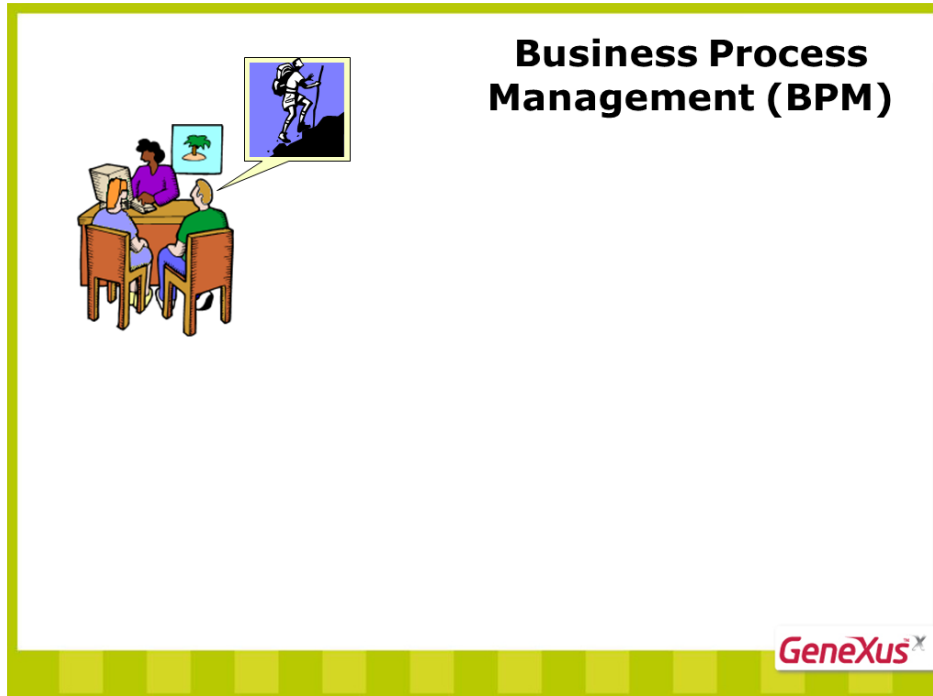
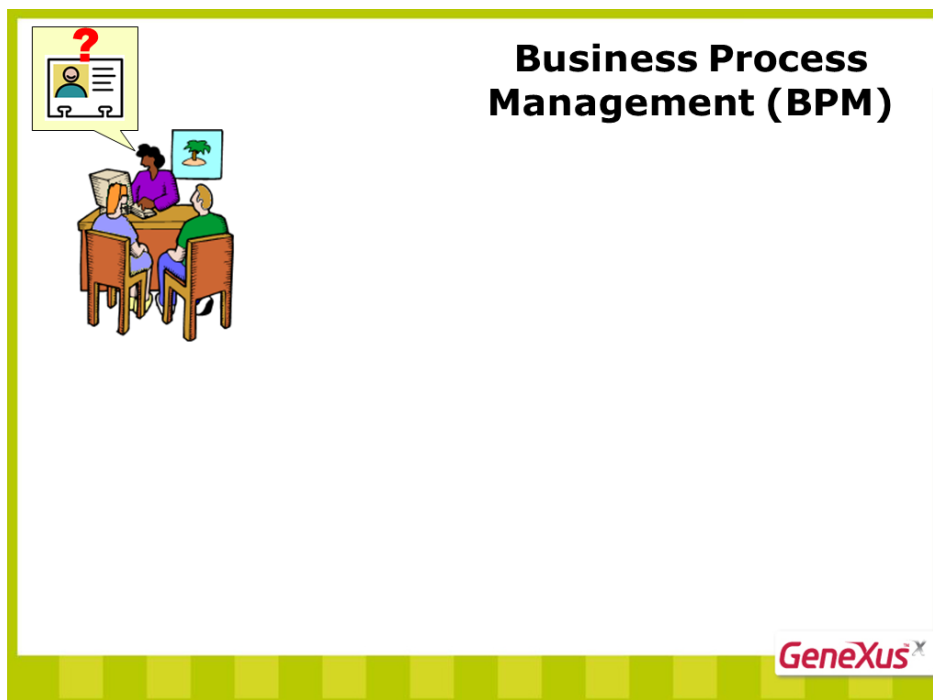


## Curso GeneXus – Definindo processos de negócios (BPM)

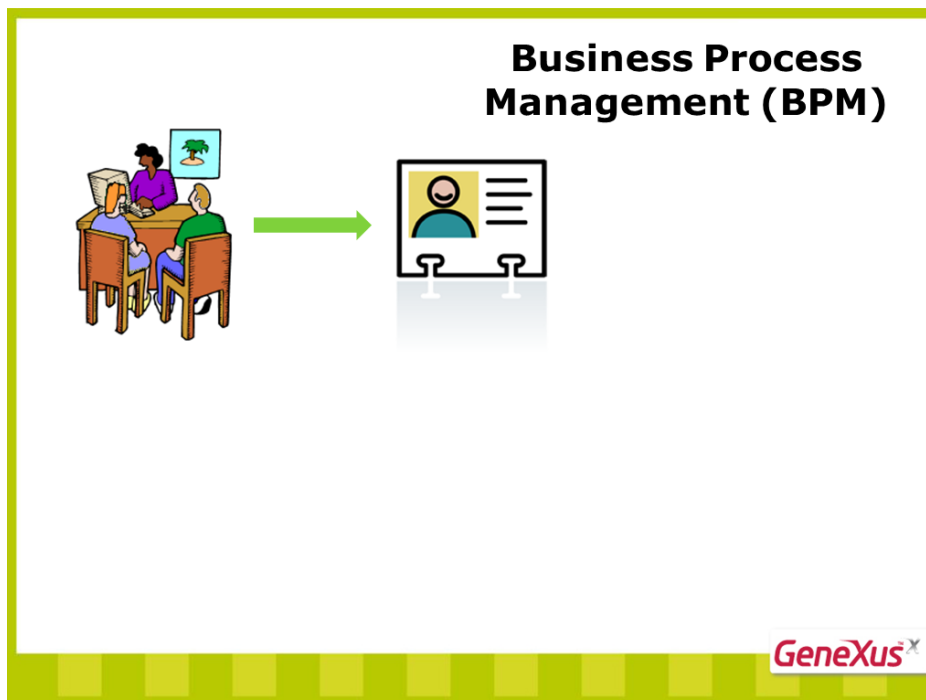
Numa agência de viagens, há trâmites que exigem que várias tarefas se realizem em sequência, numa certa ordem; quando se deseja fazer uma reserva de um pacote turístico para determinada atração, por exemplo.



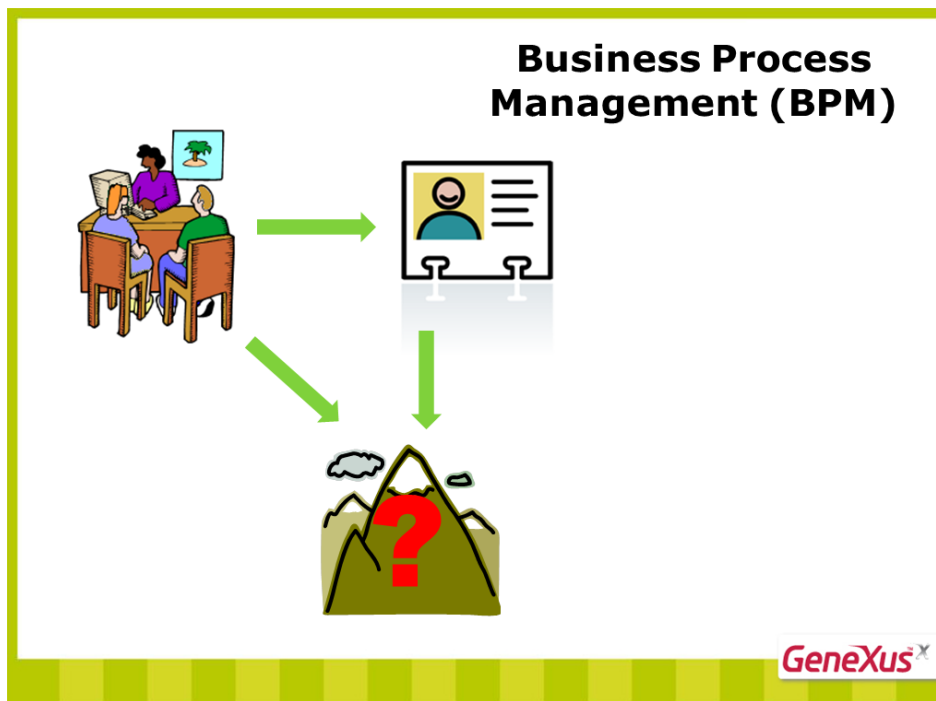
Ao fazer a reserva, é preciso verificar se a pessoa que solicita o pacote é cliente da agência.



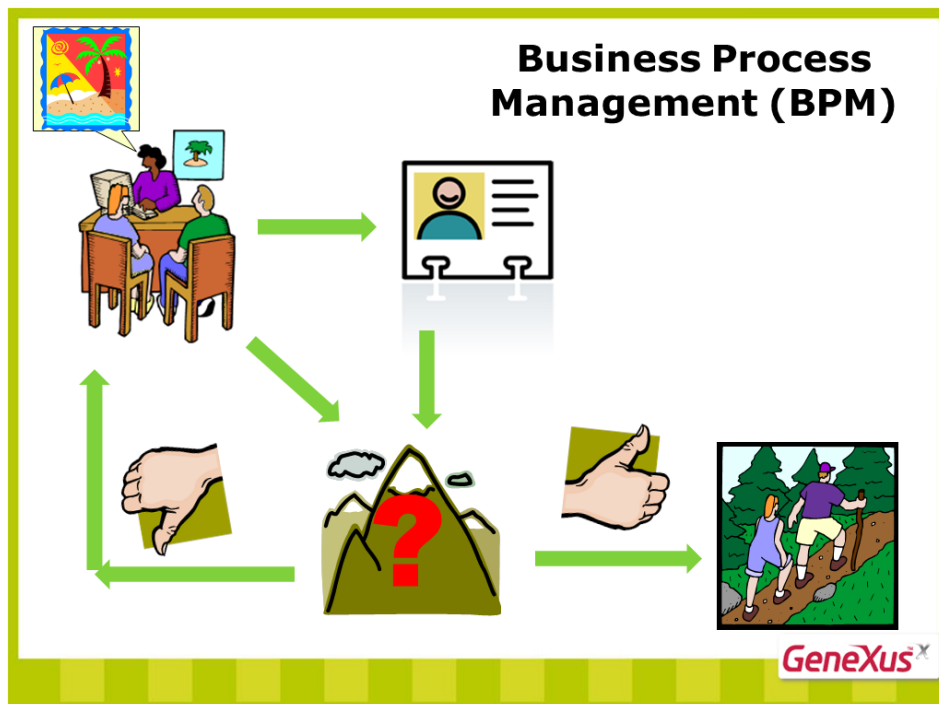
Se não for, deve-se inseri-lo como cliente.



Feito isso, é preciso verificar se existem pacotes disponíveis para a quantidade de pessoas que querem fazer a viagem.



Se há lugares disponíveis, a reserva será feita; caso contrário, deve-se oferecer outro pacote ao passageiro.



Uma sequência de passos como a que acabamos de ver é o que se denomina processo de negócio.

Em GeneXus, podemos definir processos de negócios como este. Vejamos como fazer!

Como precisamos registrar as reservas dos pacotes turísticos, o que fizemos anteriormente foi criar uma transação de reservas com atributos para armazenar o identificador e a data da reserva, a quantidade de pessoas que usarão o pacote, o identificador e o nome do cliente da reserva e ReservationAvailable, para marcar se o pacote turístico está ou não disponível.

A captura de tela mostra a interface do GeneXus com a aba "Reservation" selecionada. A tabela de atributos é a seguinte:

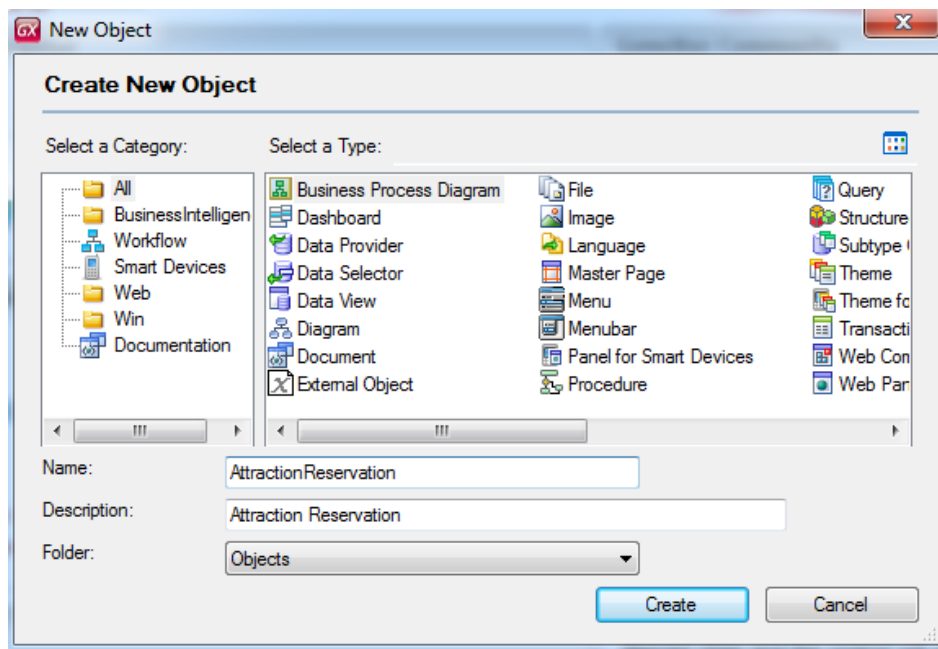
Name	Type	Description	Formula	Nullable
Reservation	Reservation	Reservation		
ReservationId	Id	Reservation Id		No
ReservationDate	Date	Reservation Date		No
ReservationQty	Numeric(4,0)	Reservation Qty		No
CustomerId	Id	Customer Id		Yes
CustomerName	Name	Customer Name		
ReservationAvailable	Boolean	Reservation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

Na barra de ferramentas inferior, as opções "Structure", "Web Form", "Win Form", "Rules", "Events", "Variables", "Help", "Documenta..." e "Patterns" são visíveis.

