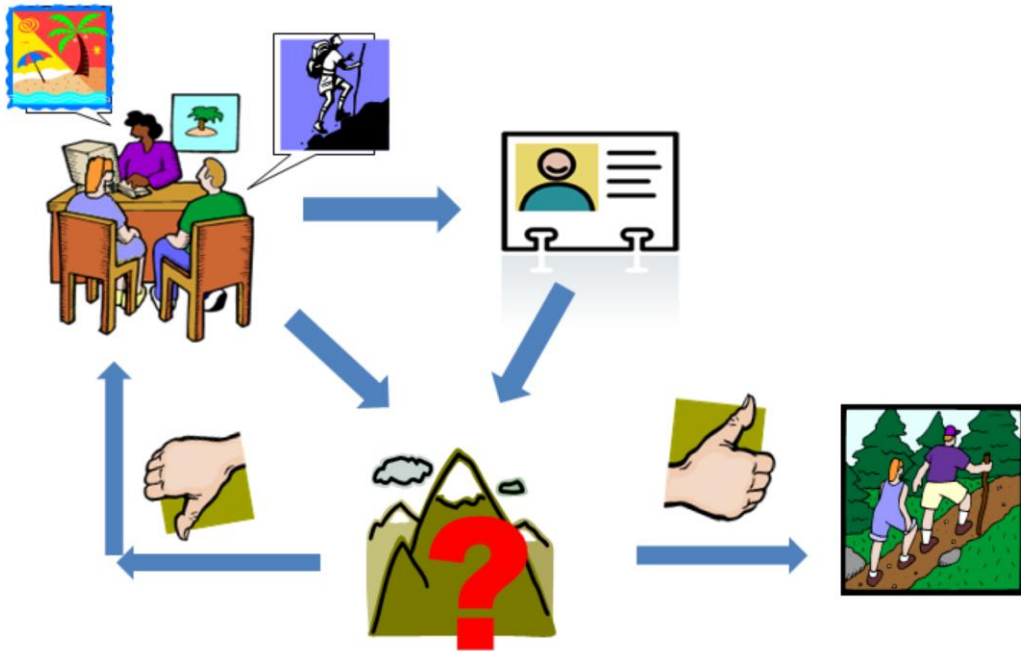


Administración de procesos de negocio (BPM)

GeneXus 16

Proceso de negocios de reservar un paquete turístico



En la agencia de viajes, hay trámites que requieren de que varias tareas se realicen unas a continuación de otras, en un cierto orden, por ejemplo, cuando se desea hacer una reserva de un paquete turístico para una determinada atracción.

Cuando se ingresa la reserva, hay que verificar que la persona que solicite el paquete sea un cliente de la agencia y en caso de que no lo sea, se deberá ingresar como cliente.

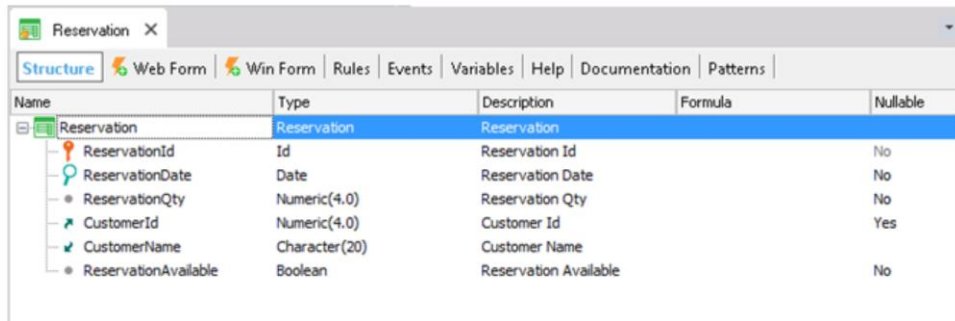
Una vez completo este paso, es necesario verificar que haya paquetes disponibles para la cantidad de personas que deseen realizar el viaje.

Si hay lugares disponibles, la reserva será asignada y en caso contrario, se le debe ofrecer al pasajero otro paquete.

Una secuencia de pasos como la que acabamos de ver, es lo que se denomina proceso de negocio.

En GeneXus podemos definir procesos de negocios como éste.

Creamos una transacción Reservation



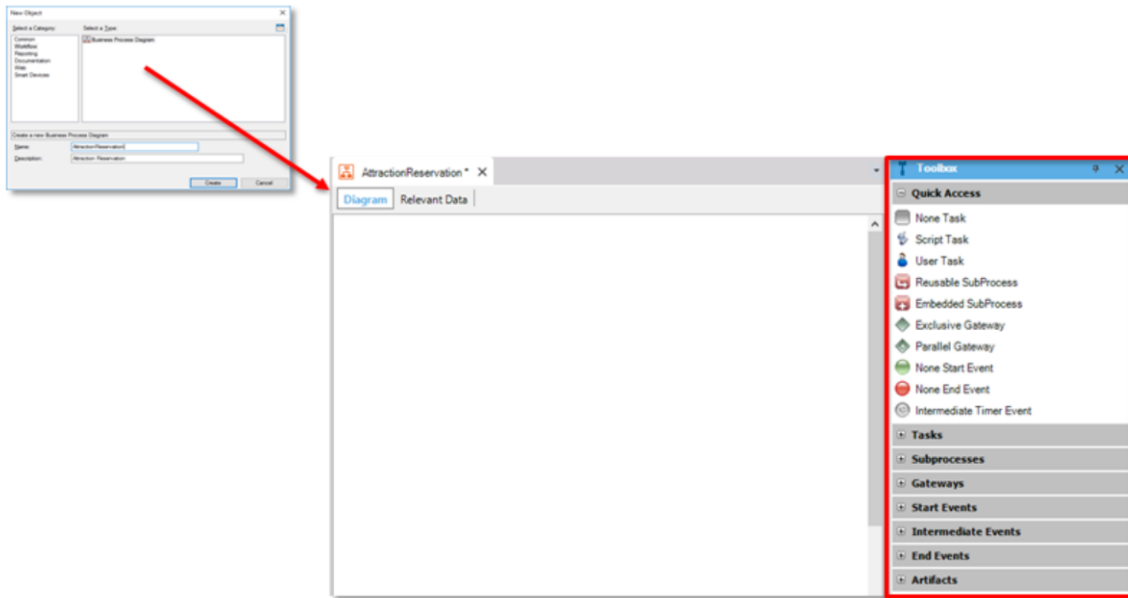
The screenshot shows the 'Structure' view of a transaction named 'Reservation'. The table below represents the data shown in the screenshot.

Name	Type	Description	Formula	Nullable
Reservation	Reservation	Reservation		
ReservationId	Id	Reservation Id		No
ReservationDate	Date	Reservation Date		No
ReservationQty	Numeric(4,0)	Reservation Qty		No
CustomerId	Numeric(4,0)	Customer Id		Yes
CustomerName	Character(20)	Customer Name		
ReservationAvailable	Boolean	Reservation Available		No

Como necesitamos registrar las reservas de paquetes turísticos, lo que hicimos previamente fue crear una transacción de reservas, con atributos para almacenar el identificador, fecha de la reserva, cantidad de personas que usarán el paquete, Identificador y nombre del cliente de la reserva y ReservationAvailable para marcar si el paquete turístico está disponible o no.

Para el atributo CustomerId, seleccionamos su columna Nullable con el valor Yes, para indicar que al momento de ingresar una reserva, podemos no tener aún al identificador de cliente, de la persona que está adquiriendo la reserva.

Para modelar un proceso de negocios en GeneXus creamos un objeto Business Process Diagram



Ahora vamos a crear un objeto del tipo diagrama de procesos de negocios. Hacemos ...File...New... y elegimos un objeto del tipo Business ProcessDiagram. Lo llamamos AttractionReservation.

Comenzamos a modelar el proceso de negocios

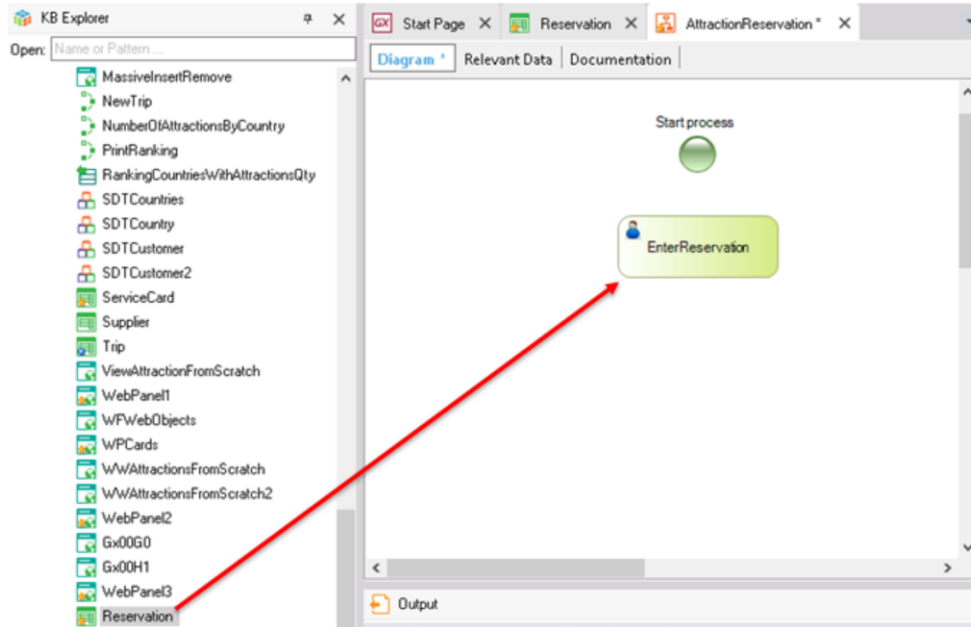


Vemos una página en blanco del diagrama que vamos a construir. Si abrimos la toolbox, vemos que hay una gran cantidad de símbolos que podemos usar.

