

Prototyping, a process' instances and history

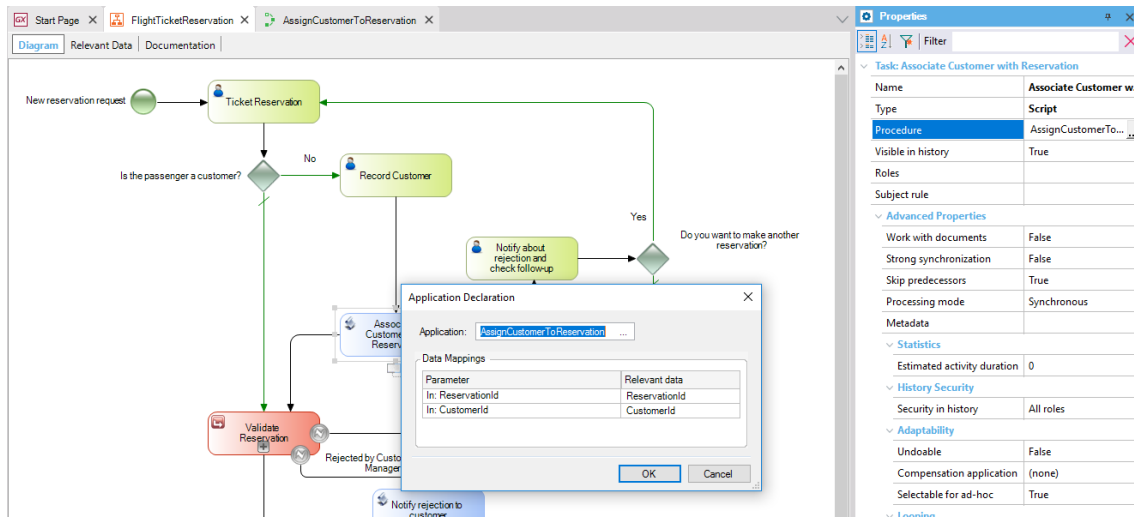
In previous videos on Automation we used objects of the transaction type to record a ticket reservation and create the passenger as a customer for the company. We must now associate the customer we just created with the reservation.

To execute the association we will use the AssignCustomerToReservation procedure which receives by parameter the reservation identifier and the customer identifier. In the source it uses a For Each so that, in the position of the reservation it associates it with the customer desired.



```
1 Parm(in:&ReservationId, in:&CustomerId);
2 |
Subroutines
1 For each Reservation
2   Where ReservationId = &ReservationId
3   CustomerId = &CustomerId
4 Endfor
5
6
```

To assign the procedure to the AssociateCustomerWithReservation task, in its Procedure property we select the AssignCustomerToReservation procedure.



We also map the relevant data ReservationId and CustomerId. Then press OK.

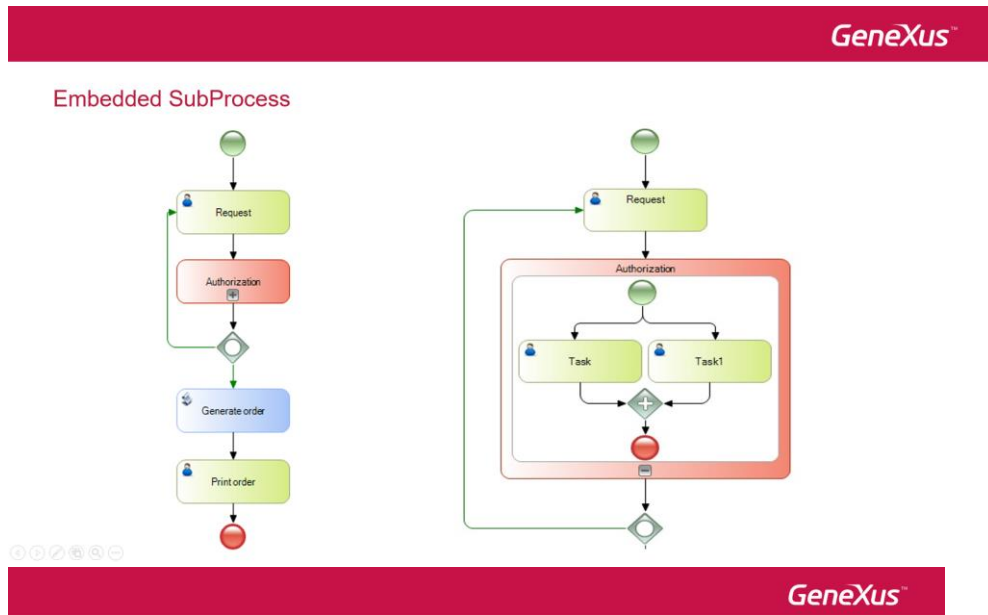
Back in our diagram we can see that following the execution of the AssociateCustomerWithReservation task, or also in the case where adding the passenger as