Para o atributo CustomerId, selecionamos em sua coluna Nullable o valor Yes, para indicar que no momento de inserir uma reserva, podemos não ter ainda o identificador do cliente, da pessoa que está adquirindo a reserva... Em seguida, salvamos...

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	Id	Customer Id		Yes
CustomerName	Name	Customer Name		
ReservationAvailable	Boolean	Reservation Avai...		No

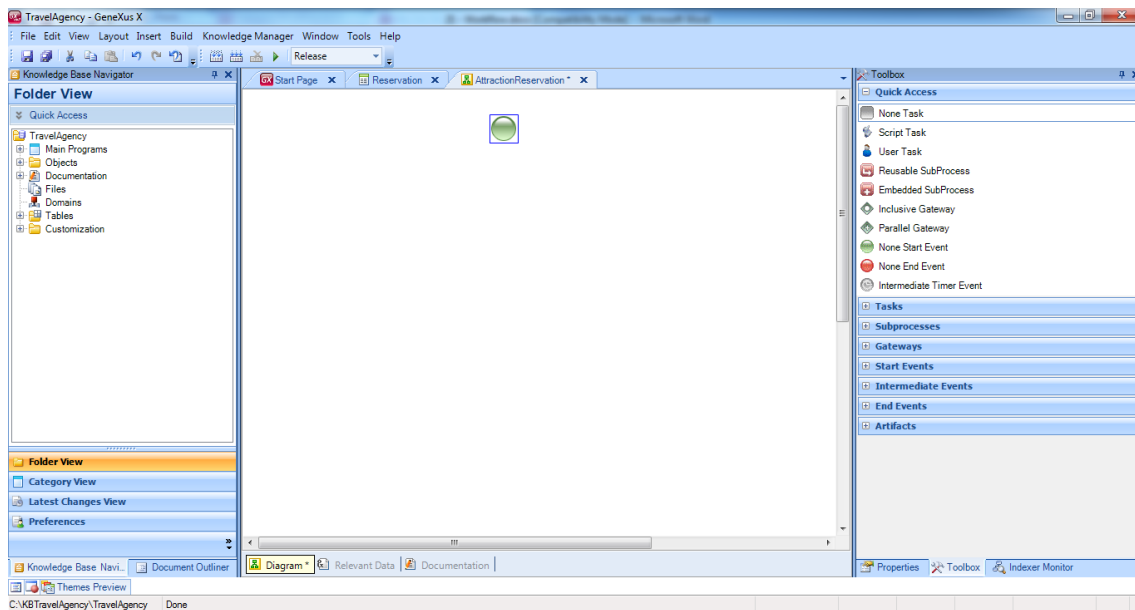
Agora, vamos criar um objeto do tipo diagrama de processos de negócios. Clicamos em File... New... e escolhemos um objeto do tipo Business ProcessDiagram que chamaremos AttractionReservation.



Vemos uma página em branco, é a página do diagrama que vamos construir.

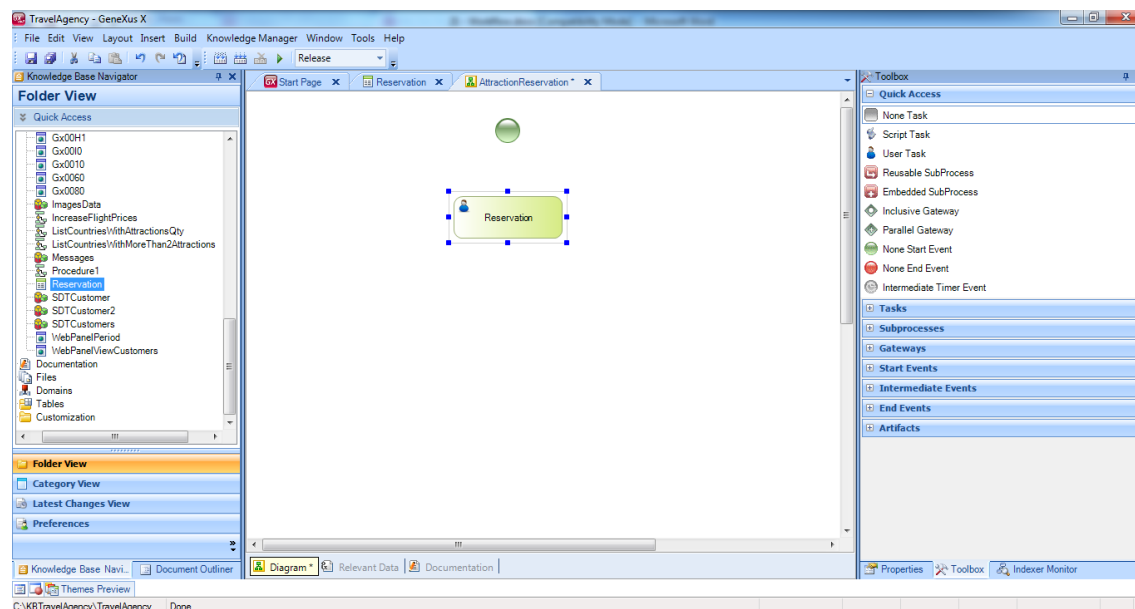
Se abrirmos a caixa de ferramenta, veremos que existe uma grande quantidade de símbolos que podemos usar.

Para marcar o começo de nosso processo de negócios, arrastemos um símbolo de NoneStartEvent.

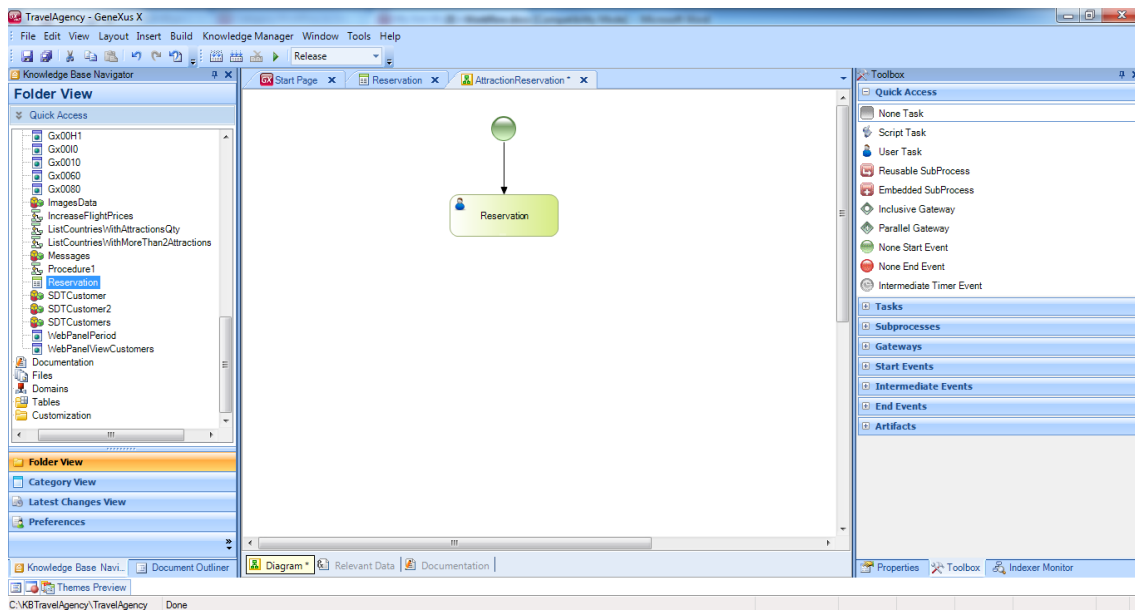


Como a primeira atividade que vamos realizar é inserir uma reserva, localizamos a transação Reservation na janela de Folder View e a arrastamos até o diagrama.


Vemos que se criou um retângulo verde com o nome Reservation com um desenho de uma pessoa na parte superior, à esquerda. Isso nos indica que foi criada uma **atividade do tipo interativa**, já que a transação permite que um usuário insira dados através dela.




Para conectar o nó de Start com a transação, clicamos na parte inferior do círculo verde e, sem soltar, arrastamos até que a ponta da flecha toque a margem superior do retângulo da transação.



Segundo o processo que continua na agência de viagens, o sistema deve controlar se a pessoa que contrata a reserva já é ou não cliente da empresa.

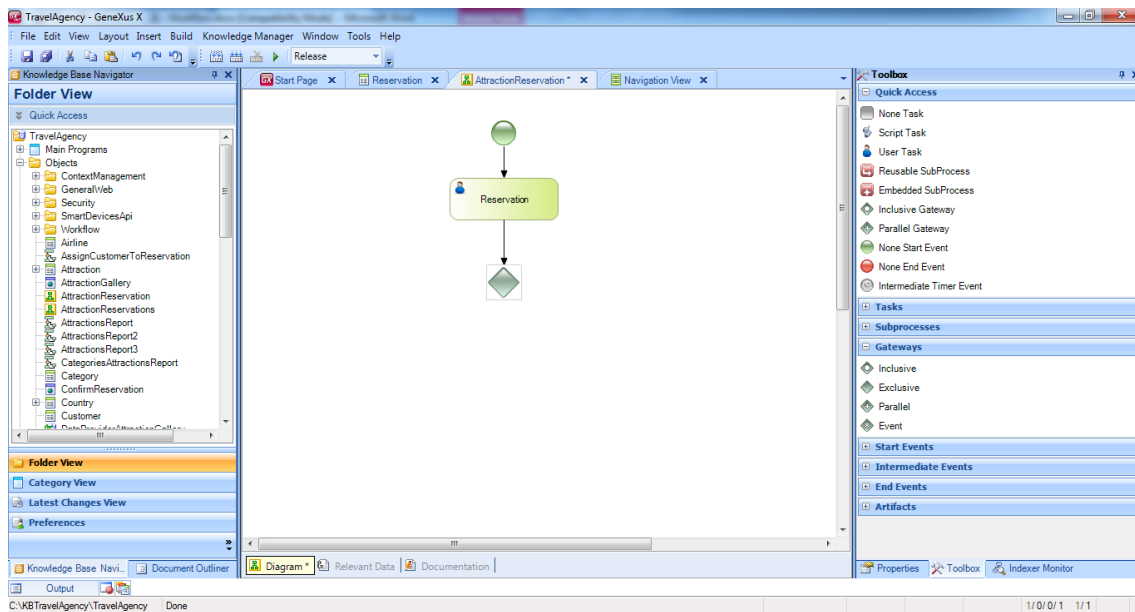


## Business Process Management (BPM)



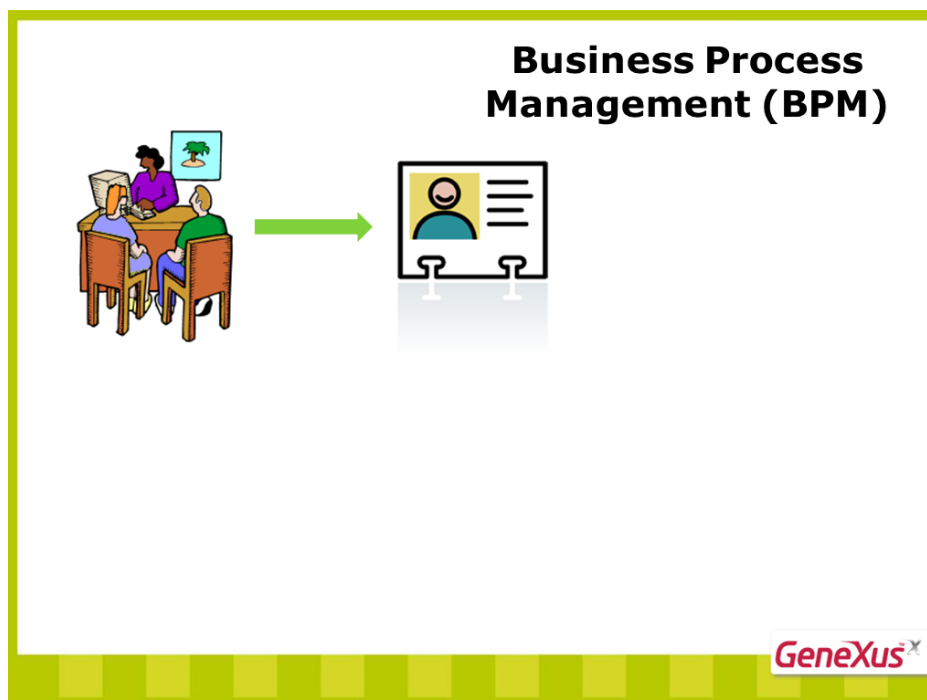
Para representar uma **decisão** no diagrama, vamos à barra de ferramentas, clicamos sobre Gateway, arrastamos um nó **Exclusive Gateway** sobre o diagrama e o unimos partindo-se da transação Reservation.