Para marcar el comienzo de nuestro proceso de negocios, arrastremos un símbolo de NoneStartEvent. Vemos que cuando el símbolo queda seleccionado, aparecen a su alrededor una serie de símbolos que son los que podríamos utilizar para continuar el diagrama. Esta sugerencia nos ahorra bastante tiempo de ir a la barra de herramientas para arrastrar otros símbolos.

Presionamos F2 sobre el None Start Event y escribimos Start process.

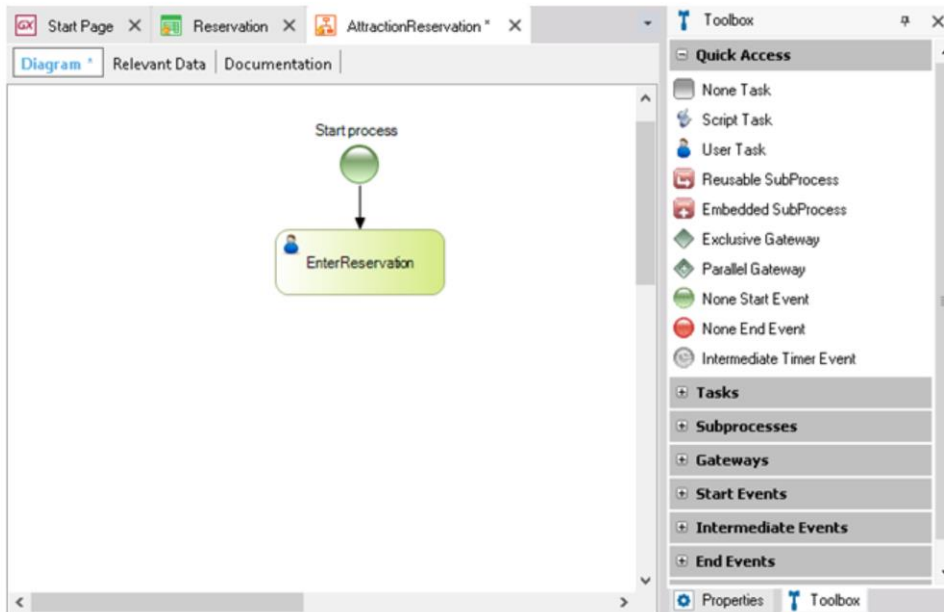
Agregamos la transacción Reservation



Como la primera actividad que vamos a realizar es ingresar una reserva, ubicamos a la transacción Reservation en la ventana del **KB Explorer** y la arrastramos hacia el diagrama.

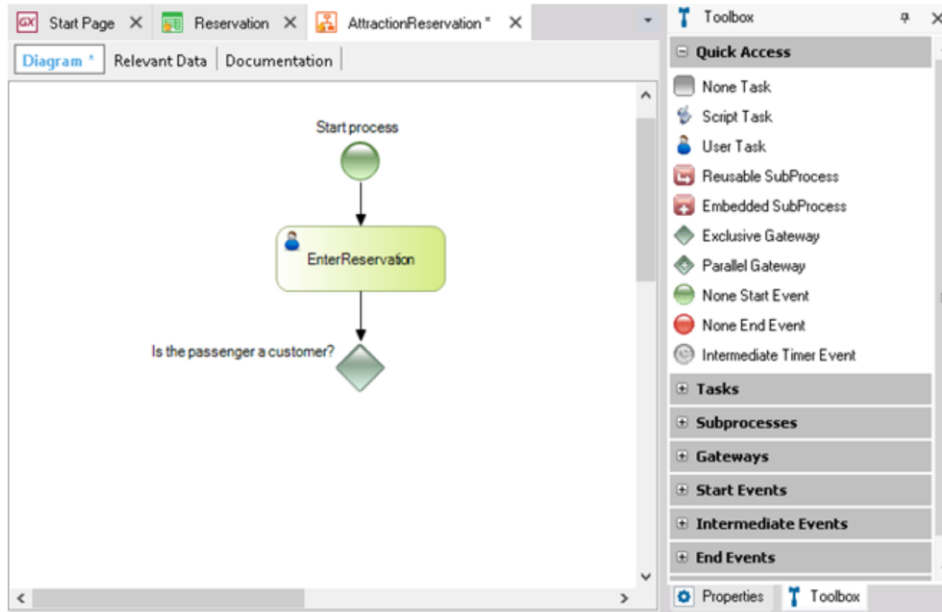
Vemos que se creó un rectángulo verde con el nombre Reservation y que tiene una figura humana en el extremo superior izquierdo. Esto nos indica que se creó una tarea del tipo interactiva, ya que la transacción permite que un usuario ingrese datos mediante la misma. Presionamos F2 y le cambiamos el nombre a EnterReservation.

Conectamos la tarea Reservation



Para conectar el nodo de Start con la transacción, hacemos clic en la parte inferior del círculo verde y sin soltar, arrastramos hasta que la punta de la flecha toque el borde superior del rectángulo de la transacción.

Agregamos un nodo de decisión

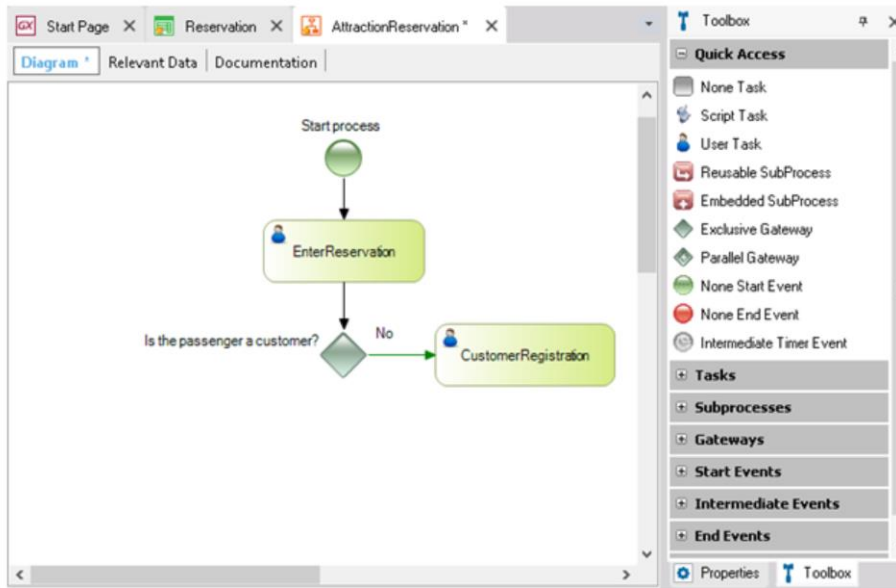


Según el proceso que siguen en la agencia de viajes, el sistema debe controlar si la persona que contrata la reserva, ya es cliente de la empresa o no.

Para representar una decisión en el diagrama, vamos a la toolbar, hacemos clic sobre Gateways, arrastramos un nodo Exclusive Gateway sobre el diagrama y lo unimos desde la transacción Reservation. Con F2 escribimos la pregunta "Is the passenger a customer?" y luego haciendo clic sobre el mismo ubicamos el texto.

Este tipo de nodo evalúa una condición y dependiendo del resultado, hace que el flujo siga hacia abajo (que sería el curso normal) o hacia la derecha del símbolo (que sería el curso alternativo).

Definimos el flujo alternativo

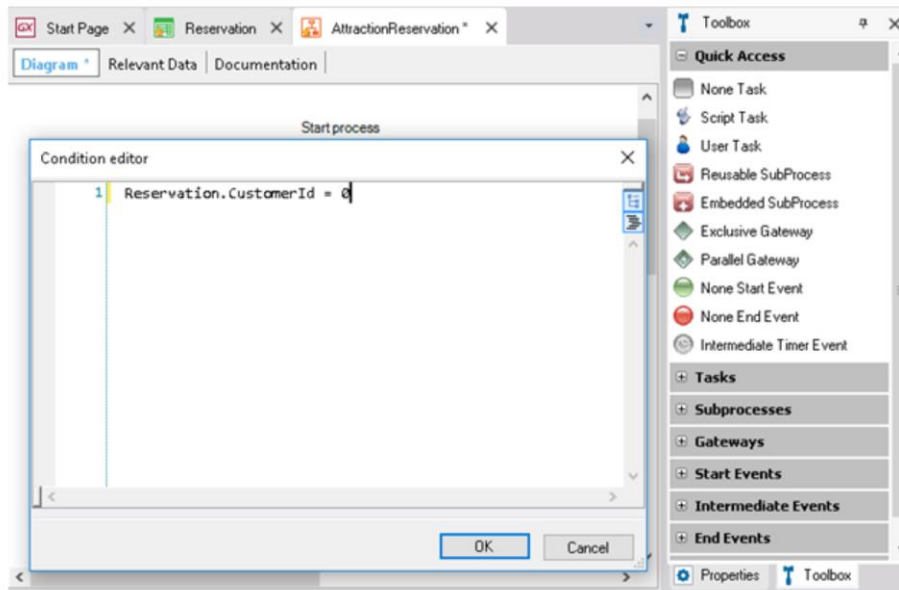


En nuestro ejemplo debemos definir la condición del Gateway que hará que el flujo siga el curso normal o el flujo alternativo, para agregar un cliente.