customer was not necessary, we must validate the reservation by means of the ValidateReservation subprocess.

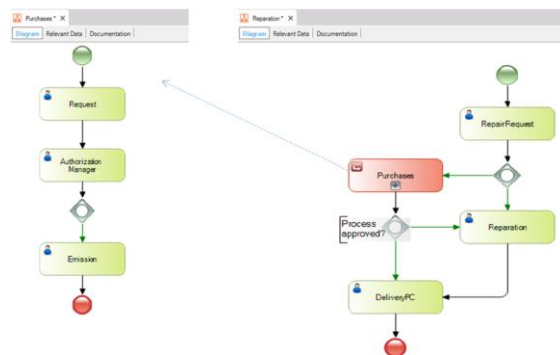
In a close look at the symbol of the subprocess we selected in the modeling stage, if we compare it to those in the Toolbar, we will see that we used a subprocess of the embedded type.

We may use one of either two types of subprocesses: embedded or reusable.

Embedded subprocesses are used when we want to encapsulate several tasks in a complex diagram in order to improve comprehension.



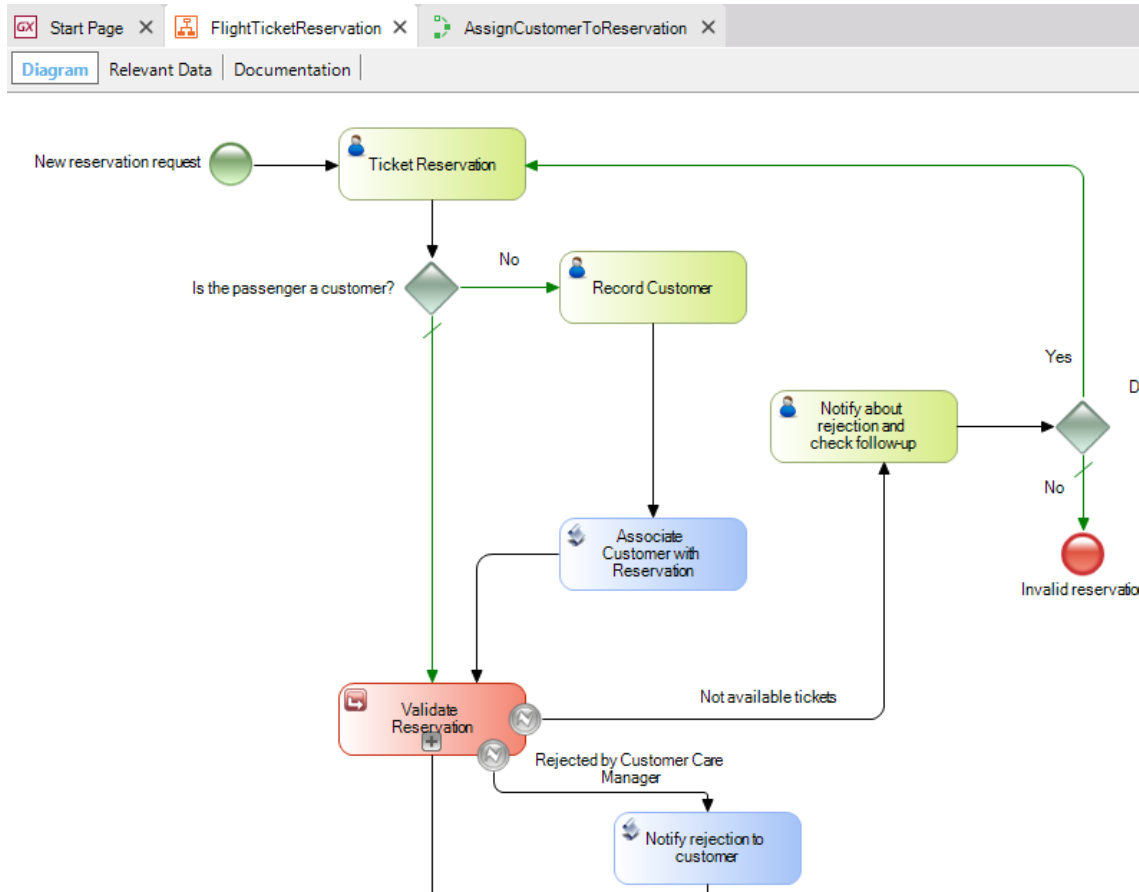
Reusable SubProcess



Reusable subprocesses enable us to invoke an independent process and the possibility of reusing its functionality in the calling process.