Esse tipo de nó avalia uma condição e dependendo do resultado, faz com que o fluxo continue para baixo (que seria o curso normal) ou à direita do símbolo (que seria o curso alternativo).

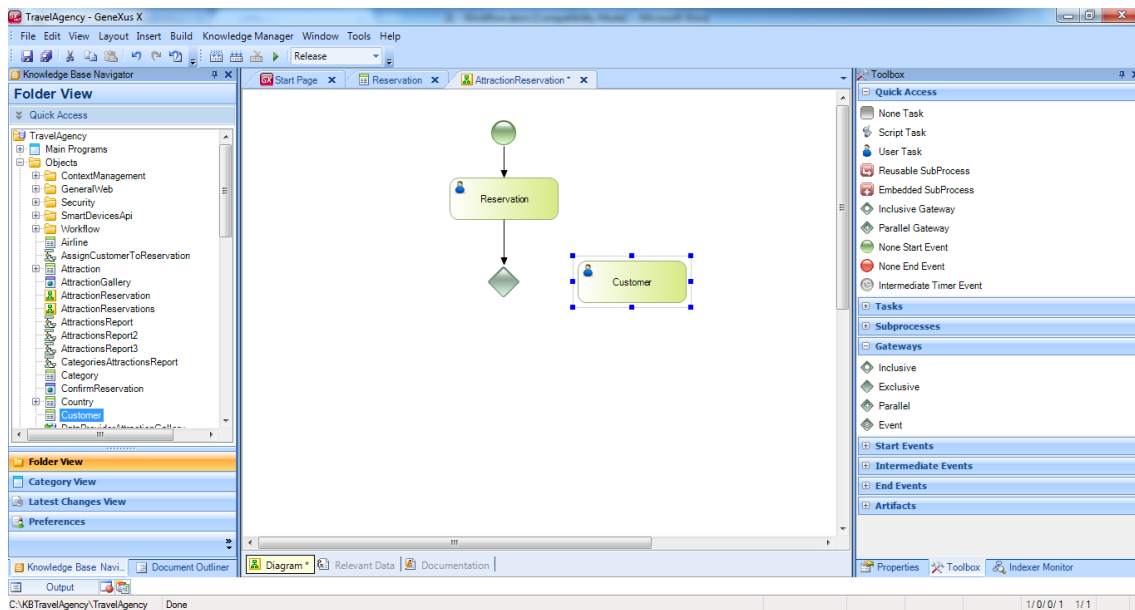


No nosso exemplo, devemos definir a condição do Gateway que fará com que o fluxo continue o curso normal ou siga o fluxo alternativo, para acrescentar um cliente.

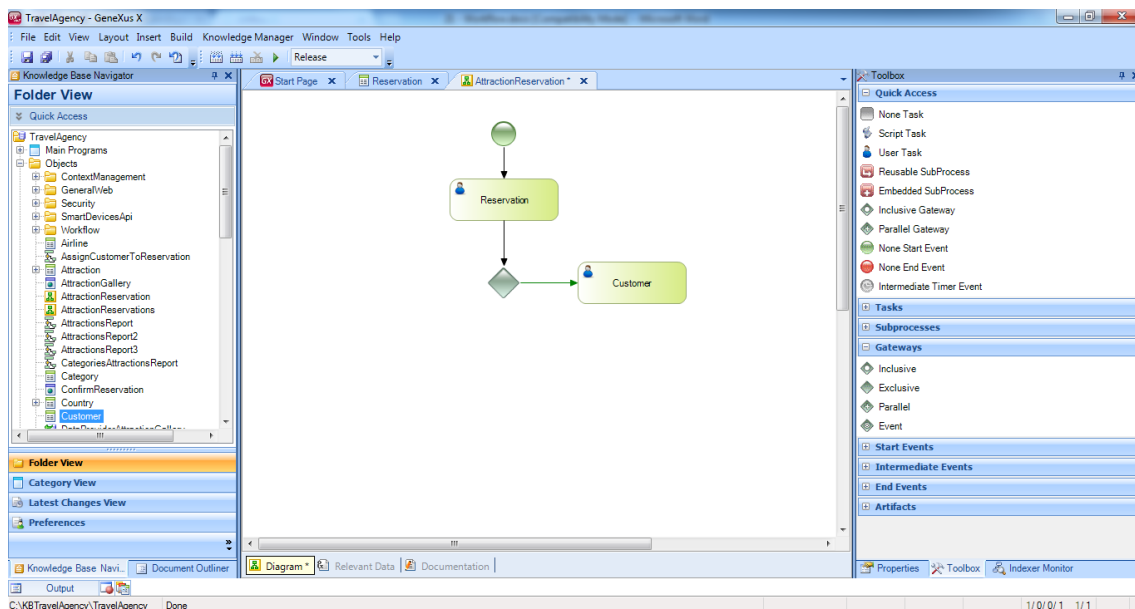
Em outras palavras, devemos definir o seguinte: se a pessoa que deseja fazer a reserva do pacote turístico **não é** cliente da empresa, invoca-se a transação Customer para inseri-lo como cliente.



Para começar, arrastamos a transação Customer ao diagrama...



e conectamo-la partindo-se do nó Exclusive Gateway...



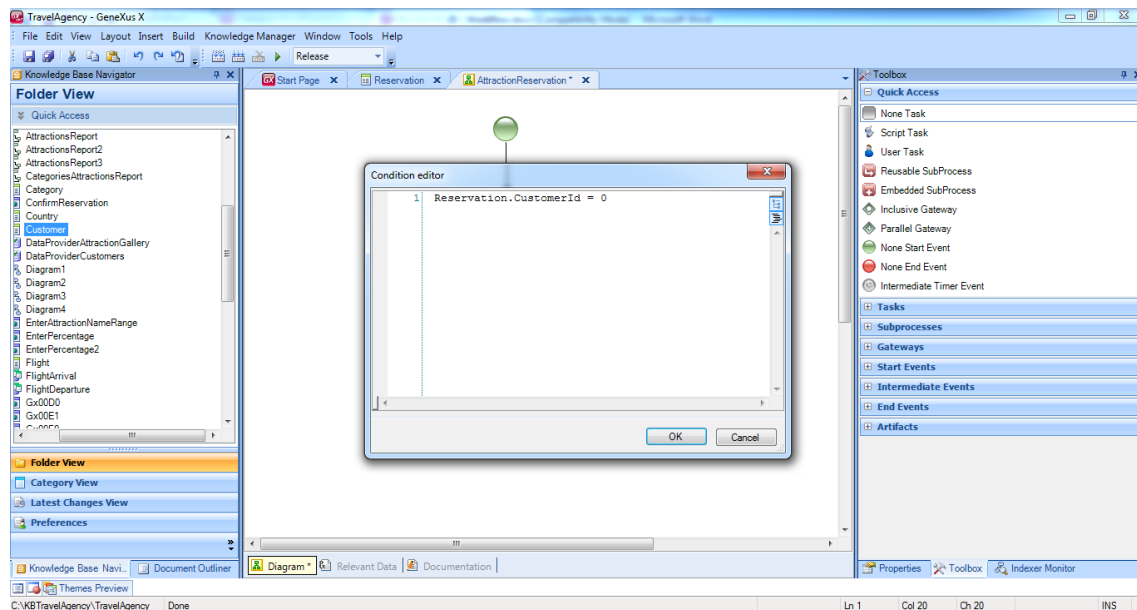
Para completar a definição da decisão, temos que acrescentar a condição que fará com que partindo-se do nó de decisão, o fluxo do processo continue à direita invocando a transação Customer, ou continue para abaixo, para tarefa seguinte.

No nosso caso, o caminho à direita é o fluxo alternativo; o caminho pra baixo, por sua vez, é o fluxo habitual, quando a pessoa que solicita a reserva já é cliente da agência de viagens.

Para definir a condição que provoca a bifurcação, damos um duplo clique na flecha verde que une o Gateway com a transação Customer e vemos que se abre uma janela do **editor de condições**.

Escrevemos a seguinte expressão: `Reservation CustomerId = 0`, para indicar que o fluxo deve seguir esse caminho se quando se fez a reserva, o atributo CustomerId ficou com o valor 0.

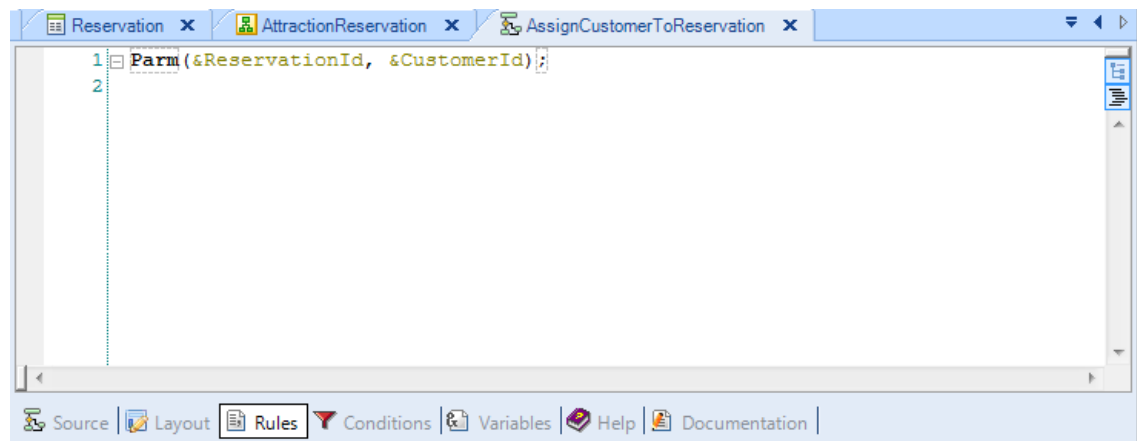




Se se segue o fluxo alternativo, a transação de Customer será aberta para que acrescentemos a pessoa da reserva como cliente.

Assim que inserido, devemos associar o cliente recém-criado à reserva.

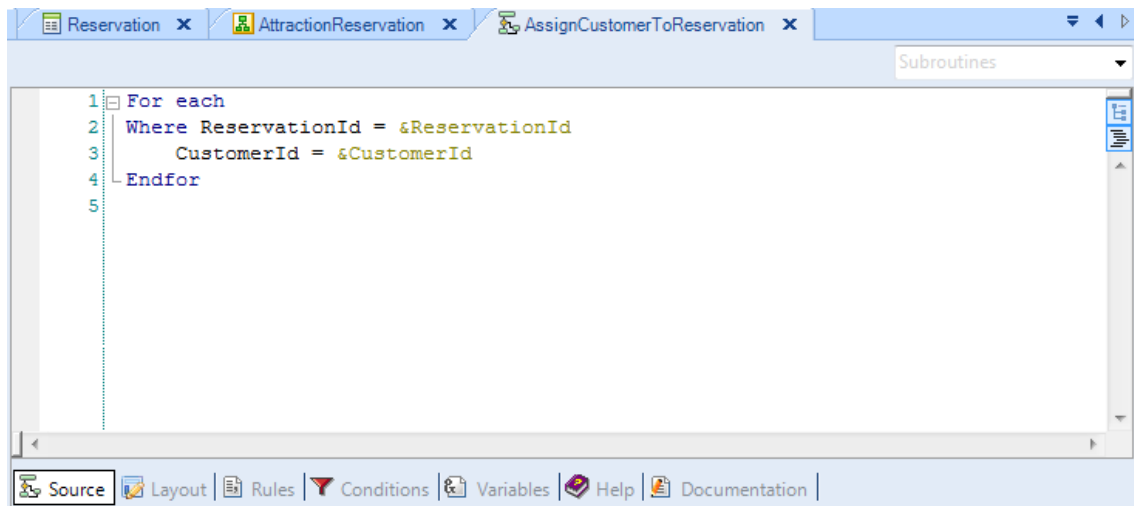
Para isso, devemos criar um procedimento... que chamaremos AssignCustomerToReservation e, na seção de regras, escrevemos uma regra parm com os parâmetros &ReservationId e &CustomerId...



Essas variáveis armazenam o identificador da reserva que criamos no início e o identificador do cliente que queremos associar a tal reserva.

Em Source escrevemos... For each... WhereReservationIdigual... na variável ReservationId que recebemos por parâmetro. Depois, atribuímos a CustomerId o valor da variável &CustomerId e fechamos o For Each.

Dessa forma, atribuímos ao cliente recém-criado a reserva inserida anteriormente.

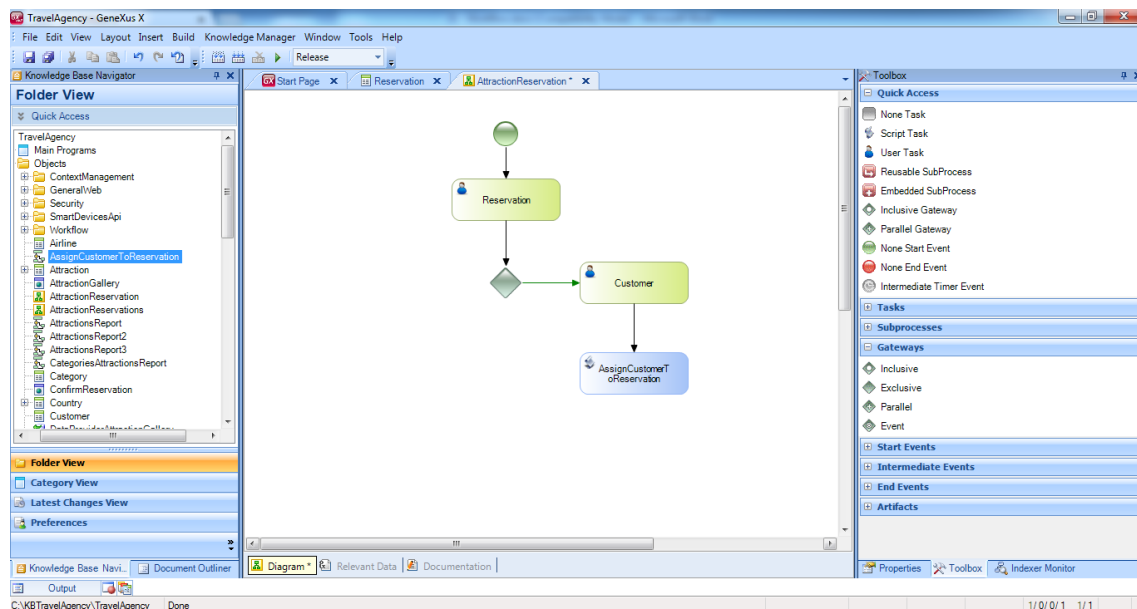


Salvamos e voltamos ao diagrama.