Es decir, tenemos que hacer que si la persona que desea hacer la reserva del paquete turístico no es cliente de la empresa, se invoque a la transacción Customer, para agregarlo como cliente.

Para comenzar, arrastramos a la transacción Customer al diagrama y la conectamos desde el nodo Exclusive Gateway. Con F2 le cambiamos la descripción de la tarea a CustomerRegistration. Luego seleccionamos la conexión y con F2 le ponemos la etiqueta "No".

Definimos la condición de bifurcación



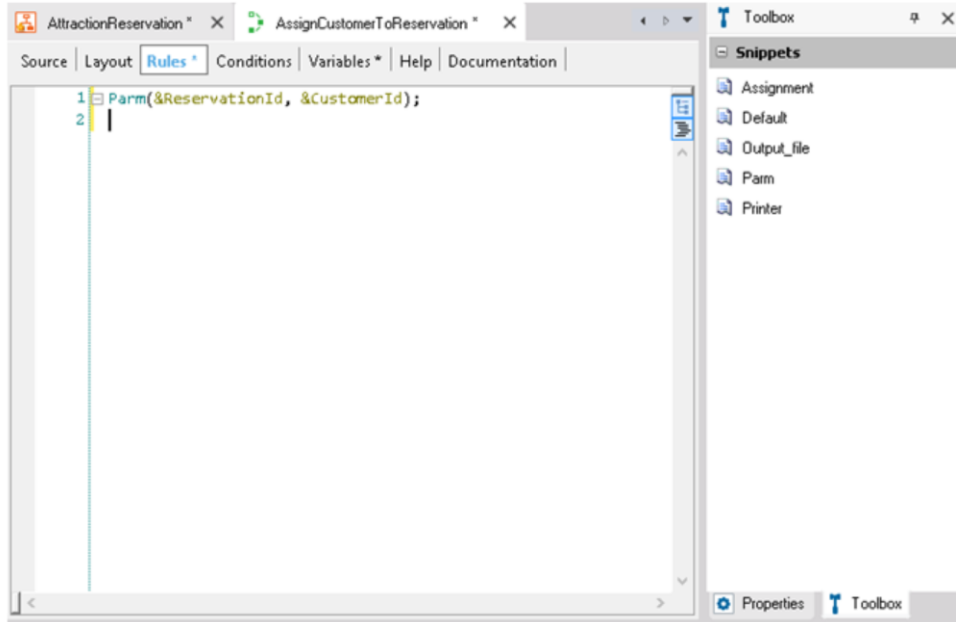
Para completar la definición de la decisión, tenemos que agregar la condición que hará que partiendo del nodo de decisión, el flujo del proceso siga hacia la derecha invocando a la transacción Customer, o siga hacia abajo hacia la siguiente tarea.

El camino hacia la derecha sería en nuestro caso el flujo alternativo, mientras que el camino hacia abajo sería al flujo habitual, es decir, cuando el que solicita la reserva ya es cliente de la agencia de viajes.

Para definir la condición que provoque la bifurcación, hacemos doble clic en la flecha verde que une el Gateway con la transacción Customer y vemos que se abre una ventana del editor de condiciones.

Escribimos la siguiente expresión: `Reservation.CustomerId = 0`, para indicar que el flujo debe tomar ese camino si cuando se ingresó la reserva, el atributo CustomerId se dejó con valor 0.

Asociamos el cliente a la reserva



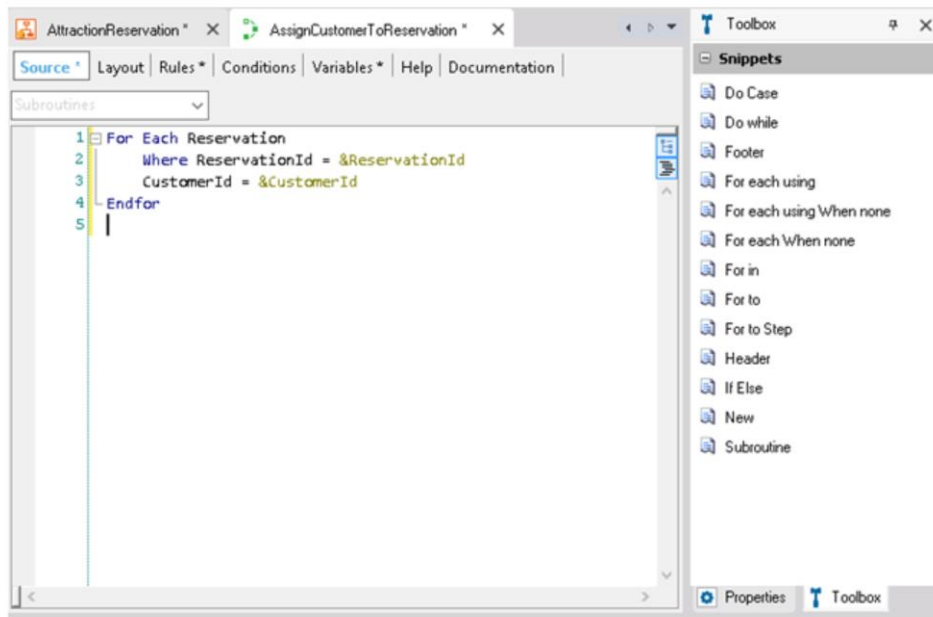
Si se sigue el flujo alternativo, vemos que se abrirá la transacción de Customer para que agreguemos a la persona de la reserva, como cliente.

Una vez que confirmamos el ingreso, debemos asociar el cliente recién creado a la reserva.

Para eso, debemos crear un procedimiento... le llamamos AssignCustomerToReservation y en la sección de reglas escribimos una regla parm, con los parámetros &ReservationId y &CustomerId...

Estas variables almacenan al identificador de la reserva que creamos al principio y al identificador del cliente que queremos asociarle a dicha reserva.

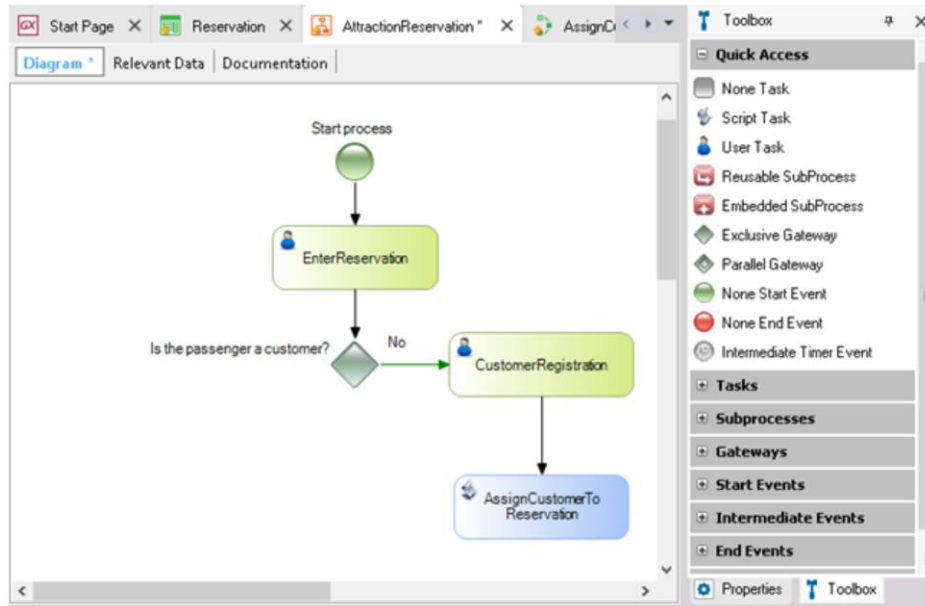
Asociamos el cliente a la reserva



En la source escribimos...For each... WhereReservationId igual ...a la variable ReservationId que recibimos por parámetro. Luego asignamos a CustomerId el valor de la variable &CustomerId y cerramos el For Each.

De esta forma le asignamos el cliente recién creado, a la reserva ingresada previamente.