Since they are independent processes, their relevant data is also independent, though there is a mechanism for associating them with one another.

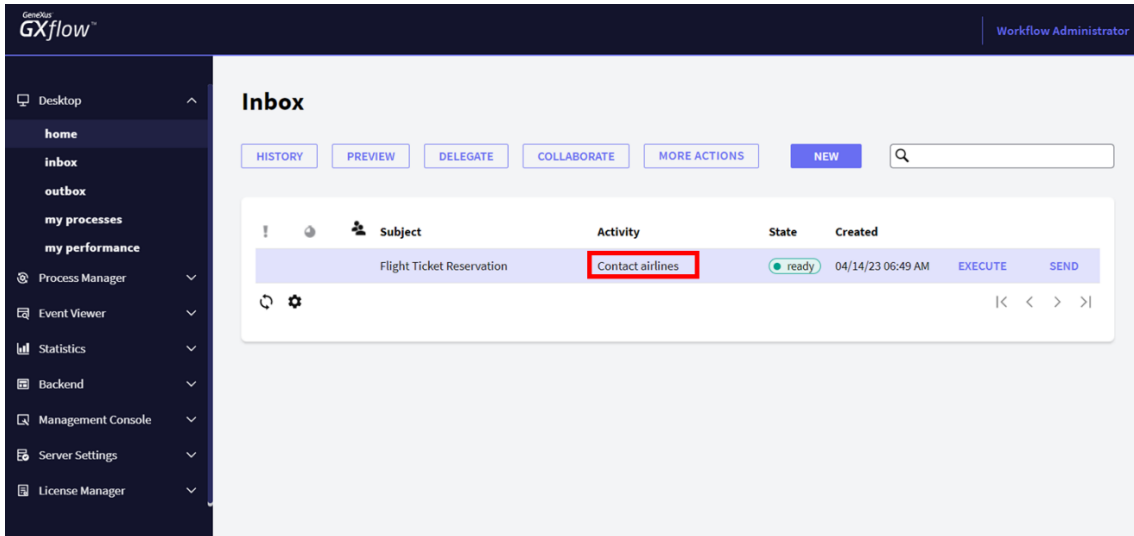
Having defined the ValidateReservation process in our example as an independent process diagram object, then we must change the subprocess symbol we used to a different one of the reusable type. So we go back to GeneXus to modify the FlightTicketReservation diagram... Then delete the embedded subprocess, drag a subprocess of the reusable type and restore the connections. Then we associate the subprocess with the ValidateReservation process.



