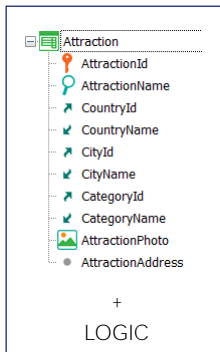


Actualización de la base de datos

Usando Business Components. Ejemplo

GeneXus™

Business Component



&attraction

AttractionId	7
AttractionName	Forbidden City
CountryId	3
CountryName	China
CityId	1
CityName	Beijing
CategoryId	2
CategoryName	Monument
AttractionPhoto	
AttractionAddress	

+

&attraction.**Load**(Pk)
 &attraction.**Insert**()
 &attraction.**Update**()
 &attraction.**Delete**()

 &attraction.**Success**()
 &attraction.**Fall**()
 &attraction.**GetMessages**()

 &attraction.Mode()
 &attraction.**Save**()
 &attraction.**InsertOrUpdate**()

En el video anterior vimos cómo insertar, modificar y eliminar un registro a partir de un business component correspondiente a una transacción de un nivel.

Backoffice new requirement

Paris



Monument



Museum



Museum

Tourist site

Ahora nos resultará sencillo implementar el siguiente requerimiento para el back-office de la agencia de viajes:

Supongamos que en un momento dado surge la necesidad de crear una **categoría especial**, "Tourist site", que agrupe a atracciones turísticas muy importantes de una ciudad, independientemente de que hasta el momento estuvieran categorizadas como monumentos, museos, etcétera.

Backoffice new requirement

Beijing



Monument

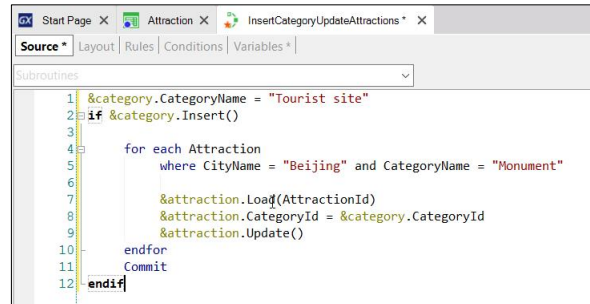
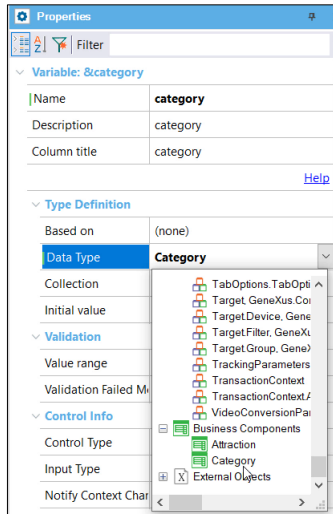


Monument

Tourist site

Empecemos por algo un poquito más sencillo: supongamos que en la agencia necesitan cambiar todas las atracciones de Beijing que antes fueran de tipo Monument, para que ahora sean de esta nueva categoría. Nos piden hacerlo por código, para no tener que hacerlo de forma interactiva, una a una a través de la pantalla de work with (aquí solo vemos dos atracciones de Beijing, pero podrían haber muchas más y además al usuario se le podría pasar por alto alguna).

Implementation



Utilizaremos el procedimiento que vimos. Borramos todo lo que habíamos codificado para empezar de vuelta.

Como vamos a tener que ingresar una nueva categoría, debemos crear su Business Component. Y ahora definir una variable de ese tipo de datos. Como el Id es auto-numerado, no le damos valor, y sí se lo asignamos al elemento `CategoryName` e insertamos.

Si todo fue bien, entonces el siguiente paso será modificar la categoría de todas las atracciones turísticas de Beijing que hasta el momento fueran monumentos.

Entonces debemos recorrer esas atracciones con un `for each`, donde la ciudad sea Beijing y la categoría Monument.

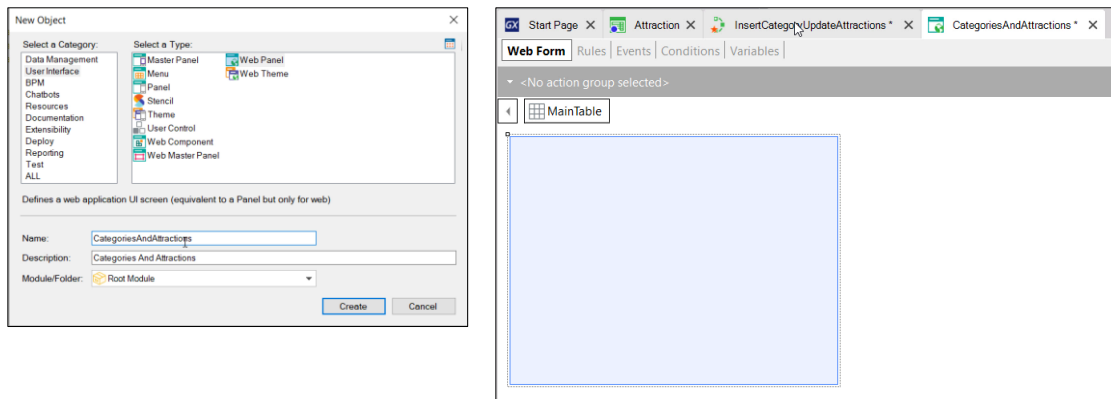
En cada iteración del `for each` estaremos posicionados sobre una atracción de esas, y lo que hacemos es cargarla en la variable `&attraction`, cambiarle la categoría por la nueva (observemos que aquí estamos recuperando el valor que le dio la base de datos al insertarse la categoría, el Id, auto-numerado). Y por último lo que hacemos es modificar el registro en la base de datos, disparándose todas las reglas de la transacción, como ya vimos.