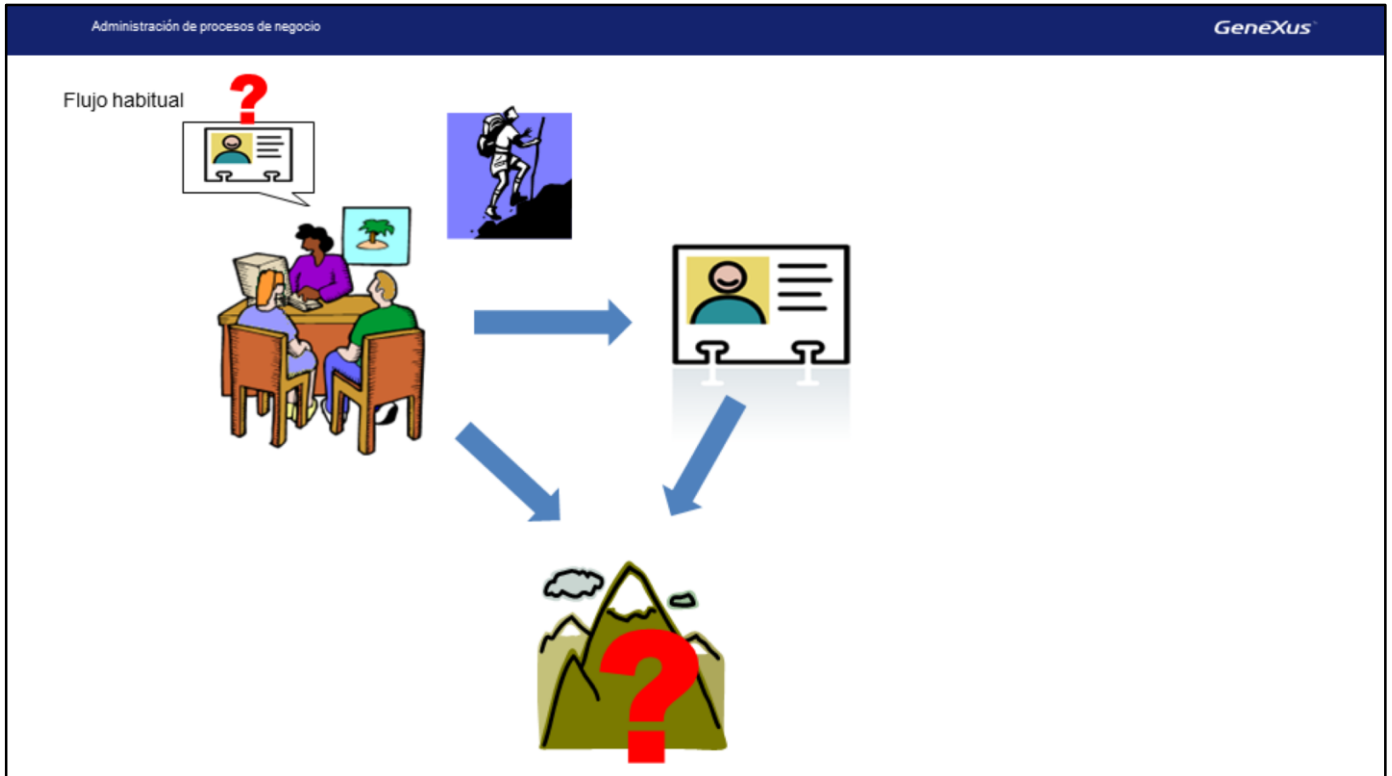
Asociamos el cliente a la reserva



Ahora arrastramos el procedimiento recién creado hacia el diagrama. Vemos que se nos creó un rectángulo color azul, que significa que la tarea que se creó es del tipo no interactiva, también llamada tipo batch.

Esto es así porque el procedimiento se ejecuta sin intervención de una persona, es decir que no hay interacción con el usuario ingresando datos, como en el caso de una transacción.

Por último, conectamos al procedimiento AssignCustomerToReservation desde la transacción Customer....

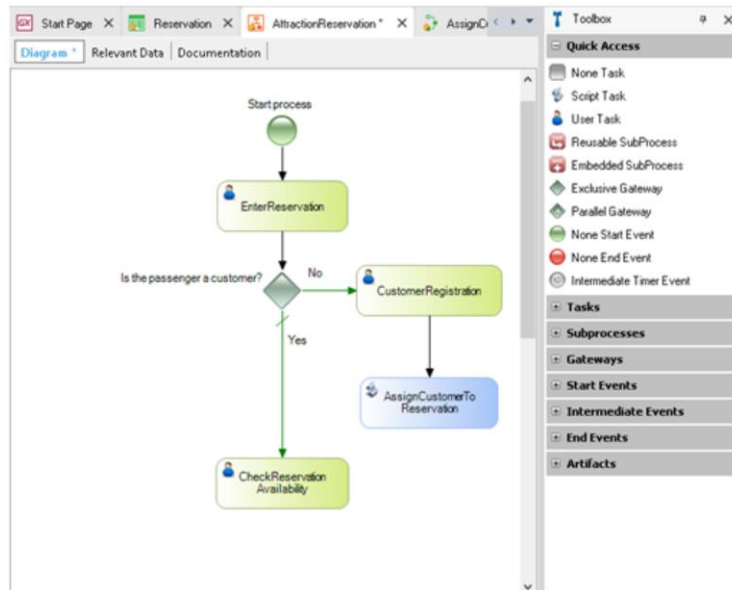


Hasta ahora hemos descrito las tareas que se realizarán si se ejecuta el flujo alternativo, es decir, si hay que agregar un cliente nuevo y asignarlo a la reserva.

Ahora agregamos la actividad que se realizará si el flujo es el habitual, es decir, si no es necesario invocar a la transacción cliente.

En este punto, el empleado de la agencia debería verificar la disponibilidad de la reserva (por ejemplo viendo si hay vuelos disponibles, si quedan cupos en el paquete solicitado para las fechas previstas, etc.) y finalmente indicarle al sistema si la reserva puede completarse o no.

Definimos el flujo habitual



Para ingresar este dato, debemos contar con una pantalla donde podamos confirmar o cancelar la reserva. Para eso, podemos abrir nuevamente la transacción Reservation y marcar a la reserva como disponible o no, mediante el atributo ReservationAvailable.

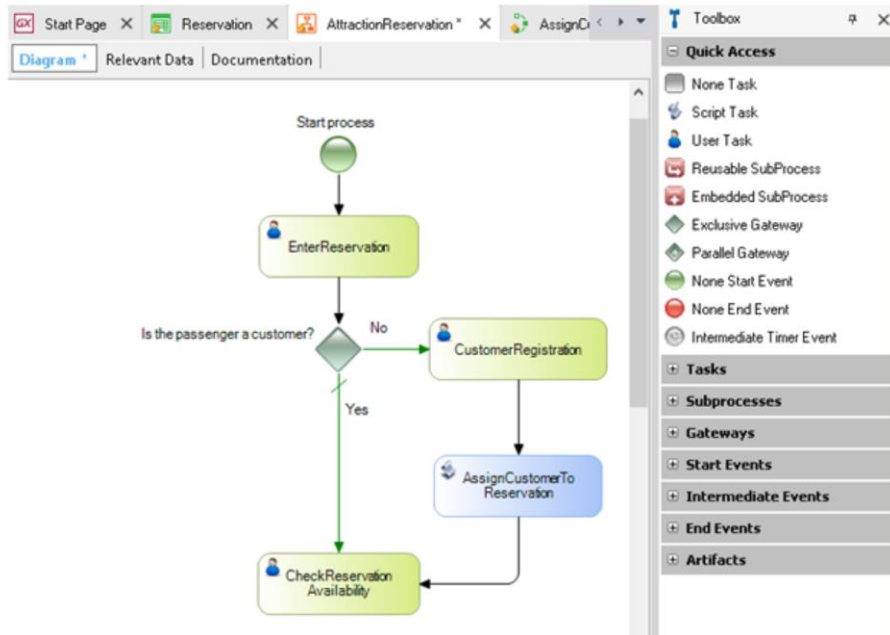
Para esto, ubicamos a la transacción Reservation en la ventana de **KB Explorer**, la arrastramos hacia el diagrama y la conectamos desde el Gateway.

Vemos que se creó una actividad con el nombre Reservation1, así que presionamos F2 y cambiamos su nombre a CheckReservationAvailability. Notemos que aquí no le estamos cambiando el nombre a la transacción, sino a la actividad del diagrama que invoca a la transacción Reservation.

Cuando insertamos el Gateway que nos permitía tomar la decisión, dijimos que el flujo hacia la derecha que iba a la tarea CustomerRegistration era el flujo alternativo y el flujo que saldría hacia abajo era el flujo normal, es decir cuando la persona que hacía la reserva ya era cliente de la agencia.

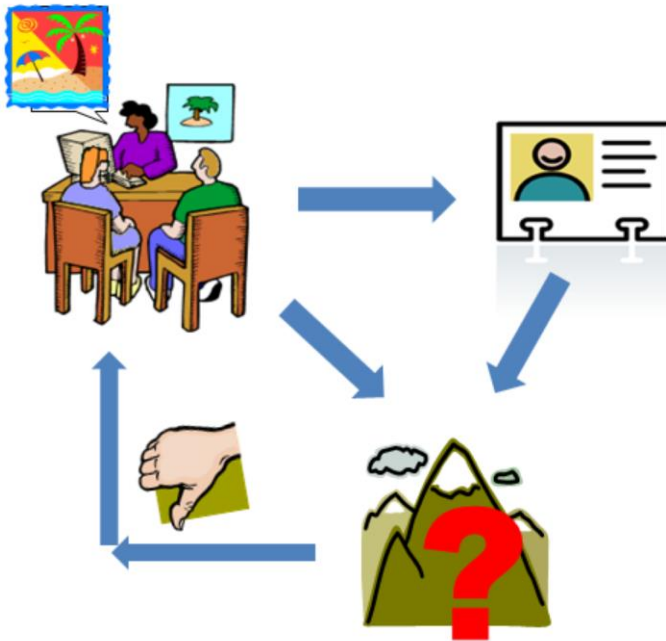
Para indicar que el flujo hacia abajo es el flujo habitual, seleccionamos la conexión y en la ventana Propiedades ponemos su propiedad ConditionType en el valor Default. Observemos en el diagrama que el flujo quedó señalizado con una raya de color verde que cruza al mismo. Presionamos F2 y lo etiquetamos con el texto "Yes".