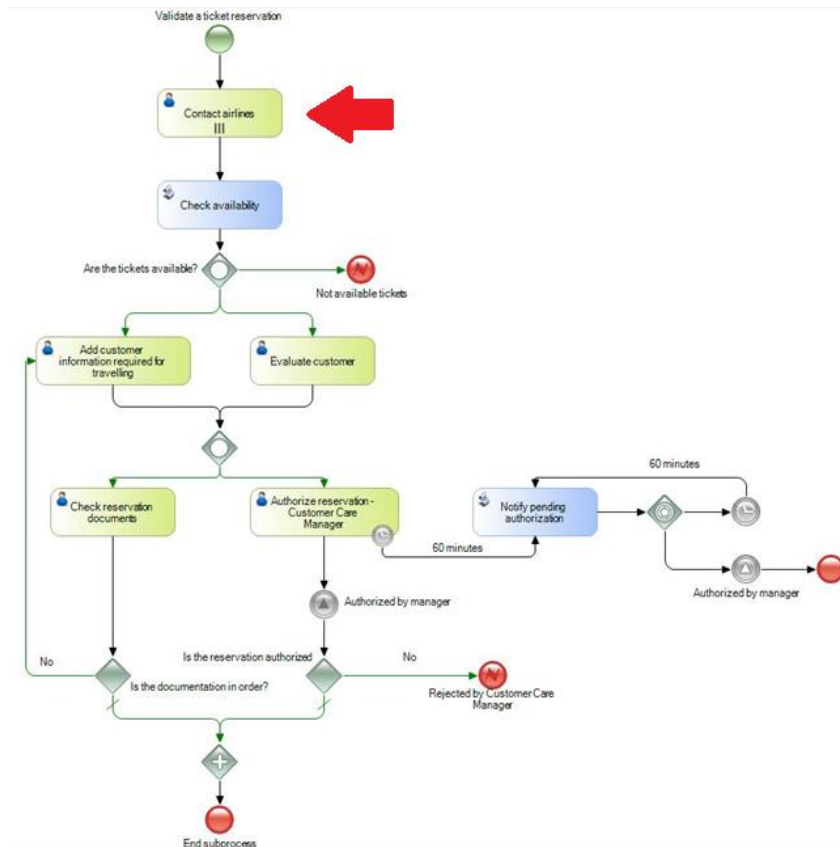
This type of adjustments to the model may be done in the automation stage, because in the modeling stage we are not concerned with which will be the GeneXus objects we associate with the diagram.

We will now execute the process again to include these changes. We select the FlightTicketReservation object, right click and select Run.

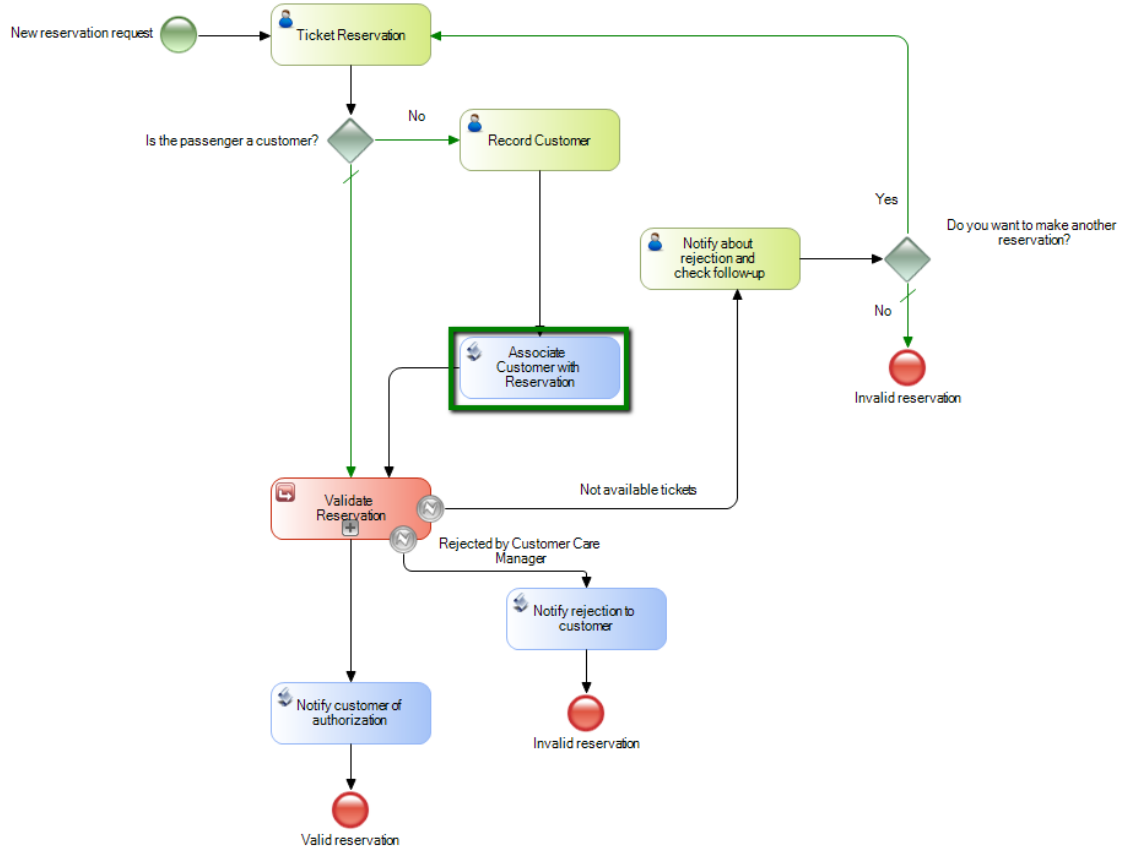
Here, again, we find the incoming tray. We execute the TicketReservation task, we do not enter the customer, then complete this task and execute the following task, RecordCustomer. We enter the customer, confirm and close the window.