Agora, arrastamos o procedimento recém-criado até o diagrama. Vemos que foi criado um retângulo de cor azul, que significa que a tarefa criada é do tipo **não interativa**, também chamada tipo **batch**.

Isso é assim porque o procedimento executa-se sem intervenção de uma pessoa, ou seja, não há interação com o usuário inserindo dados, como no caso de uma transação.

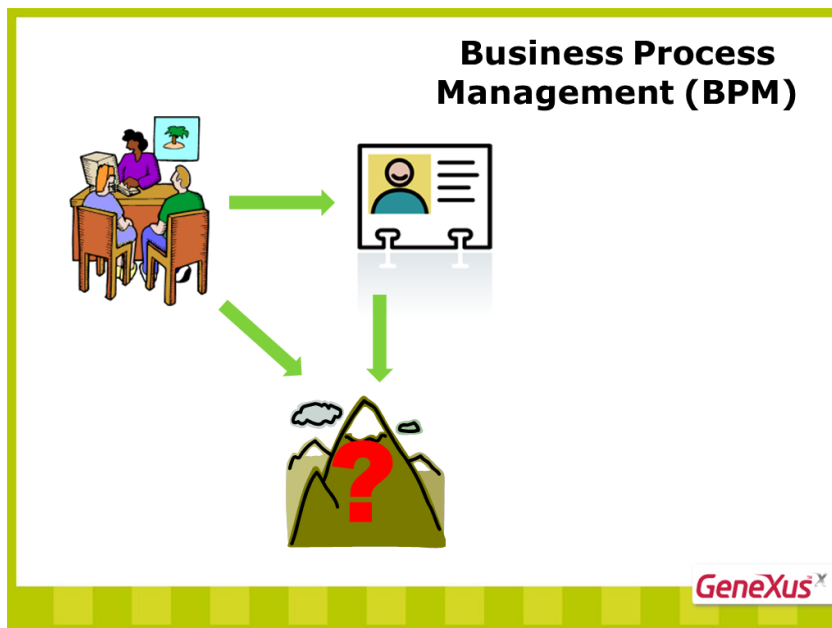
Por último, conectamos o procedimento AssignCustomerToReservation da transação Customer...



Muito bem, até agora descrevemos as tarefas que se realizarão se o fluxo alternativo é executado, ou seja, se é preciso acrescentar um cliente e atribuí-lo à reserva.

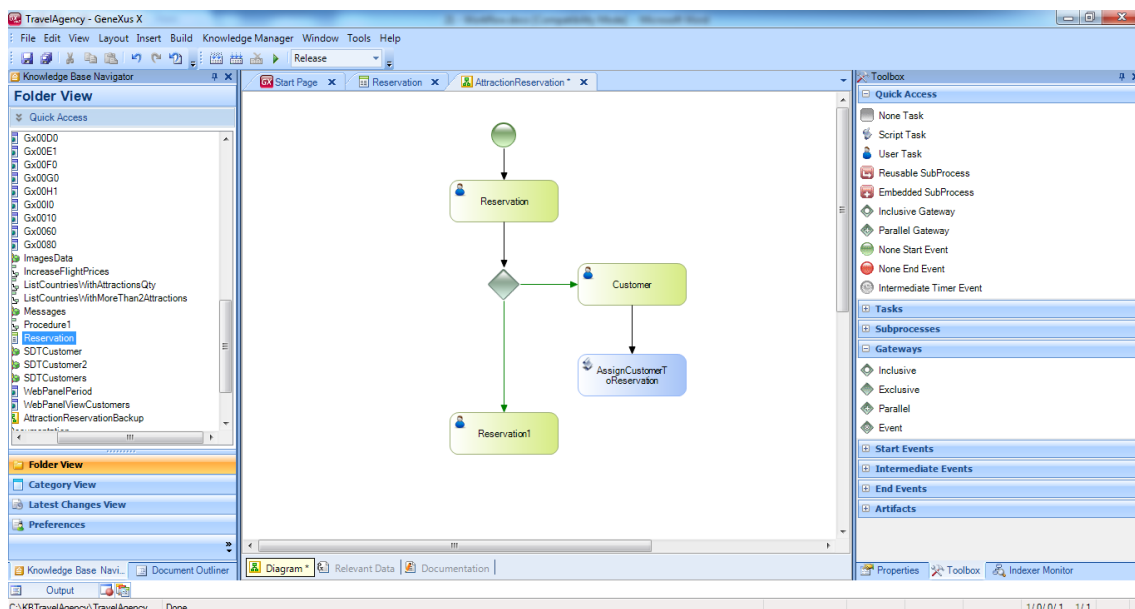
Agora, acrescentamos a atividade que se realizará se o fluxo for o habitual, isto é, se não for necessário invocar a transação cliente.

Neste ponto, o funcionário da agência deveria verificar a disponibilidade da reserva (por exemplo, vendo se há voos disponíveis, se ainda há lugares no pacote solicitado para as datas previstas etc) e finalmente indicar ao sistema se a reserva pode ou não ser finalizada.

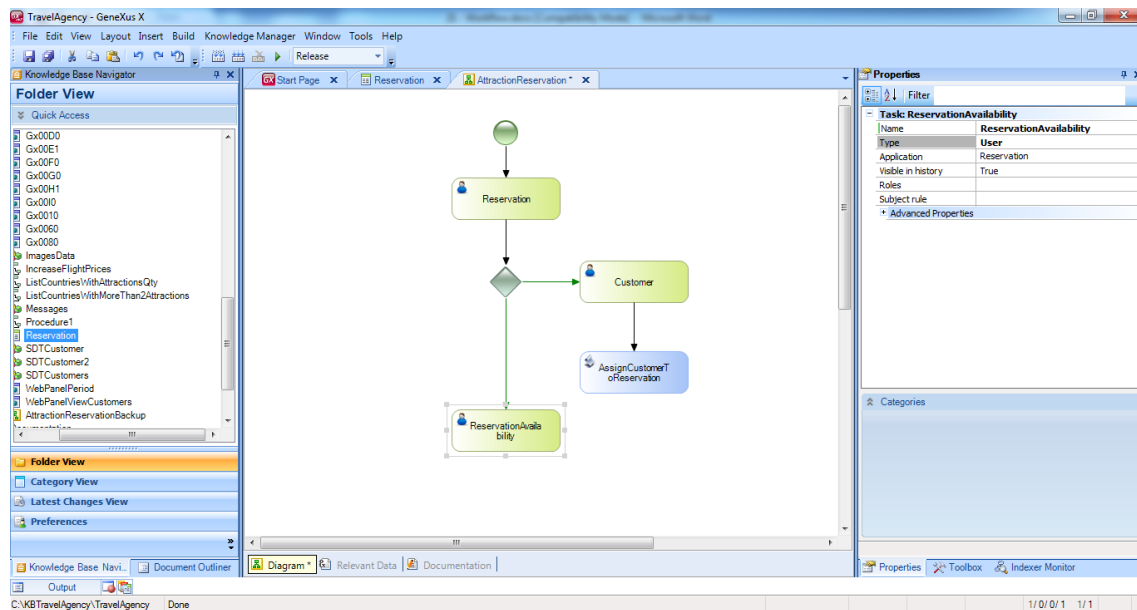


Para inserir este dado, devemos contar com uma tela onde possamos confirmar ou cancelar a reserva. Para isso, podemos abrir novamente a transação Reservation e marcar a reserva como disponível ou não, mediante o atributo ReservationAvailable.

Para isso, localizamos a transação Reservation na janela de Folder View, arrastamos até o diagrama e a conectamos partindo-se do Gateway.

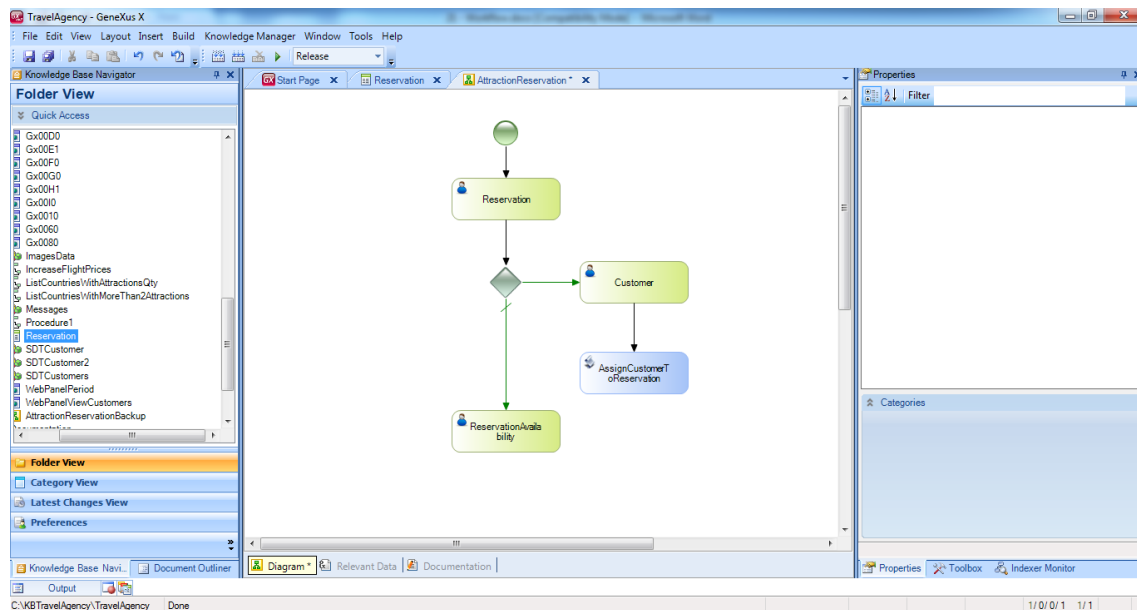


Vemos que foi criada uma atividade com o nome **Reservation1**, o próximo passo é pressionar F2 e mudar seu nome para ReservationAvailability. Notemos que aqui não estamos mudando o nome da transação, mas da atividade do diagrama que invoca a transação Reservation.

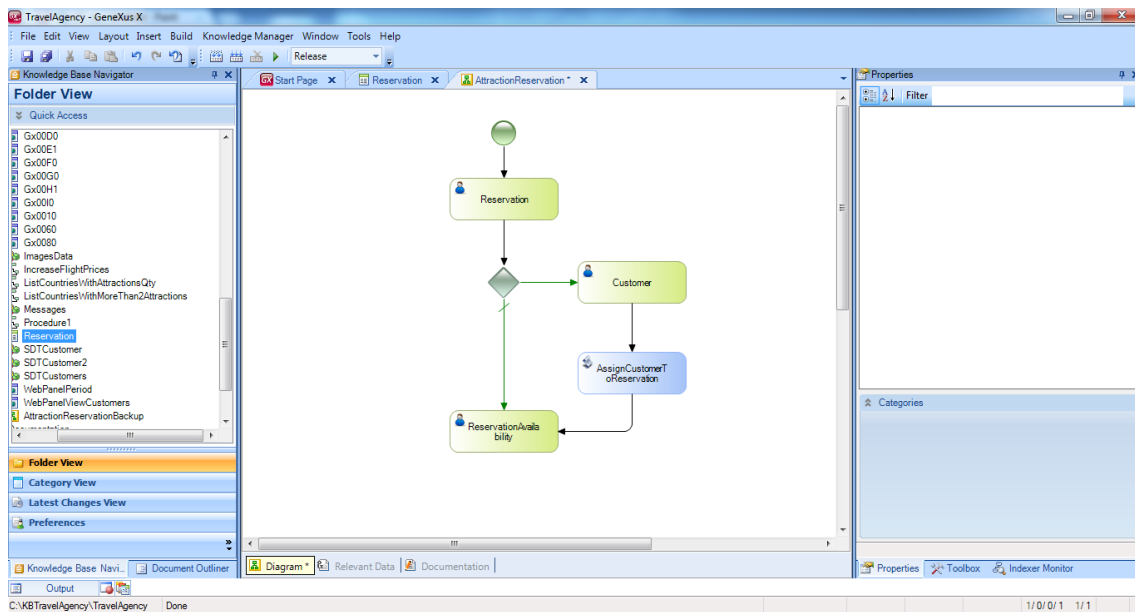


Quando inserimos o Gateway que nos permitia tomar a decisão, dissemos que o fluxo à direita que ia à transação Customer era o fluxo alternativo e o fluxo que sairia para abaixo era o fluxo normal, de quando a pessoa que fazia a reserva já era cliente da agência.