Conectamos el fin del flujo alternativo



Aprovechamos ahora para conectar a la tarea AssignCustomerToReservation a la tarea CheckReservationAvailability, ya que una vez completada la asignación del cliente, se debe continuar también con la verificación de la disponibilidad de la reserva.

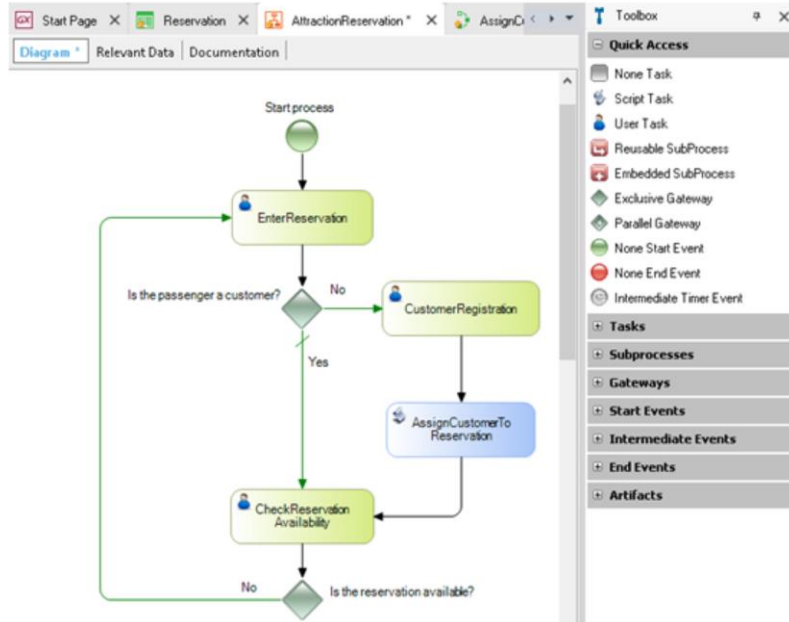
La reserva no está disponible



Una vez que finaliza la tarea `CheckReservationAvailability`, debemos evaluar el valor que le hayamos puesto al atributo `ReservationAvailable`.

Si el checkbox se dejó desmarcado, significa que la reserva no estará disponible y se debe ofrecer al cliente un nuevo paquete para reservar.

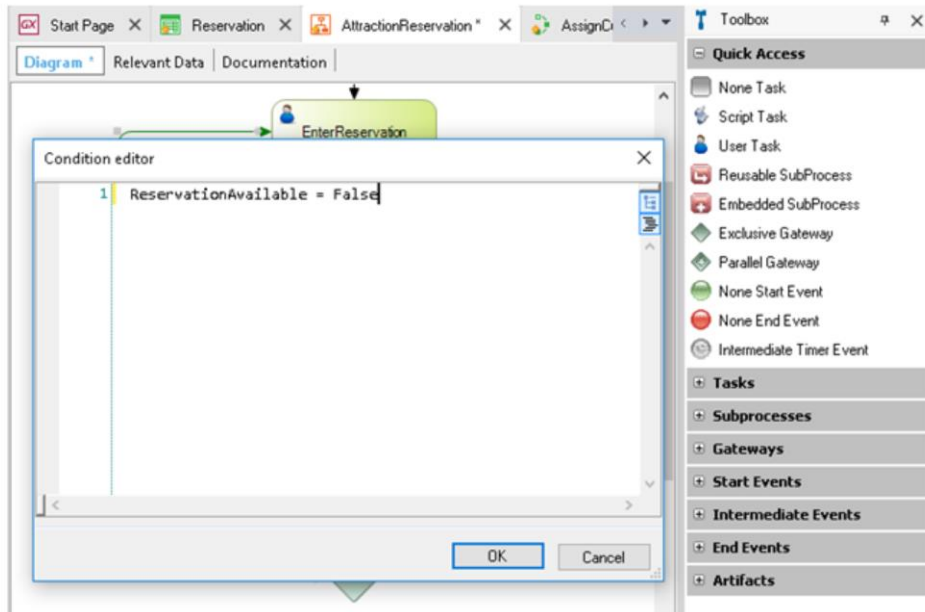
La reserva no está disponible



Para reflejar esto, insertamos desde la toolbar un Exclusive Gateway y lo conectamos desde la tarea CheckReservationAvailability. Cambiamos su descripción a "Is the reservation available?".

Luego conectamos el flujo alternativo del mismo (que lo dibujamos hacia la izquierda) a la tarea EnterReservation, para que se pueda agregar una reserva nueva. Etiquetamos el conector con "No".

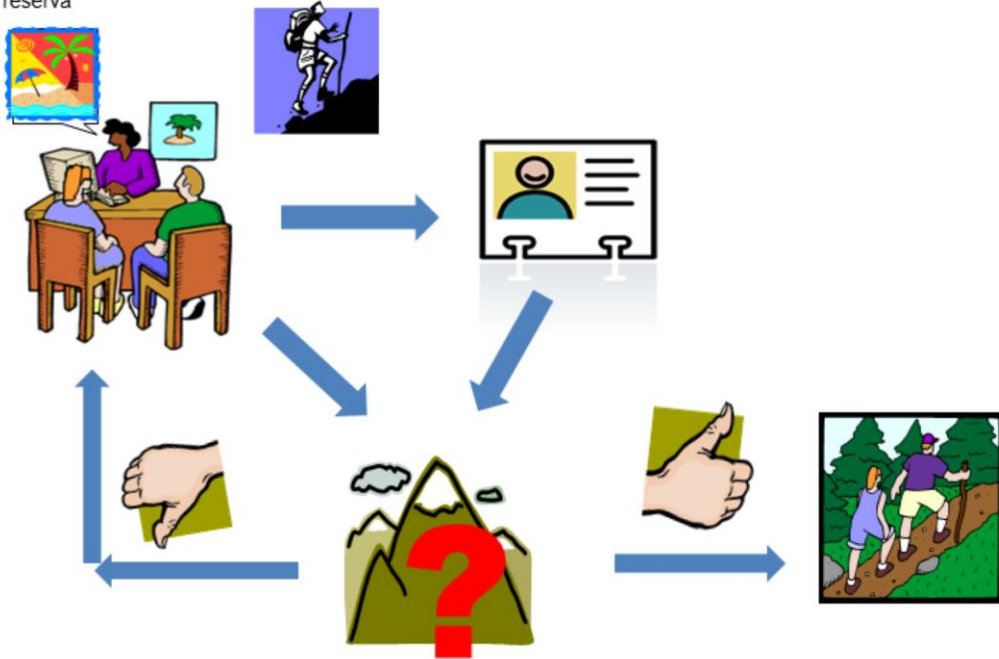
Definimos la condición de reserva no disponible



Ahora debemos ingresar la condición que debe darse para que el flujo alternativo se cumpla, es decir cuando la reserva no está disponible.

Para indicar esto en el diagrama, hacemos doble clic sobre la conexión que sale hacia la izquierda y escribimos ReservationAvailable=False.

Se confirma la reserva

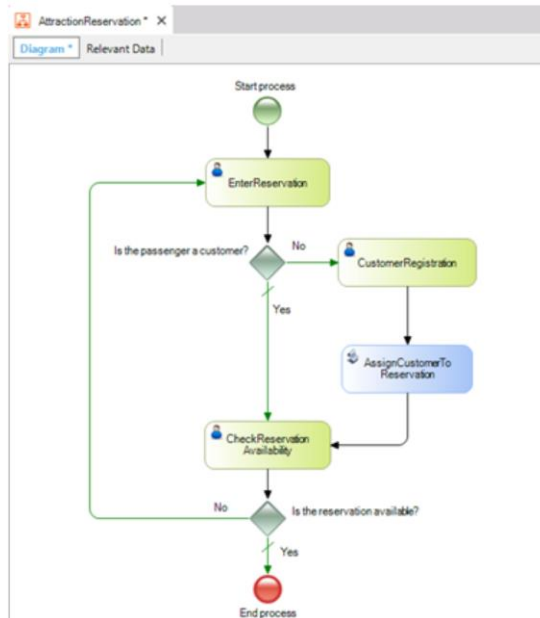


Hasta aquí analizamos todo lo que sucede en el caso de que hubiera que ofrecer otra alternativa de viaje al cliente, ya que la reserva no se había podido confirmar.

Lo que nos falta hacer ahora es tomar en cuenta el caso de que la reserva se confirme.

En este caso ya no habrá más tareas y deberá finalizar el proceso.

Definimos el fin del proceso



Para indicar que queremos terminar el diagrama, insertamos desde la toolbar un símbolo de NoneEndEvent y lo conectamos desde el Gateway. Le asignamos el texto "End process".