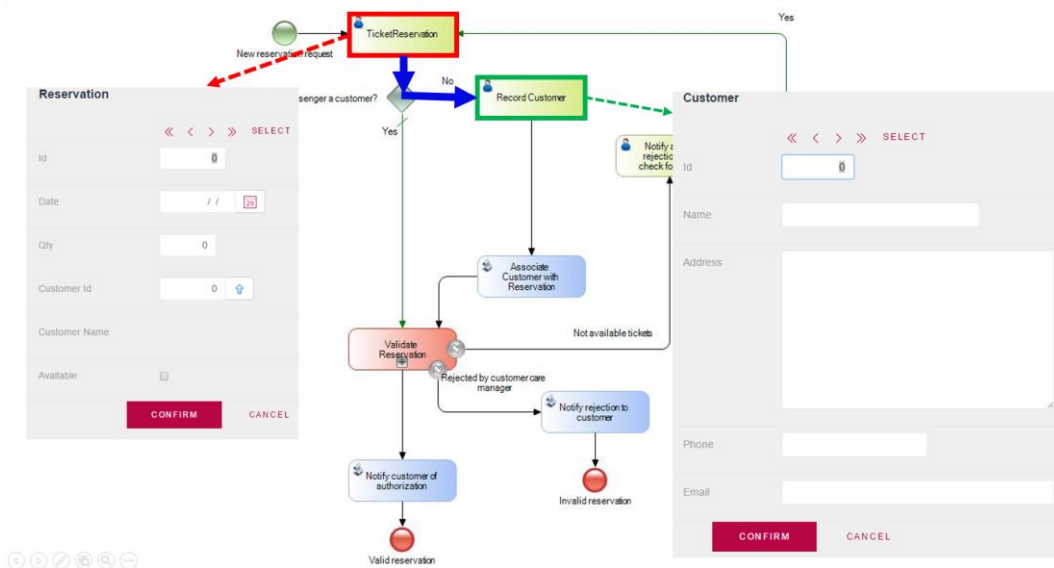
The following task shown by the incoming tray is ContactAirlines, the first task of the ValidateReservation subprocess.



Because it is a batch task, the AssociateCustomerWithReservation task was executed by the Workflow engine, and therefore it is not shown in the incoming tray.