Para indicar que o fluxo para baixo é o fluxo normal, selecionamos a conexão e na janela Propriedades colocamos sua propriedade ConditionalType no valor Default. Observemos no digrama que o fluxo ficou assinalado com uma linha de cor verde que o cruza.

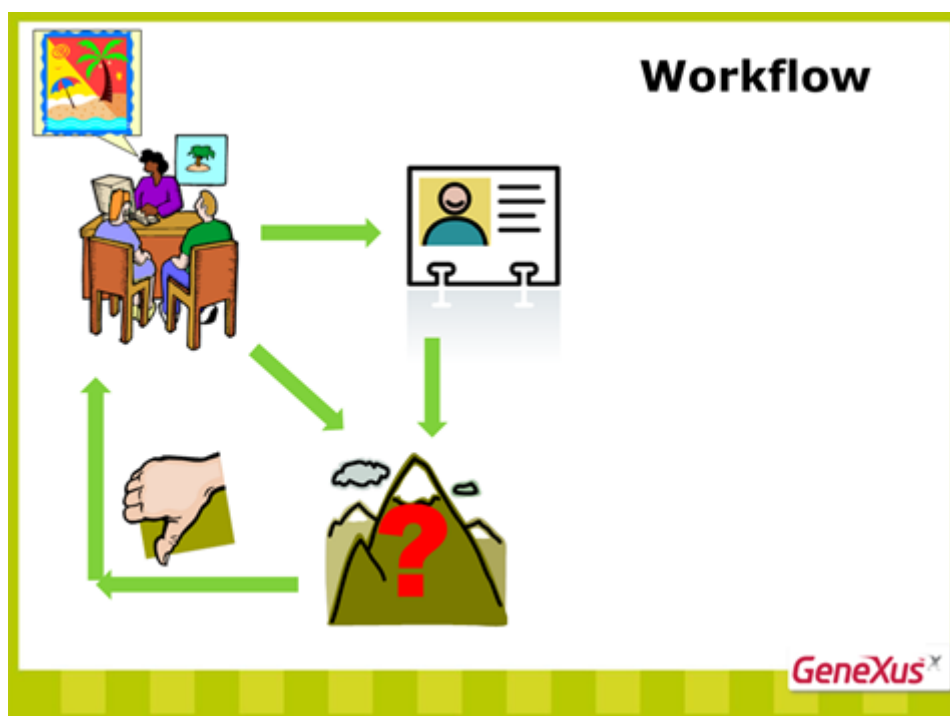


Aproveitamos agora para conectar a tarefa AssignCustomerToReservation à tarefa ReservationAvailability, já que depois de finalizada a atribuição do cliente, deve-se continuar também com a verificação da disponibilidade da reserva.

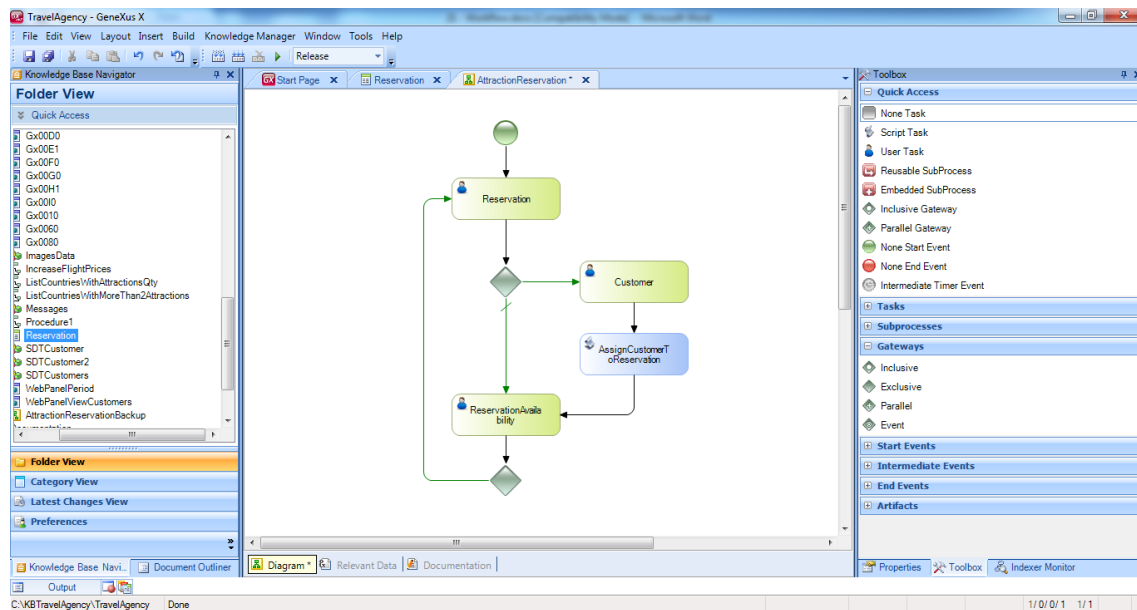


Continuando com nosso diagrama...

Finalizada a tarefa ReservationAvailability, devemos avaliar o valor que atribuímos ao atributo ReservationAvailable. Se a caixa de seleção ficou desmarcada, significa que a reserva não estará disponível e deve-se oferecer um novo pacote para reservar.

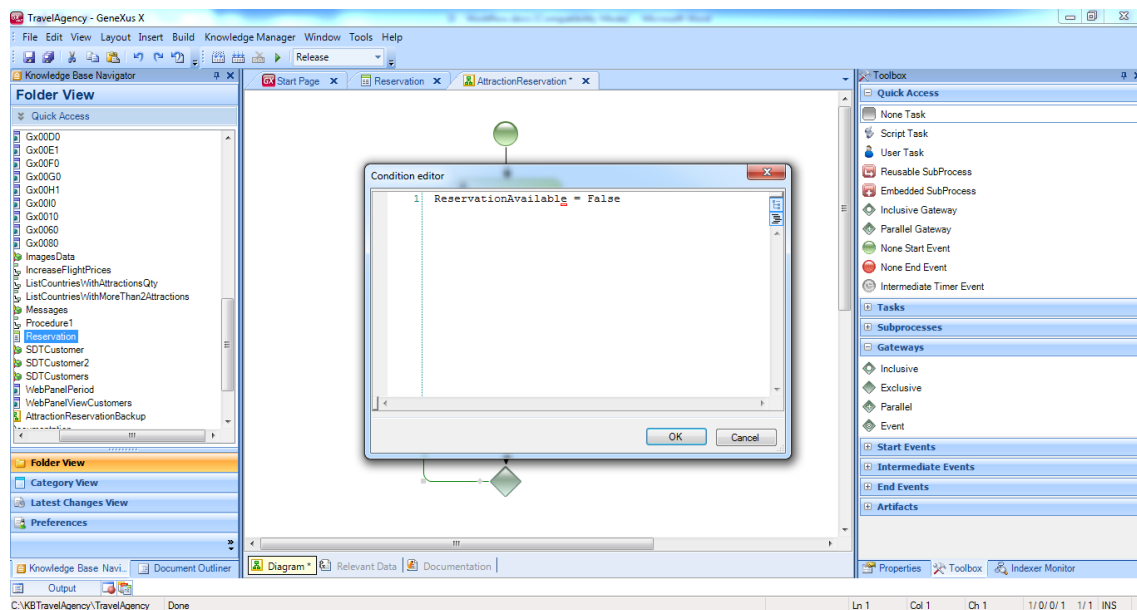


Pra expressar isso, inserimos da barra de ferramentas um Exclusive Gateway e o conectamos partindo-se da tarefa ReservationAvailability. Depois, conectamos seu fluxo alternativo (que desenhamos indo para a esquerda) à tarefa Reservation, para que se possa inserir uma nova reserva.

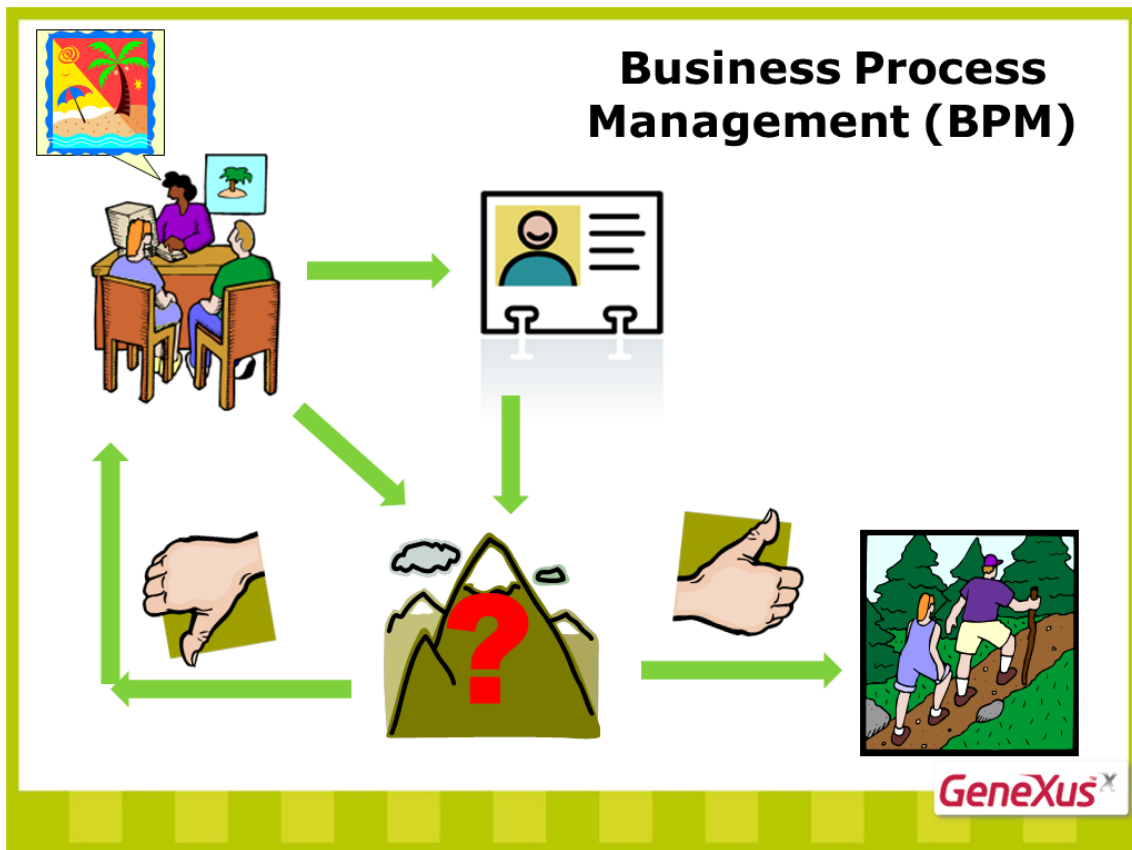


Agora devemos inserir a condição exigida para que o fluxo alternativo se cumpra, isto é, quando a reserva não estiver disponível.

Pra indicar isso no diagrama, damos um clique duplo sobre a conexão que sai à esquerda e escrevemos ReservationAvailable=False.



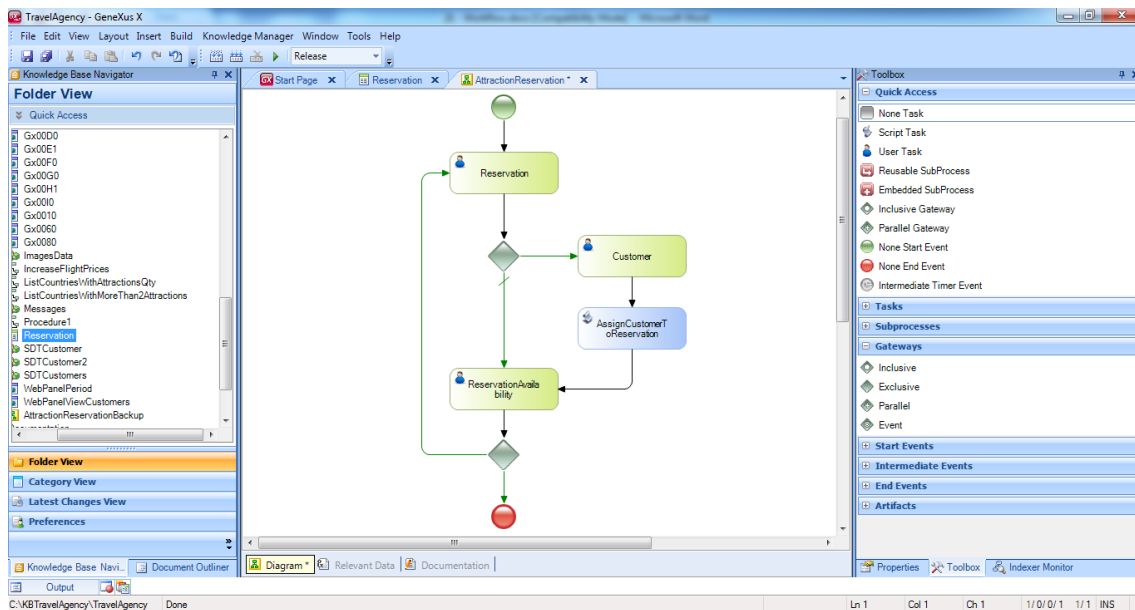
Até aqui, analisamos tudo o que acontece caso tivesse que oferecer uma alternativa de viagem ao cliente, já que a reserva não pode ser confirmada. O que nos falta fazer é considerar o caso em que a reserva se confirme.