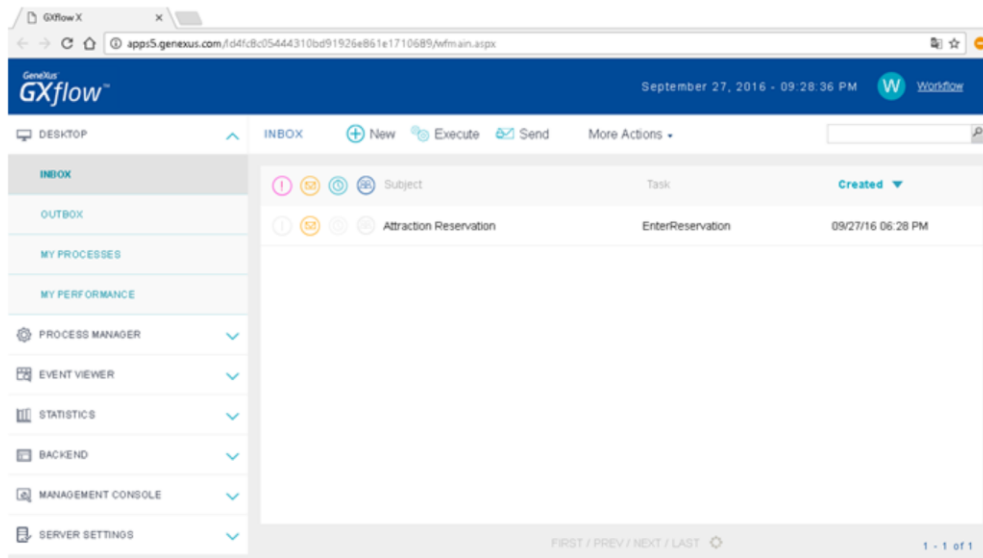
Esta conexión hacia abajo es el flujo normal del Gateway, de modo que cuando la reserva se confirma, finalizará el proceso.

Para indicar esto, seleccionamos la conexión y en la ventana Propiedades ponemos su propiedad ConditionType en el valor Default. Finalmente le asignamos a la conexión el texto "Yes".

Hasta aquí hemos finalizado el diagrama de nuestro proceso de negocio.

Para probar el funcionamiento del mismo, vamos a ejecutarlo.

Ejecutamos el diagrama



Sobre la solapa con el nombre del diagrama, damos botón derecho y elegimos Run...

En el análisis de impacto GeneXus nos informa que se va a crear la tabla Reservation, lo cual es lo esperado, así que procedemos a reorganizar y a continuar con la ejecución.

Vemos que se nos abre una pantalla con el Cliente GXflow, que se parece a una bandeja de entrada de e-mails, donde se muestra para cada usuario, las tareas que tiene pendiente de realizar, según el proceso definido.

Si observamos en esta bandeja de entrada, vemos un sobre cerrado y que en la columna Subject dice: AttractionReservation, indicándonos que tenemos definido ese proceso y en la columna Taks, tenemos pendiente la tarea EnterReservation.

Para ejecutarla, seleccionamos la misma y presionamos el botón de Execute, o bien hacemos doble clic sobre la tarea...

Ejecutamos la tarea Reservation

The screenshot shows the GeneXus GXflow interface with a 'Reservation' modal form. The form contains the following fields and values:

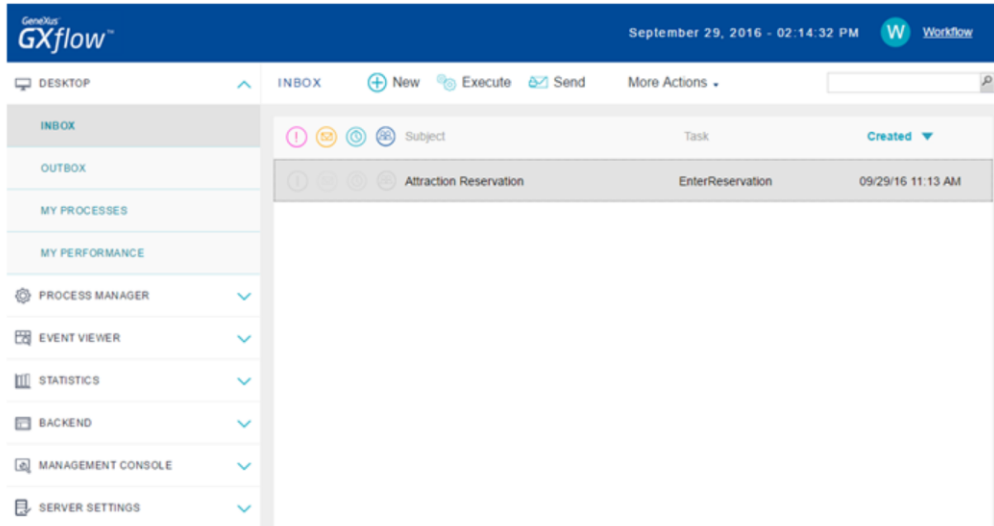
- Id:** Empty text input field.
- Date:** Date picker showing 09/29.
- Qty:** Text input field containing '0'.
- Customer Id:** Text input field containing '0' with a dropdown arrow.
- Customer Name:** Empty text input field.
- Available:** Text input field with a small icon.

At the bottom of the modal, there are two buttons: 'CONFIRM' (highlighted in red) and 'CANCEL'.

Se ejecuta la transacción Reservation, que es el objeto GeneXus que quedó asociado a la tarea EnterReservation, ya que para crear esta tarea arrastramos la transacción sobre el diagrama. Dejamos el id en blanco porque está autonumerado, ingresamos la fecha del día...ponemos la cantidad de pasajeros en 2... y vamos a dejar el CustomerId en 0, porque la persona que desea hacer la reserva no es cliente aún de la empresa.

Presionamos Confirmar.

Enviamos la tarea Reservation

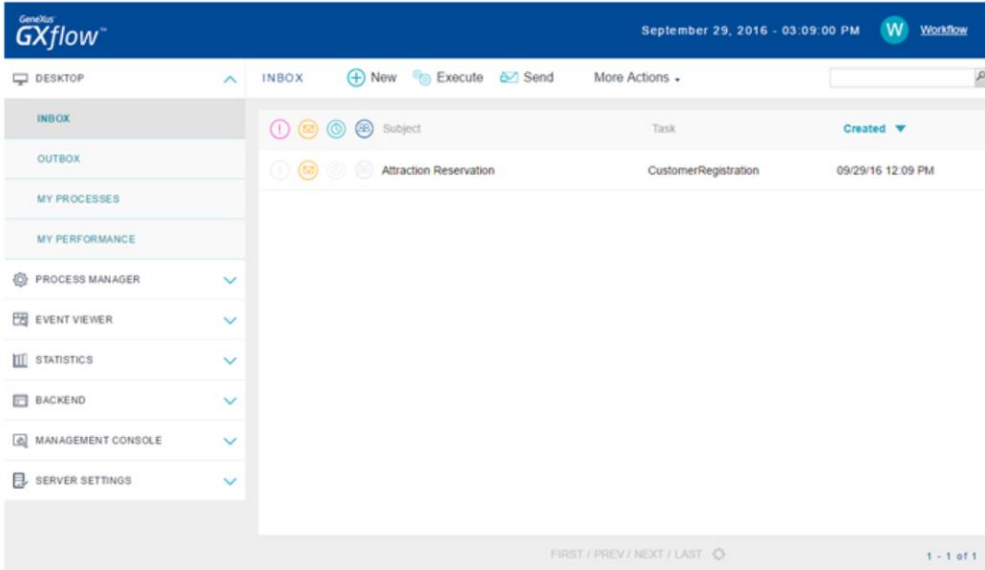


The screenshot displays the GeneXus GXflow interface. The top navigation bar includes the logo, the date and time (September 29, 2016 - 02:14:32 PM), and a 'Workflow' indicator. Below the navigation bar, there are several tabs: 'DESKTOP', 'INBOX', 'New', 'Execute', 'Send', and 'More Actions'. The 'INBOX' tab is active, showing a list of tasks. The first task is 'Attraction Reservation' with the task name 'EnterReservation' and a creation time of '09/29/16 11:13 AM'. The task is highlighted in a grey row, indicating it has been executed. The left sidebar contains various menu items such as 'PROCESS MANAGER', 'EVENT VIEWER', 'STATISTICS', 'BACKEND', 'MANAGEMENT CONSOLE', and 'SERVER SETTINGS'.

Ahora la tarea no tiene más un sobre cerrado, sino que aparece el renglón en gris que muestra que la tarea ya fue ejecutada.

Para pasar a la tarea siguiente presionamos Send.

Siguiente tarea: trn Customer



The screenshot displays the GeneXus Gxflow interface. The top navigation bar includes the Gxflow logo, the date and time (September 29, 2016 - 03:09:00 PM), and a 'Workflow' indicator. Below the navigation bar, there are several menu items: 'DESKTOP', 'INBOX', 'OUTBOX', 'MY PROCESSES', 'MY PERFORMANCE', 'PROCESS MANAGER', 'EVENT VIEWER', 'STATISTICS', 'BACKEND', 'MANAGEMENT CONSOLE', and 'SERVER SETTINGS'. The 'INBOX' menu item is selected, and the main content area shows a list of tasks. The first task is 'Attraction Reservation' with a 'CustomerRegistration' task name and a creation date of '09/29/16 12:09 PM'. The interface also includes a search bar, a 'New' button, an 'Execute' button, a 'Send' button, and a 'More Actions' dropdown menu. At the bottom of the task list, there are navigation controls: 'FIRST / PREV / NEXT / LAST' and a page indicator '1 - 1 of 1'.