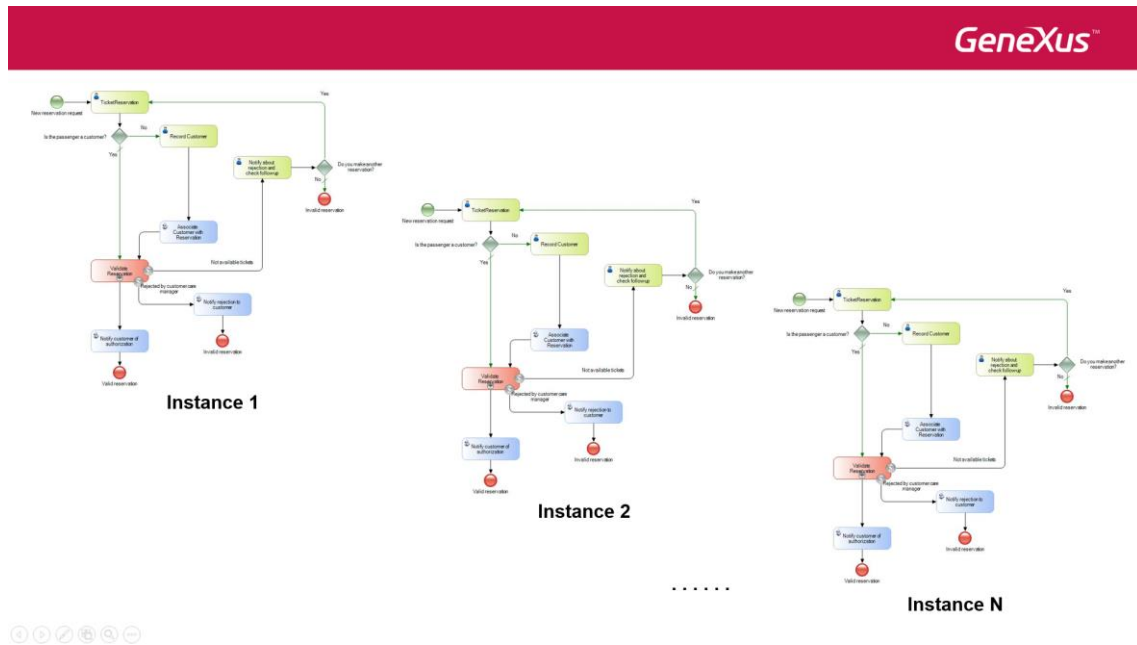


We can see that when GeneXus objects are associated with the diagram, **our system is executed to the extent that the process diagram occurs and each GeneXus object is invoked as applicable.**

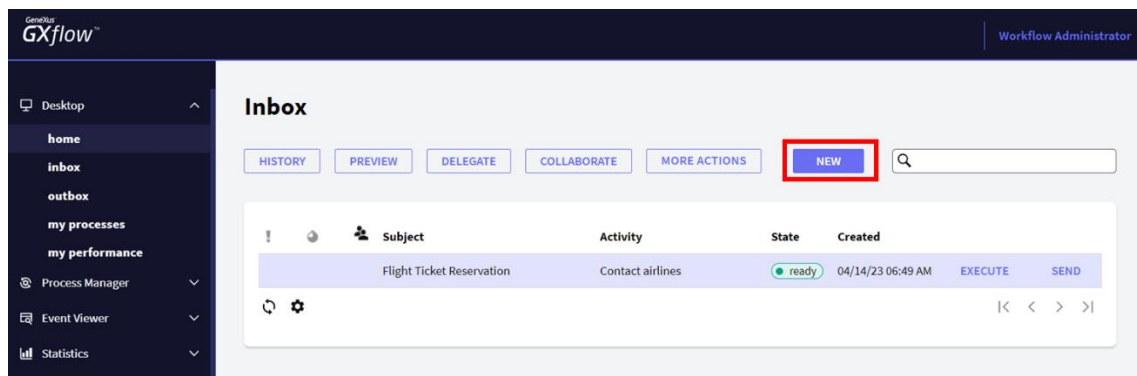


This makes it unnecessary to code link the calls between GeneXus objects -as is the case in an application where workflow is not used- because invocations are automatic.

Something we have not considered so far is the concept that a process may have several **instances** under execution at the same time. This means that it is possible to initiate different "executions" in the same process that take place simultaneously.