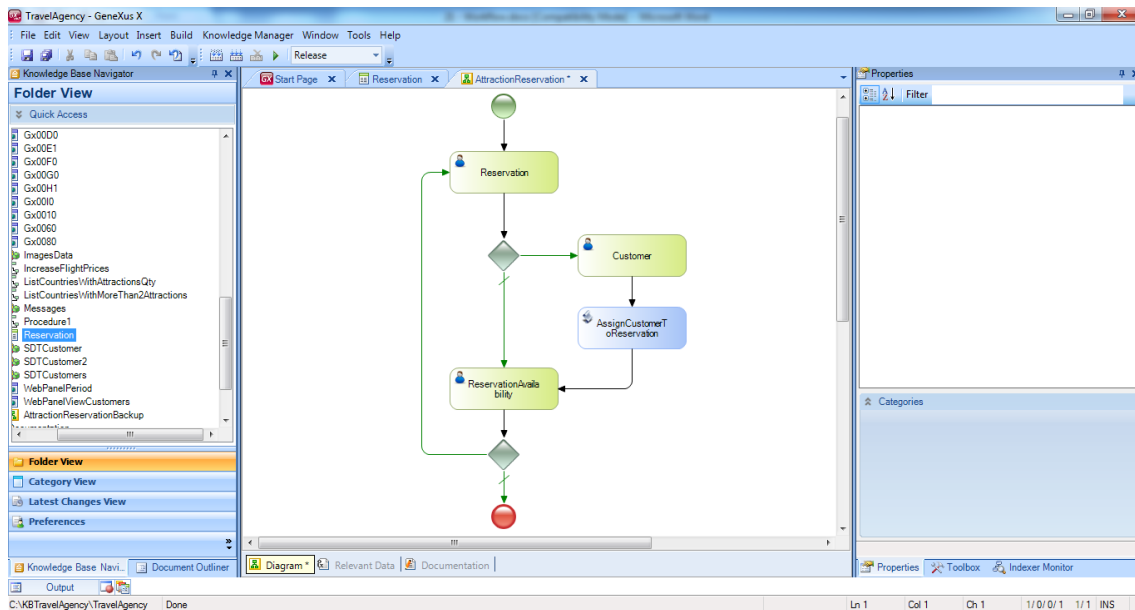
Neste caso, já não haverá mais tarefas e o processo deve ser finalizado.

Para indicar que queremos terminar o diagrama, inserimos um símbolo de NoneEndEvent tomado da caixa de ferramentas e o conectamos partindo-se do Gateway.

Esta conexão para baixo é o fluxo normal do Gateway, de modo que quando a reserva se confirma, o processo se finaliza.



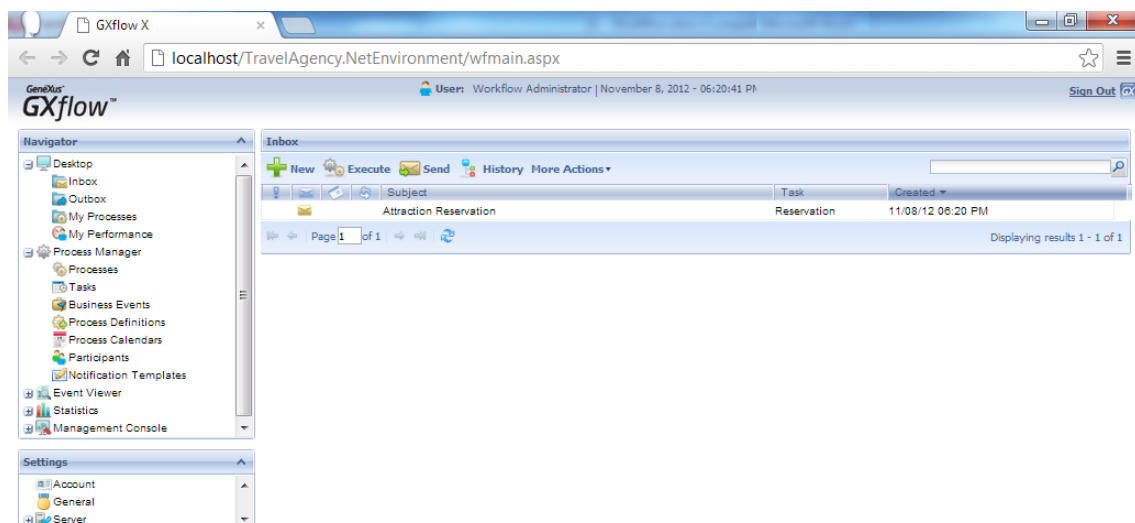
Para indicar isso, selecionamos a conexão e na janela Propriedades colocamos sua propriedade ConditionType no valor Default.



Até aqui, finalizamos o diagrama de nosso processo de negócio.

Pra testar seu funcionamento, vamos executá-lo. Sobre a guia com o nome do diagrama, clicamos com o botão direito e escolhemos Run...

Na análise de impacto, GeneXus informa que se criará a tabela Reservation, o que é o esperado, de modo que continuamos a reorganizar e a continuar com a execução.



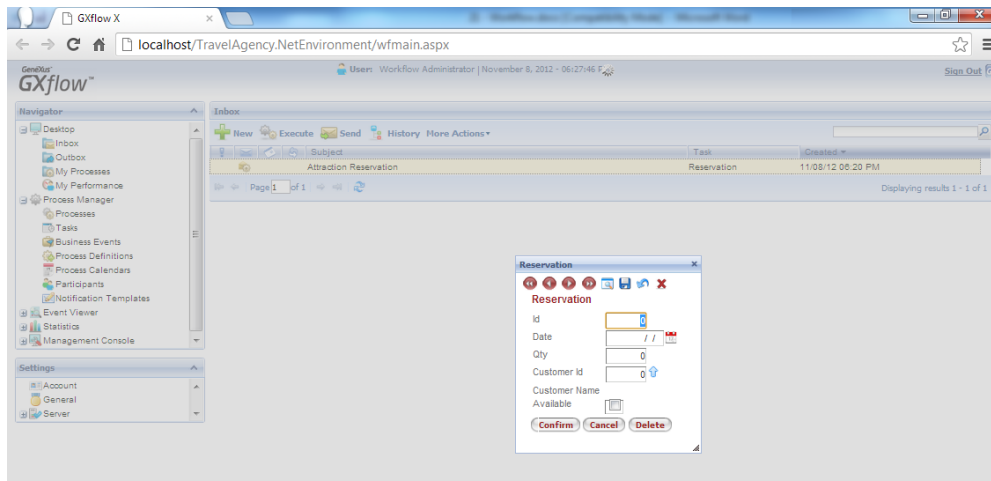
Vemos que é aberta uma janela com o Cliente GXflow, que se parece com uma caixa de entrada de e-mails, na qual são mostradas, para cada usuário, as tarefas pendentes, conforme o processo definido.



Se observarmos nessa caixa de entrada, vemos um envelope fechado e que na coluna Subject se diz: AttractionReservation, indicando que esse processo está definido e, na coluna Taks, a tarefa Reservation está pendente.

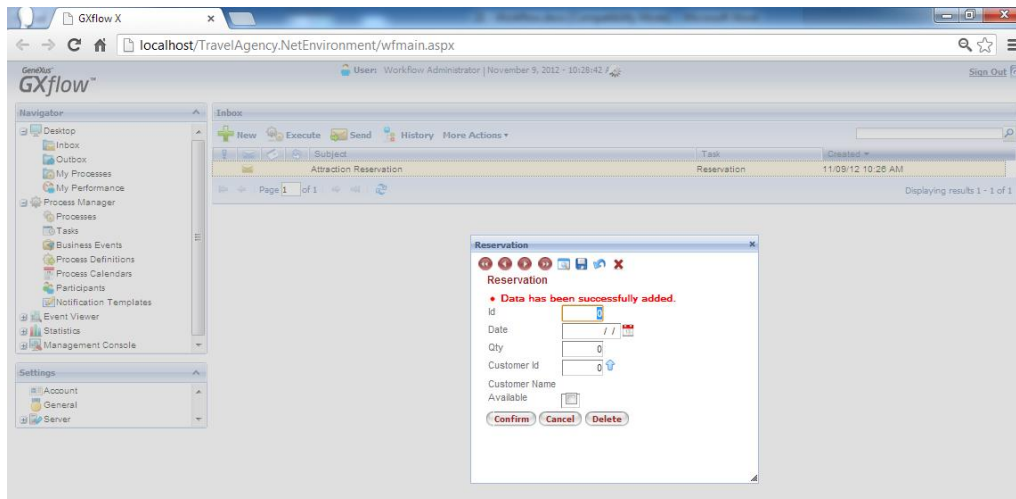
Para executá-la, selecionamos e pressionamos o botão de Execute ou clicamos com o segundo botão sobre a tarefa...

Observamos que a transação Reservation é aberta para que possamos inserir a reserva.

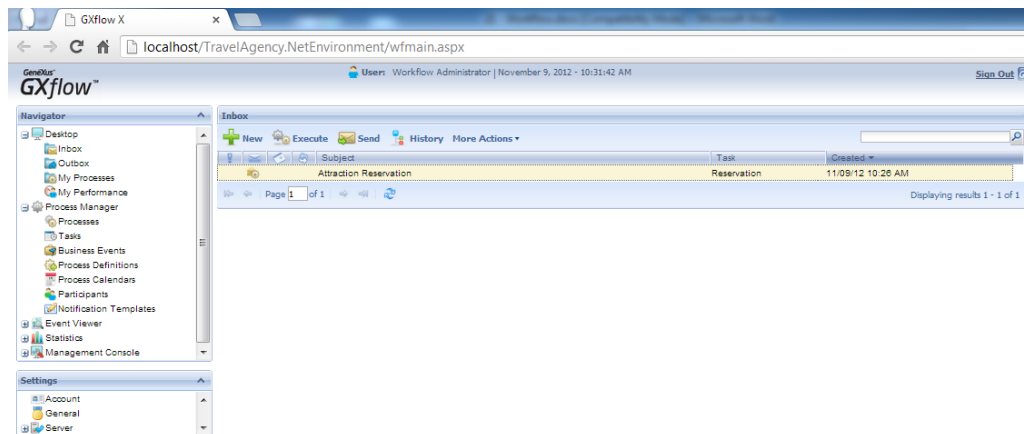


Deixamos o id em branco porque está autonumerando-se, inserimos a data do dia... definimos a quantidade de passageiros como 2... e deixamos o CustomerId em 0, porque a pessoa que deseja fazer a reserva ainda não é cliente da empresa.

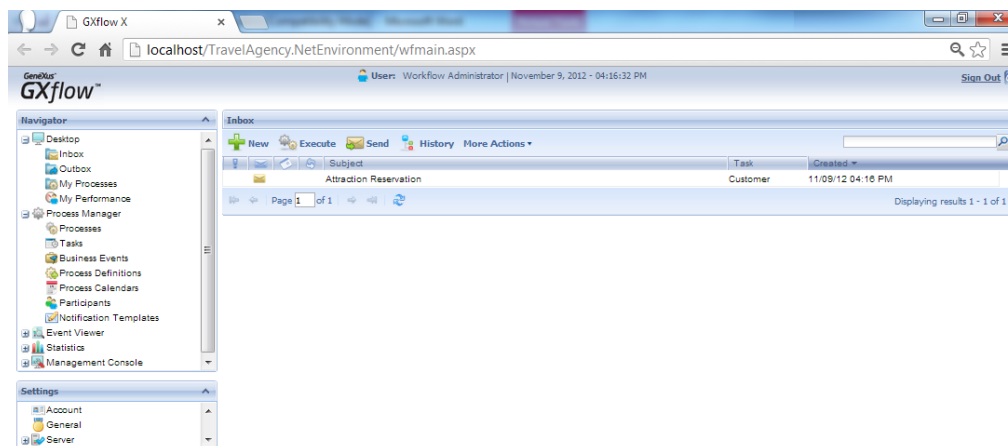
Clicamos em Confirmar.



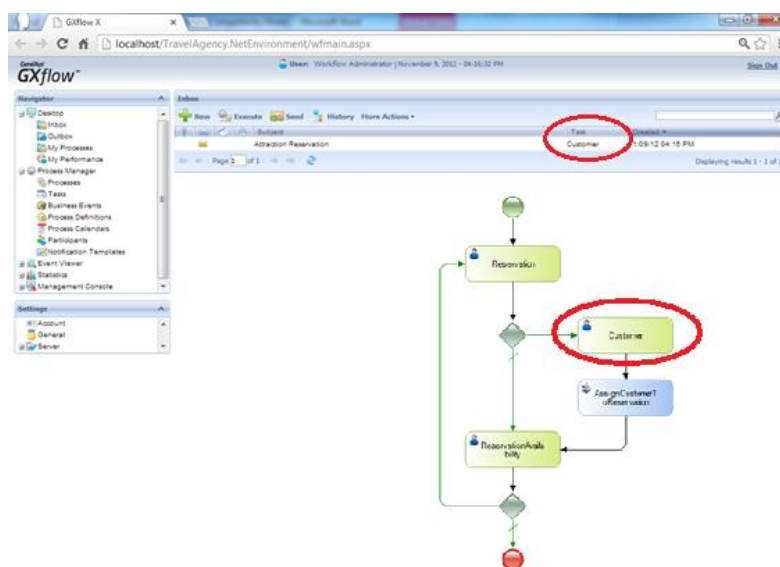
Vemos que GeneXus nos avisa que os dados foram inseridos corretamente, assim, podemos fechar a janela clicando sobre o símbolo x.



