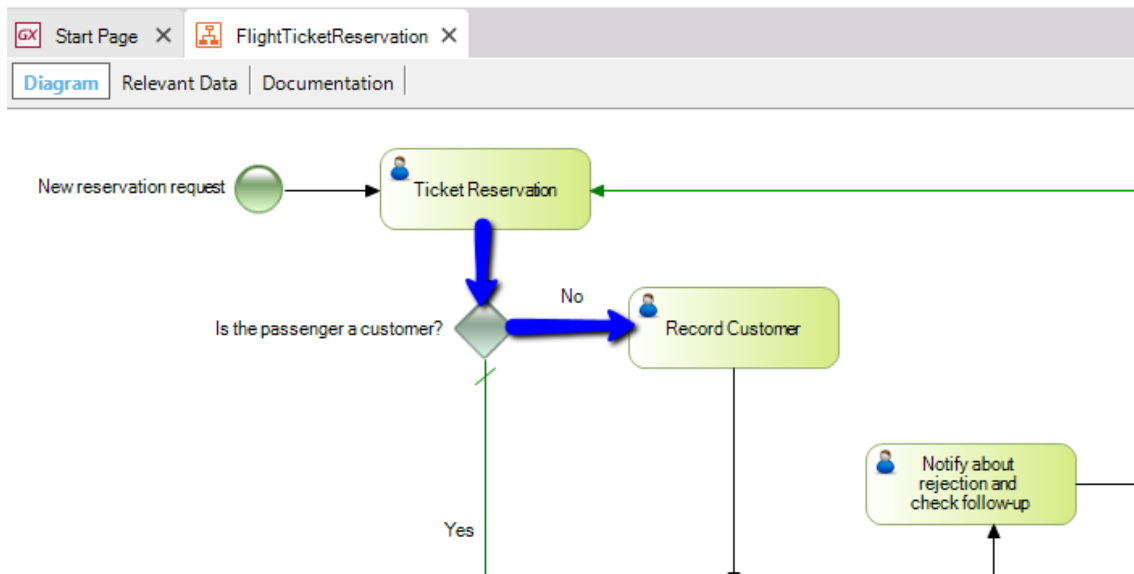
Then the condition we had entered in the connector labeled with No was evaluated.



Note that, in the condition, we only make reference to the value of the CustomerId attribute in the reservation, and do not specify the reservation involved.

As said before, **for each primary key in a transaction, one piece of relevant data is created that is mapped with the value of the attribute**. The relevant data will contain the value of the ReservationId attribute that was instanced upon creating the reservation. So, that value will be known throughout the diagram, and our position will always be in that reservation.

Since we left CustomerId with no value, the condition will be fulfilled, and the following task executed will be RecordCustomer, as the incoming tray indicated it.



To execute the task we first select it and then press the Execute button. We can see that the Customer transaction screen opens up for us to enter the individual corresponding to the reservation as a new customer for the travel agency.

The screenshot shows a web application interface with a top navigation bar containing 'Inbox', 'New', 'Execute', 'Send', 'History', and 'More Actions'. Below this is a table with columns 'Subject' and 'Task'. The 'Subject' column contains 'Flight Ticket Reservation' and the 'Task' column contains 'Record Customer'. A modal window titled 'Customer' is open, displaying a form with the following fields: 'Id' (0), 'Name' (John Parker), 'Is ACompany' (checkbox), 'Address' (6500 Main Street, Houston, USA), 'Phone' (555-123456), 'Email' (jparker@gmail.com), and 'Added Date' (08/14/13). At the bottom of the form are three buttons: 'Confirm', 'Cancel', and 'Delete'.

Once we confirm the entry we must associate the customer just created with the reservation we had entered. We will be seeing this in the video entitled Automation, Part 3.