Task	Created
Attraction Reservation	09/29/16 12:09 PM

Ahora la tarea que tenemos pendiente de ejecución es CustomerRegistration, dado que como no ingresamos identificador de cliente en la reserva, el flujo del diagrama seguirá hacia la derecha, abriendo la transacción Customer para que agreguemos a la persona, como cliente nuevo.

Ejecutamos la tarea CustomerRegistration

The screenshot shows the GeneXus GXflow interface. The main window is titled "Customer" and contains a form with the following fields:

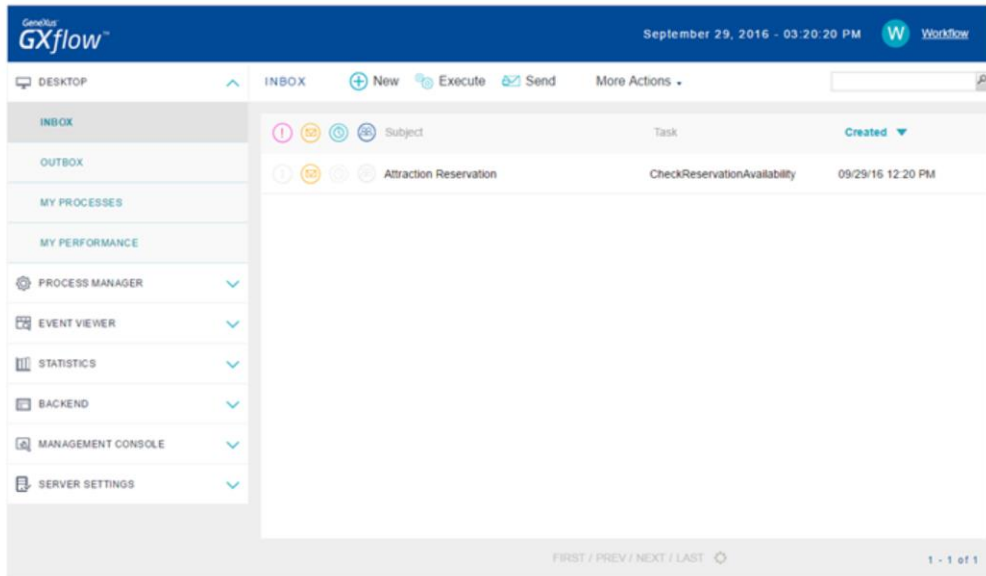
- Id: 0
- Name: Peter
- Last Name: Sloan
- Address: 64 Rockefeller St.
- Phone: 5558899
- Email: psloan@example.com
- Added Date: 09/29/16

The left sidebar shows navigation options: DESKTOP, INBOX, OUTBOX, MY PROCESSES, MY PERFORMANCE, PROCESS MANAGER, EVENT VIEWER, STATISTICS, BACKEND, MANAGEMENT CONSOLE, and SERVER SETTINGS. The right sidebar shows a "Created" dropdown and a table with a row for "CustomerRegistration" dated "09/29/16 12:09 PM".

Ejecutamos la tarea, se abre la pantalla de la transacción Customer y procedemos a ingresar al cliente.

Confirmamos y cerramos la ventana.... Ahora presionamos el botón de Send para terminar la tarea y ejecutar la siguiente.

Siguiente tarea: ReservationAvailability

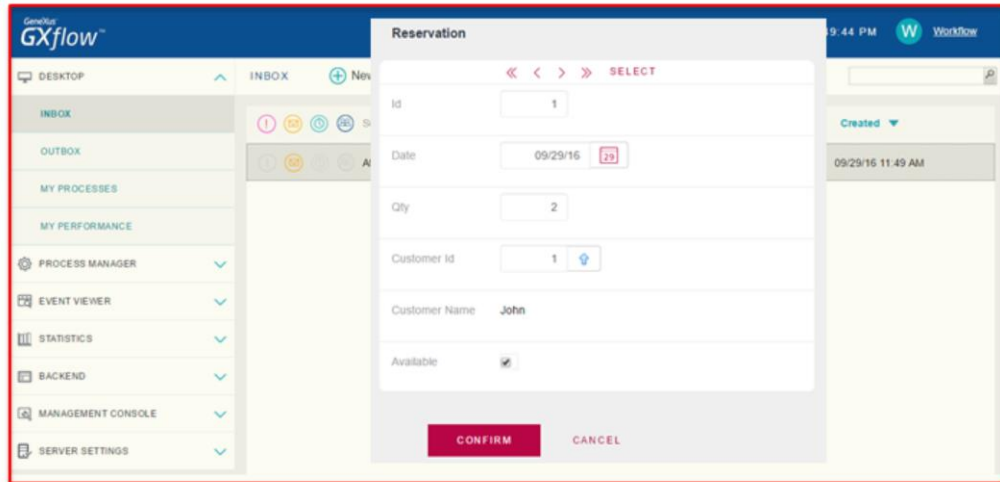


Vemos que la tarea próxima que tenemos es la llamada CheckReservationAvailability.

Si recordamos el proceso, la tarea CustomerRegistration invocaba a una tarea llamada AssignCustomerToReservation, que invocaba a un procedimiento para asignar el identificador del cliente recién ingresado a la reserva.

Como la tarea AssignCustomerToReservation es no interactiva fue ejecutada por el motor de Gxflow y no se nos presentó en la bandeja de entrada cuando finalizamos la tarea CustomerRegistration, sino que directamente se nos mostró la próxima tarea en la que tenemos que intervenir, es decir CheckReservationAvailability.

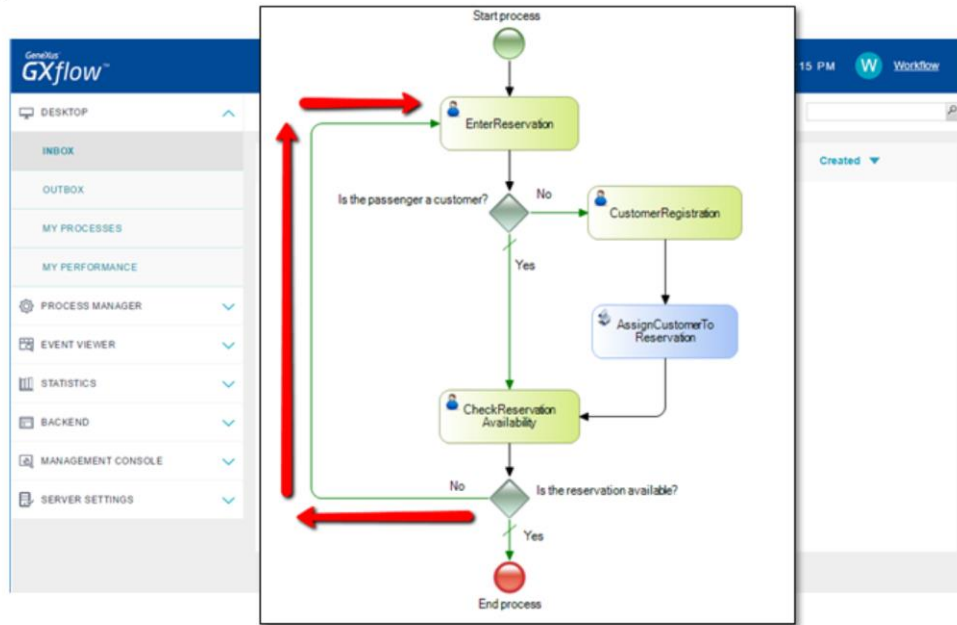
Ejecutamos la tarea CheckReservationAvailability



The screenshot displays the GeneXus GXflow application interface. On the left, a navigation menu lists various sections: DESKTOP, INBOX, OUTBOX, MY PROCESSES, MY PERFORMANCE, PROCESS MANAGER, EVENT VIEWER, STATISTICS, BACKEND, MANAGEMENT CONSOLE, and SERVER SETTINGS. The main area shows a 'Reservation' form with the following fields: Id (1), Date (09/29/16), Qty (2), Customer Id (1), Customer Name (John), and an Available checkbox that is checked. At the bottom of the form are two buttons: CONFIRM and CANCEL. The top right corner of the application shows the time 9:44 PM and the user's name Workflow.

Si ejecutamos la tarea CheckReservationAvailability, haciendo doble clic sobre la misma, se nos abre el formulario de la transacción Reservation, para que ingresemos si la reserva está disponible o no. Supongamos que sí, así que marcamos el checkbox y presionamos Confirmar.