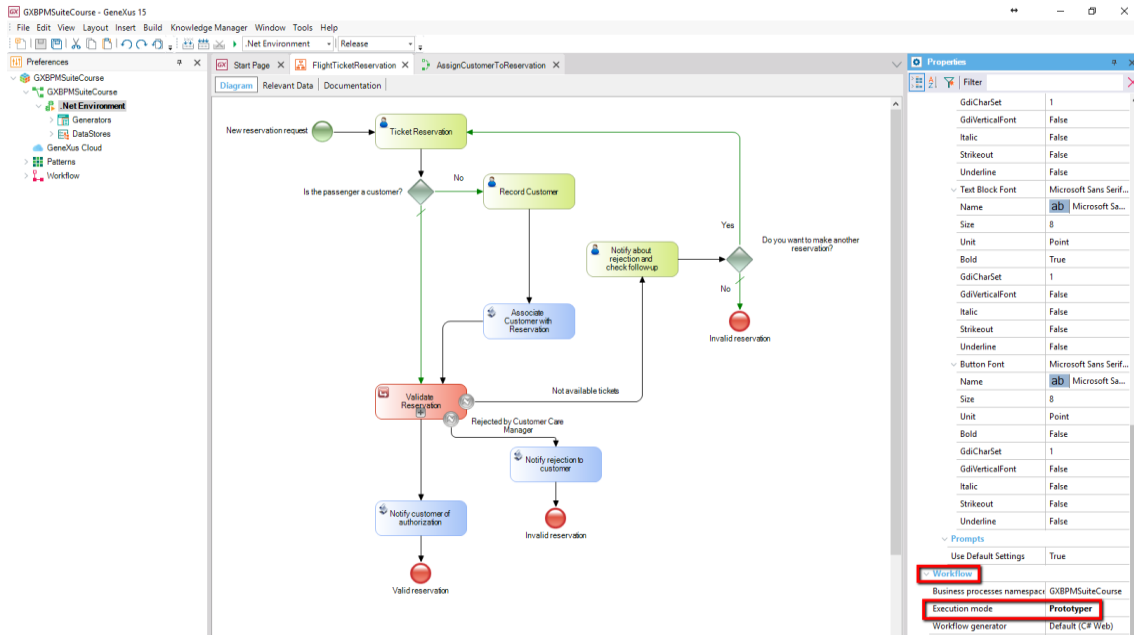
In the incoming tray, the button (New) enables us to initiate a process instance.



This is useful for testing different scenarios for executing the process or the task, a typical aspect in a **prototyping cycle**, where we carry out tests to verify the functioning of our system.

In GeneXus we can define whether we want to work with Workflow in prototype mode or in standard client mode, that is, executing the application as if it were in production.

This is something we do in Preferences, in the properties corresponding to Environment, under Workflow, in the Execution Mode property.



The main difference between the two modes is that in **prototyper mode** we can execute any task with no restrictions on roles or permits, and only the instance initiated will be executed while all previous instances will be aborted.

In **standard mode** we can see all instances in execution but we must log in to execute the workflow client, so we can only execute the tasks assigned to the role logged. We will be seeing this topic further ahead.

Another interesting feature in Workflow client is the possibility to view the process history with the paths in the diagram that were used when the process was executed.

To view this history we select Outbox, and there we select the outgoing tray. We will see that the tasks we have just executed (TicketReservation and Record Customer) are shown.

Subject	Activity	State	Created	Ended	
Flight Ticket Reservation	Record Customer	completed	04/14/23 06:49 AM	04/14/23 06:49 AM	VIEW HISTORY
Flight Ticket Reservation	TicketReservation	completed	04/14/23 06:48 AM	04/14/23 06:49 AM	VIEW HISTORY