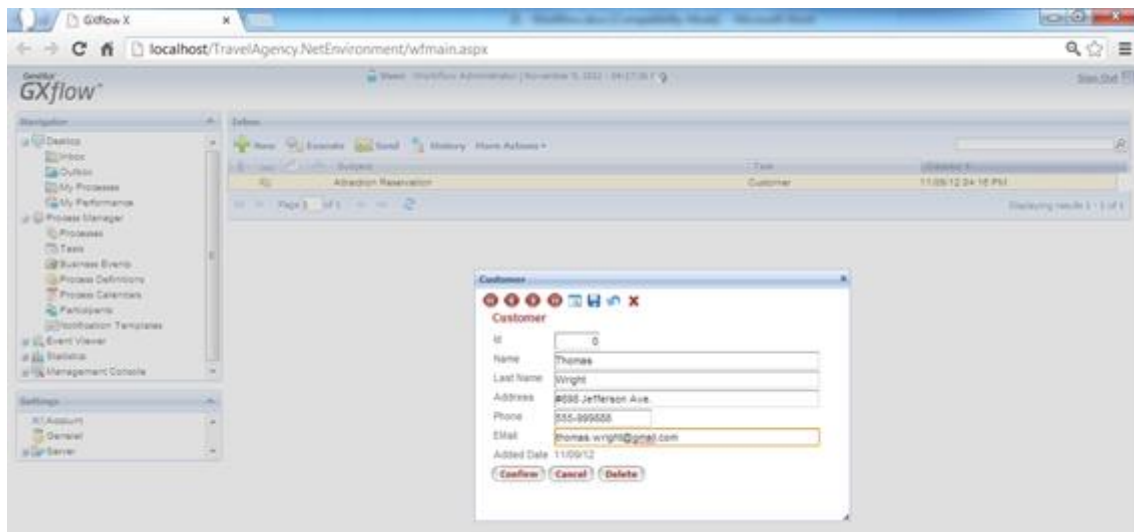
Agora, em vez de um envelope fechado, o símbolo da tarefa é um envelope, o que mostra que a tarefa já foi executada. Para passar à tarefa seguinte, pressionamos Send.



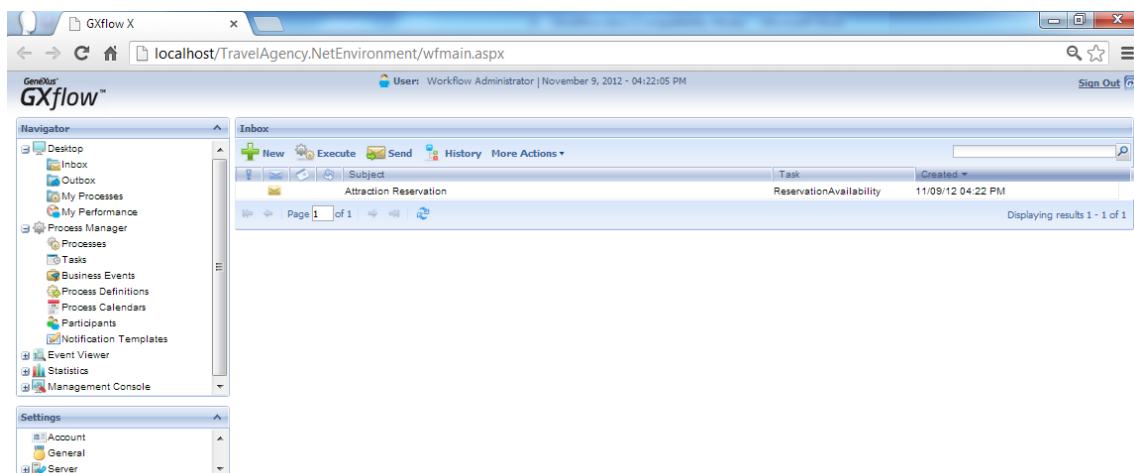
Agora, a tarefa que falta executar é a transação **Customer**, já que como não inserimos o identificador do cliente na reserva, o fluxo do diagrama seguirá à direita, abrindo a transação Customer para que possamos inserir a pessoa como cliente novo.



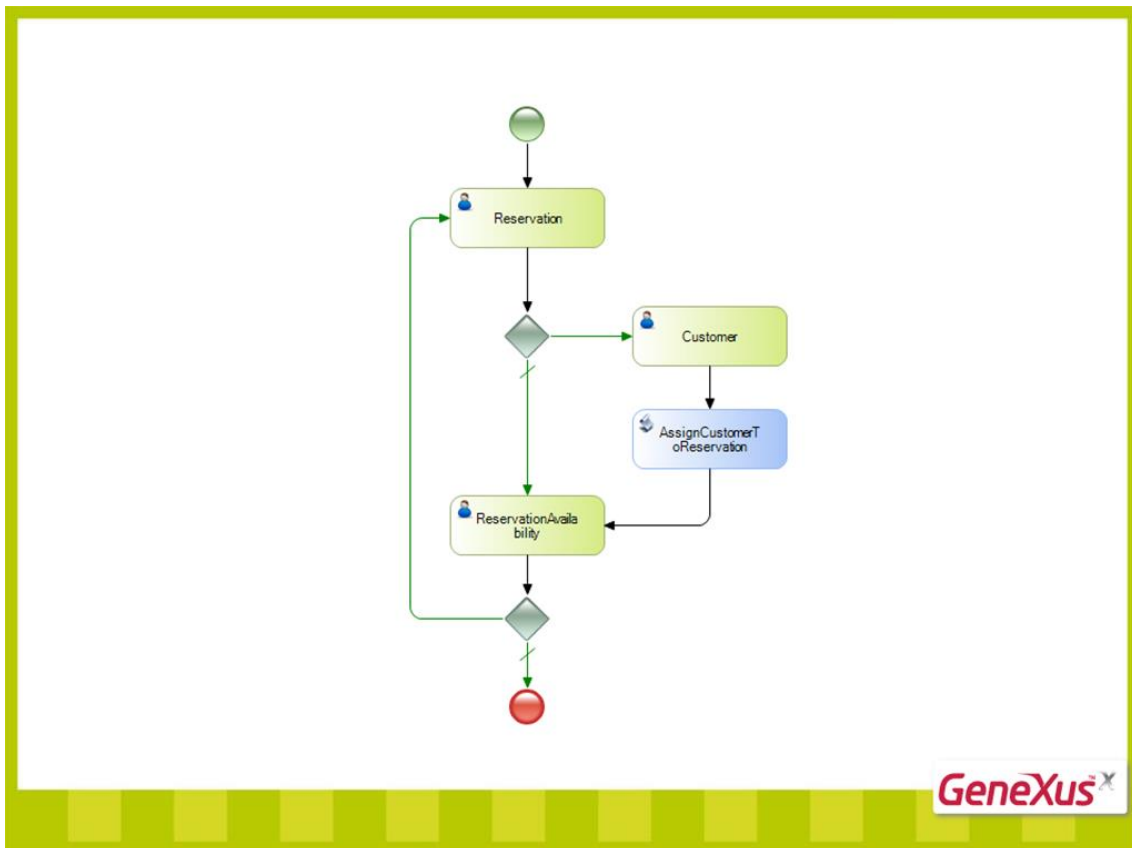
Executamos a tarefa, abre-se a tela da transação Customer e inserimos o cliente.



Confirmamos e fechamos a janela... Agora clicamos sobre o botão Send para terminar a tarefa e executar a seguinte.

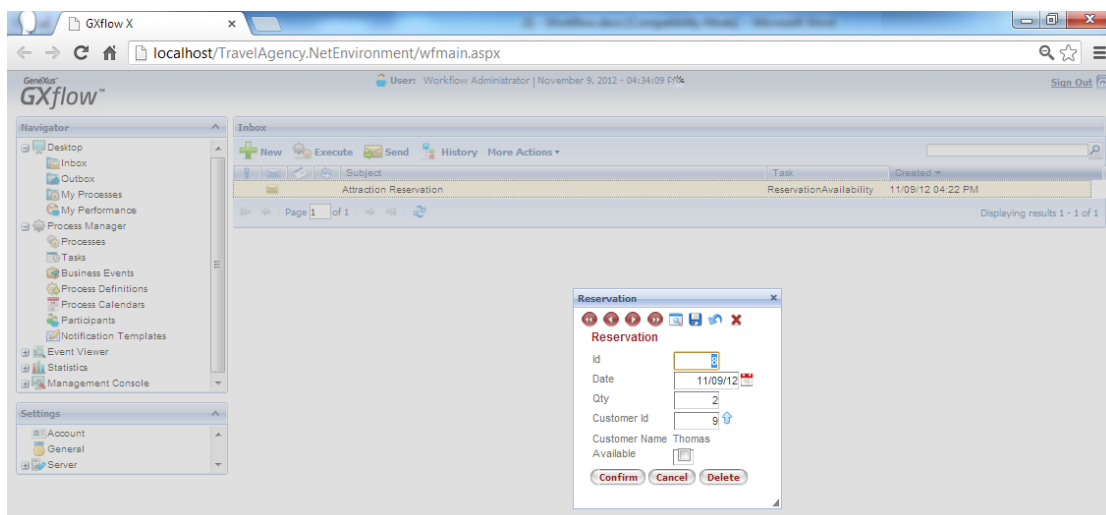


Vemos que a próxima tarefa é a chamada ReservationAvailability. Se nos lembramos do processo, a tarefa Customer invocava uma tarefa chamada AssignCustomerToReservation, que invocava um procedimento para atribuir o identificador do cliente recém inserido à reserva.

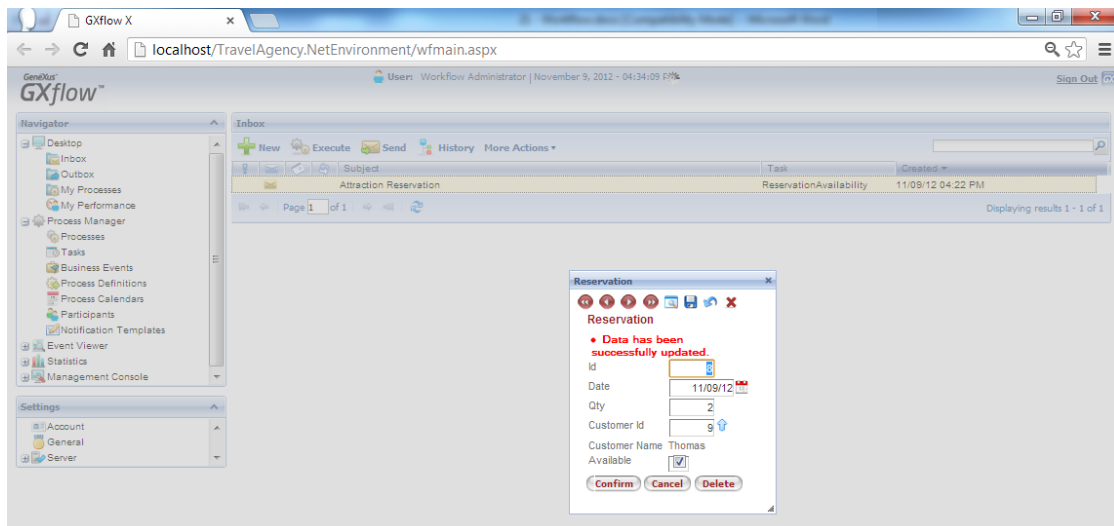


Como a tarefa AssignCustomerToReservation não é interativa, não foi apresentada na caixa de entrada quando finalizamos a tarefa Customer, mas mostrou-se diretamente a próxima tarefa na qual temos que intervir, a ReservationAvailability.

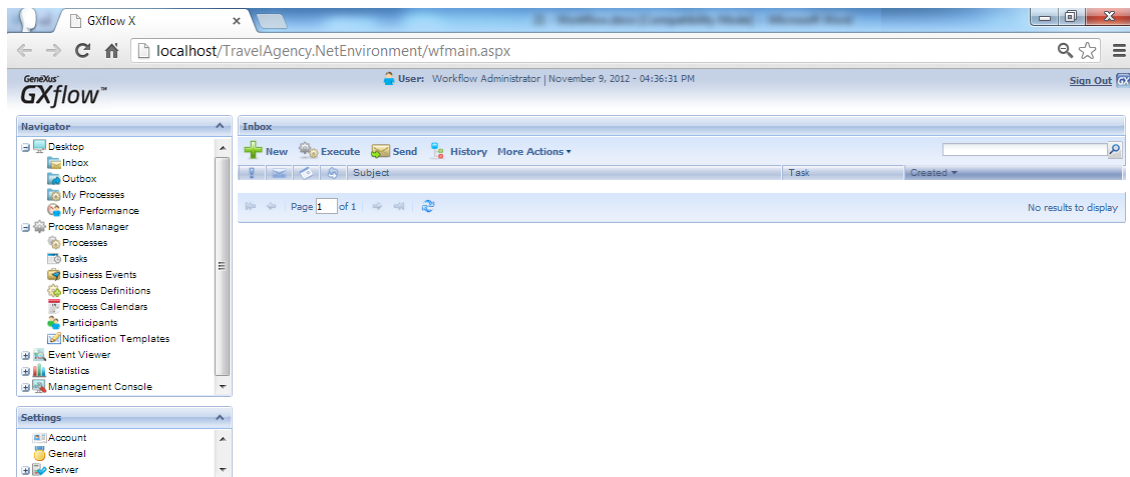
Votamos à janela GXflow e executamos a tarefa ReservationAvailability com um duplo clique sobre ela.



Agora, abre-se o formulário da transação Reservation para que possamos inserir a informação se a reserva está ou não disponível. Supondo-se que sim, marcamos a caixa de seleção e pressionamos confirmar.