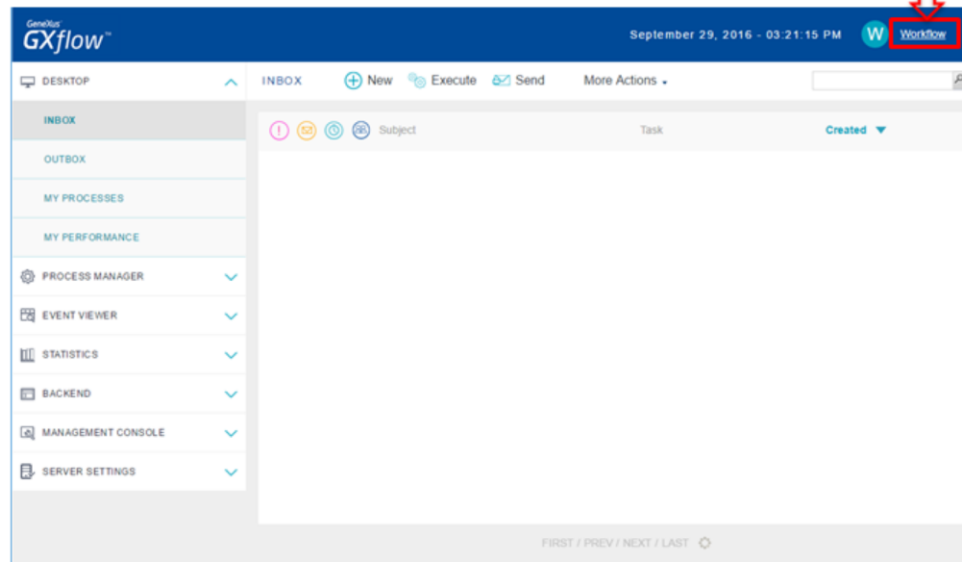
Fin del proceso



Ahora la bandeja de entrada se nos muestra vacía. Esto significa que no hay más tareas pendientes para ejecutar ya que hemos llegado al fin del proceso.

Notemos que si no hubiéramos marcado el check box, el workflow nos hubiera llevado nuevamente a la tarea EnterReservation al principio del diagrama.

Administración de usuarios



Algo que no mencionamos es que todas estas tareas fueron ejecutadas por el mismo usuario.

Esto es así mientras desarrollamos y probamos el proceso, pero en el uso real, habrá diferentes usuarios que ejecutarán las distintas tareas, de acuerdo a los perfiles que tengan en la organización.

Ver la historia del proceso

The screenshot displays the GeneXus GXflow interface. The top navigation bar includes the 'MY PROCESSES' tab, which is highlighted with a red box. Below the navigation bar, the 'MY PROCESSES' section is selected in the left sidebar, also highlighted with a red box. The main content area shows a table of process history with the following data:

Subject	Process	State	Created	Ended
Attraction Res...	AttractionRese...	completed	09/29/16 12:08 PM	09/29/16 12:21 PM
Attraction Rese...	AttractionRese...	aborted	09/29/16 11:46 AM	09/29/16 12:08 PM
Attraction Rese...	AttractionRese...	aborted	09/29/16 11:13 AM	09/29/16 11:46 AM

Otra cosa interesante es que podemos ver la historia del proceso, es decir, qué caminos del diagrama se recorrieron cuando se ejecutó el proceso.

Para ver la historia, seleccionamos My Processes en la ventana Navigator. Vemos que se muestra el proceso que acabamos de ejecutar (Attraction Reservation) que aparece en estado completado.

Ver la historia del proceso

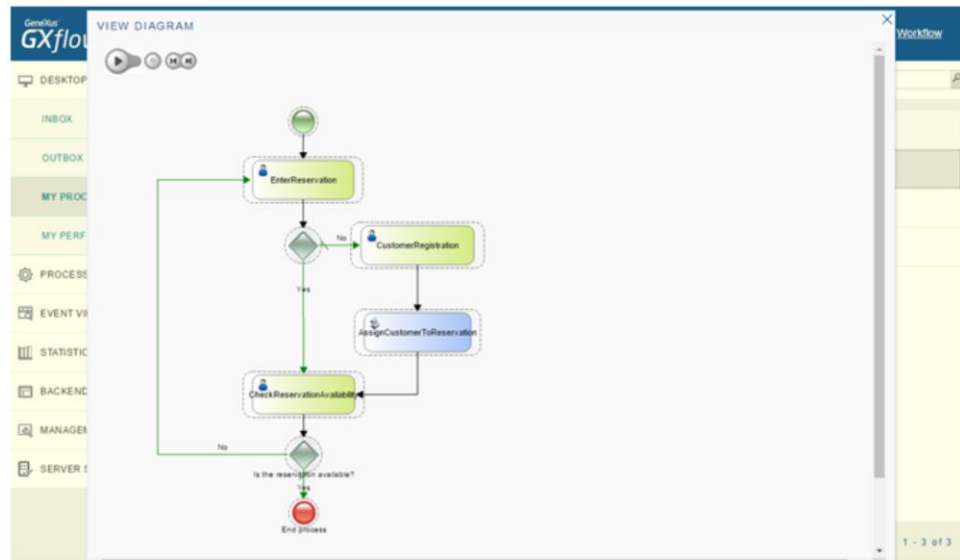
The screenshot displays the GeneXus GXflow interface. At the top, the date and time are 'September 29, 2016 - 03:25:30 PM'. The interface includes a sidebar with navigation options like 'DESKTOP', 'INBOX', 'OUTBOX', 'MY PROCESSES', 'MY PERFORMANCE', 'PROCESS ADMINISTRATION', 'EVENT VIEWER', 'STATISTICS', 'BACKEND', 'MANAGEMENT', and 'SERVER SETTINGS'. A 'HISTORY' window is open, showing a table of process steps. The table has columns for Step, State, User, Created, and Ended. The steps listed are: 'Start process', 'EnterReservation', 'Is the passenger a customer?', 'CustomerRegistration', 'AssignCustomerToReservation', 'CheckReservationAvailability', and 'Is the reservation available?'. All steps are marked as 'completed'. The 'Created' and 'Ended' columns show timestamps from 09/29/16 12:08 PM to 12:21 PM. The user for most steps is 'Workflow Administrator', and one is 'Workflow Engine'. Navigation controls like 'FIRST / PREV / NEXT / LAST' and '1 - 8 of 8' are visible at the bottom of the history window.

Step	State	User	Created	Ended
Start process	completed		09/29/16 12:08 PM	09/29/16 12:08 PM
EnterReservation	completed	Workflow Administrator	09/29/16 12:08 PM	09/29/16 12:08 PM
Is the passenger a customer?	completed		09/29/16 12:08 PM	09/29/16 12:08 PM
CustomerRegistration	completed	Workflow Administrator	09/29/16 12:09 PM	09/29/16 12:20 PM
AssignCustomerToReservation	completed	Workflow Engine	09/29/16 12:20 PM	09/29/16 12:20 PM
CheckReservationAvailability	completed	Workflow Administrator	09/29/16 12:20 PM	09/29/16 12:21 PM
Is the reservation available?	completed		09/29/16 12:21 PM	09/29/16 12:21 PM

Hacemos doble clic sobre el mismo, y vemos que se abre una ventana que nos muestra la historia del proceso.

En esta ventana vemos todas las tareas que se fueron ejecutando.

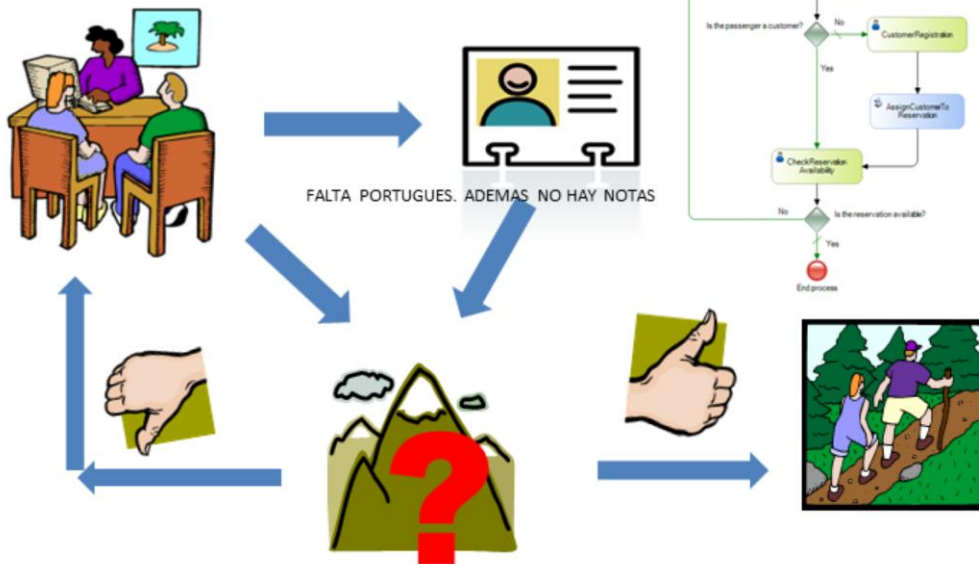
Ver el diagrama como animación



Si vamos a More Actions, View Diagram, podemos ver la historia en forma de animación.

Presionamos Play...

Business Process Management (BPM)



Hemos visto así, que GeneXus nos permite modelar procesos de negocios en forma intuitiva, resolviendo automáticamente qué objeto debe ejecutarse en cada caso, de acuerdo a la definición que hicimos a través del diagrama.

GeneXus™

The power of doing.

Videos

training.genexus.com

Documentation

wiki.genexus.com

Certifications

training.genexus.com/certifications