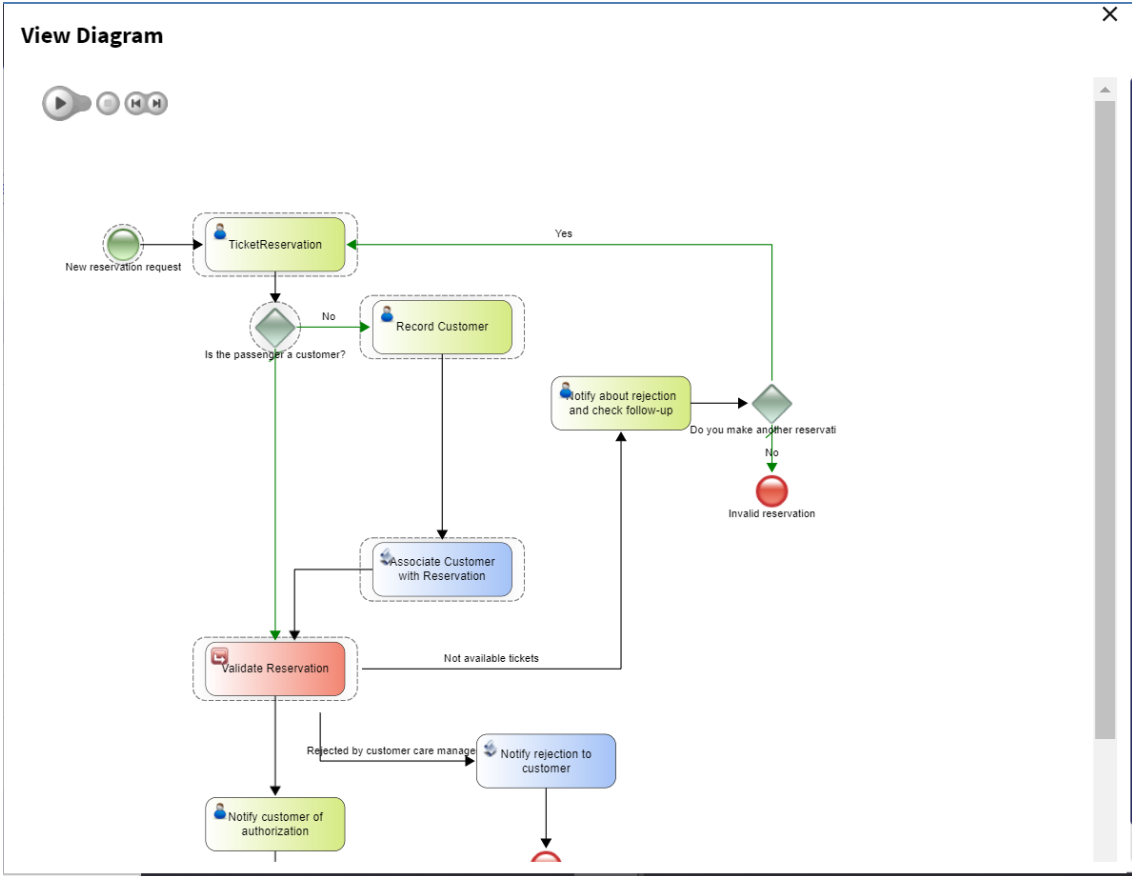
We select one of these tasks, press the View History button and we will see that a window opens up showing the process history.

History

Activity	State	User	Created	Ended		
Validate Reservation	● in process	Workflow Engine	04/14/23 06:49 AM	// 12:00 AM	CANDIDATES	VIEW DIAGRAM
Associate Customer with Reservation	● completed	Workflow Engine	04/14/23 06:49 AM	04/14/23 06:49 AM	CANDIDATES	VIEW DIAGRAM
Record Customer	● completed	Workflow Administrator	04/14/23 06:49 AM	04/14/23 06:49 AM	CANDIDATES	VIEW DIAGRAM
Is the passenger a customer?	● completed	Workflow Engine	04/14/23 06:49 AM	04/14/23 06:49 AM	CANDIDATES	VIEW DIAGRAM
TicketReservation	● completed	Workflow Administrator	04/14/23 06:48 AM	04/14/23 06:49 AM	CANDIDATES	VIEW DIAGRAM
New reservation request	● completed	Workflow Engine	04/14/23 06:48 AM	04/14/23 06:48 AM	CANDIDATES	VIEW DIAGRAM

⏮ ⚙
⏪ < > ⏩

In this window we can see all the tasks that have been executed. If we go to, View Diagram, we can view the history in animation mode by pressing Play.



We can see that the execution stopped in the ValidateReservation subprocess.

In the next video we will continue associating GeneXus objects in that subprocess diagram.