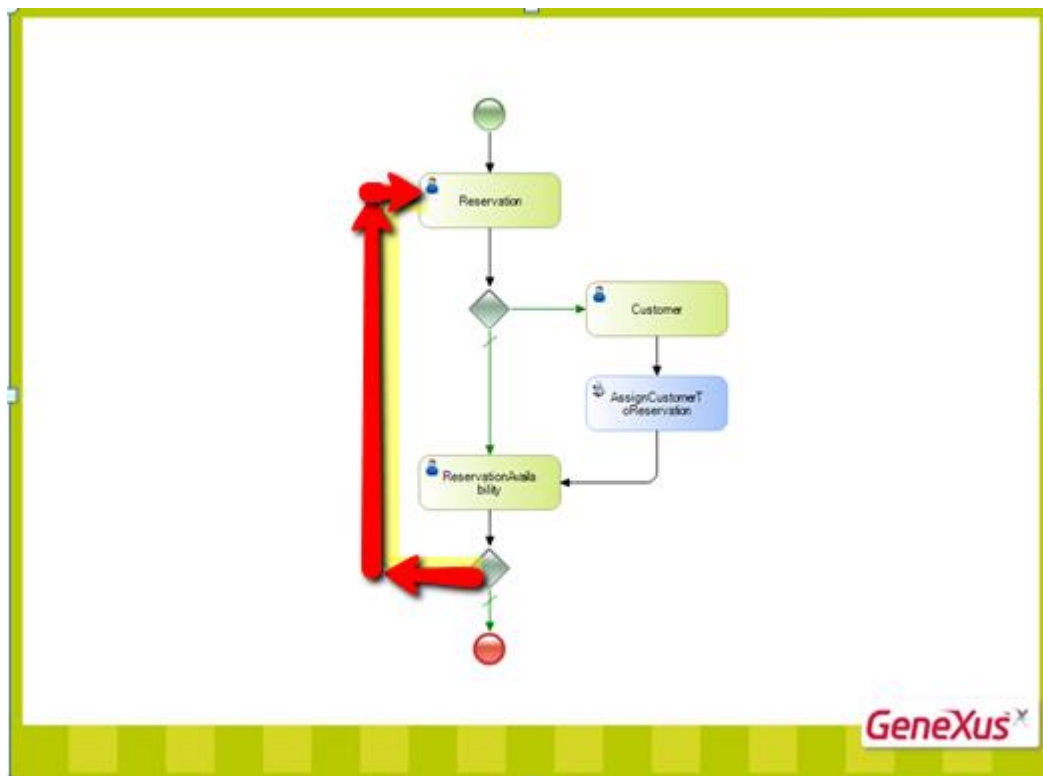


Vemos que os dados foram cadastrados corretamente, então fechamos a janela da transação e clicamos no botão Send para finalizar a tarefa.



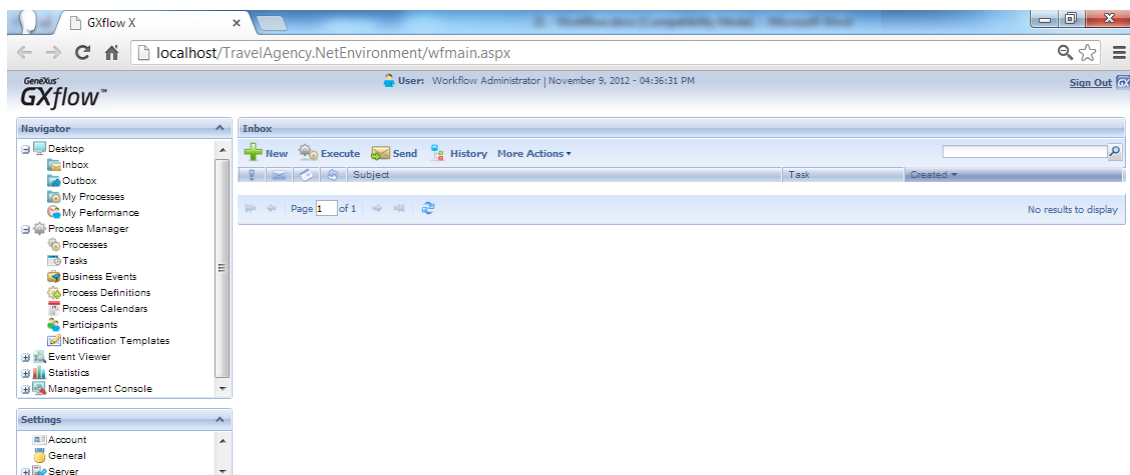
Agora, a caixa de entrada está vazia. Isso significa que não há mais tarefas pendentes, já que chegamos ao fim do processo.

Note que se não tivéssemos marcado a caixa de seleção, o workflow nos teria levado novamente à transação Reservation, ao início do diagrama.



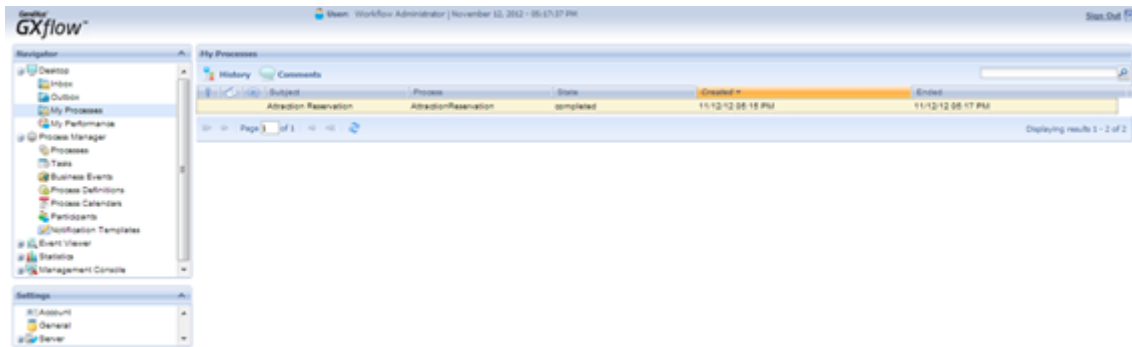
Algo que não mencionamos é que todas essas tarefas foram executadas pelo mesmo usuário.

Isso é assim enquanto desenvolvemos e testamos o processo, mas no uso real, haverá diferentes usuários que executarão as distintas tarefas, de acordo com os perfis que tenham na organização.

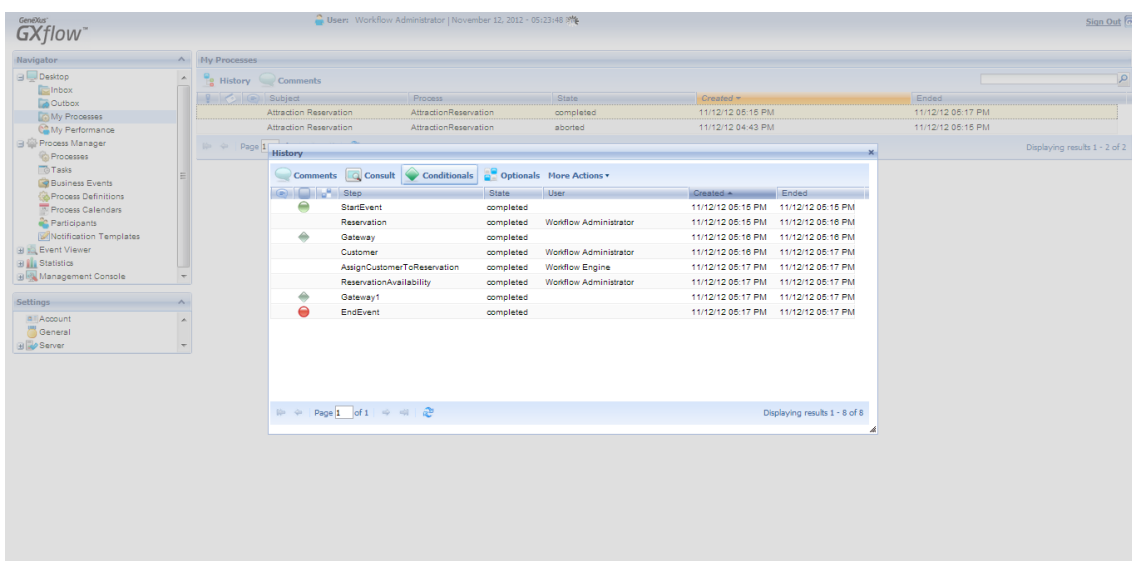


Outra coisa interessante é que podemos ver o histórico do processo, saber quais caminhos do diagrama foram sendo tomados quando o processo foi executado;

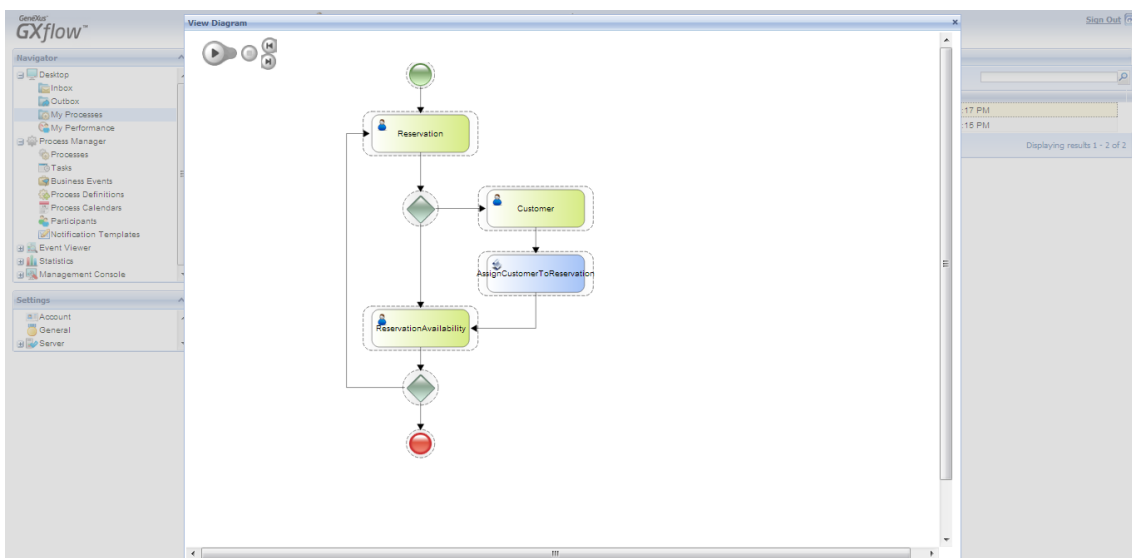
Para ver o histórico, selecionamos My Process na janela Navigator. Vemos que se mostra o processo que acabamos de executar (Attraction Reservation) que parece no estado finalizado.



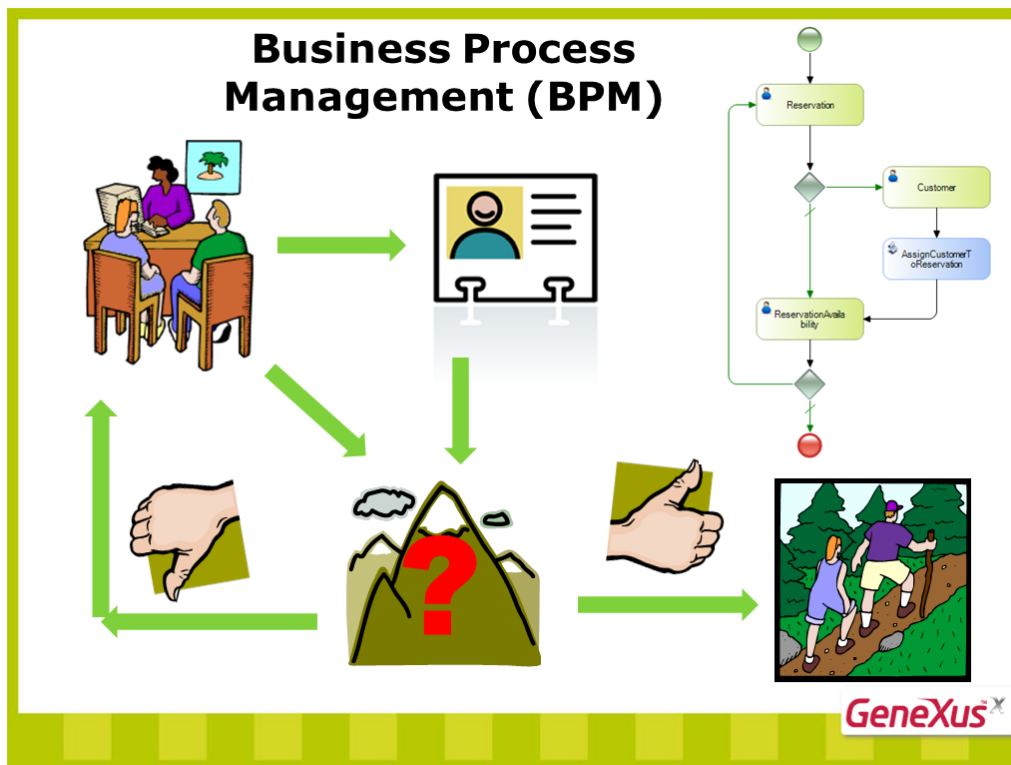
Damos um duplo clique sobre ele e vemos que é aberta uma janela que mostra o histórico do processo.



Nesta janela, vemos todas as tarefas que se executaram. Se formos a More Actions, View Diagram, podemos ver o histórico em forma de animação. Clicamos em Play.



Vimos que GeneXus nos permite modelar processos de negócios de forma intuitiva, resolvendo automaticamente quais objetos devem ser executados em cada caso, de acordo com a definição que fizemos através do diagrama.



Para saber mais sobre este tema, visite o site no endereço que se mostra na tela.