Por último escribimos un `Commit`.

Vamos a invocar a este procedimiento desde una pantalla, por lo que le

quitamos la regla outputfile y dejamos las propiedades Call protocol y Main program con sus valores default.

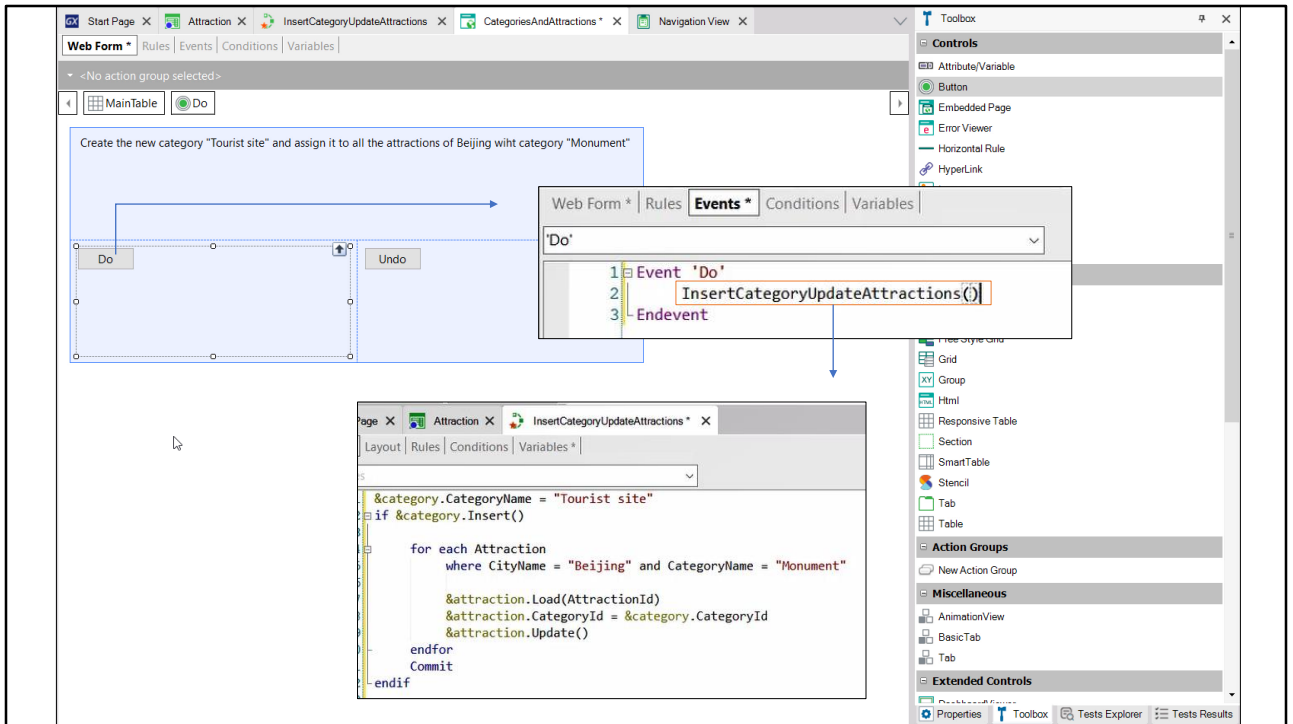
Web panel



Vamos a crear esa pantalla, ese panel. Será un objeto, del que hablaremos más adelante, de tipo Web panel, panel web.

Le llamaremos de este modo...

Aquí vemos su web form, vacío. Esto va a ser lo que va a ver el usuario en la pantalla en ejecución, en su browser.



Arrastramos desde la Toolbox un control de tipo bloque de texto, para mostrarle al usuario un mensaje, un texto, en el panel. Le colocamos este Caption, indicando que lo que se va a hacer es insertar la categoría "Tourist site" y **modificar las atracciones de Beijing**. Y arrastramos un botón, que va a tener un evento asociado de este nombre Do. Y vamos a colocar otro botón para deshacer la acción.

Queremos ir a programar el código que se va a ejecutar cuando el usuario presione el Do. Entonces vemos que el web panel tiene una sección de eventos que es donde vamos a especificar ese código. Lo que vamos a hacer aquí es invocar al procedimiento que teníamos creado. Podemos digitarlo directamente, o mejor aún, para evitar errores, arrastrarlo desde el KB Explorer. Y colocarle los paréntesis.

Vamos a probar. F5.

GeneXus Application Name

Attractions

INSERT

Id	Name	Country Id	Country Name	City Id	City Name	Category Id	Category Name	Photo	Address
9	Forbidden city	3	China	1	Beijing	2	Monument		
11	Great Wall	3	China	1	Beijing	2	Monument		

Selection List Category

Category Id

Category Name

Id Name

- 1 [Museum](#)
- 2 [Monument](#)
- 3 [Famous Landmark](#)

CANCEL

Country Name [China](#)

Veamos las categorías que tenemos ingresadas hasta el momento en la base de datos. Son estas tres.

Ahora veamos las atracciones turísticas. Vamos a re-insertar a la muralla china, que la habíamos eliminado en el video anterior.

Si filtramos por país China, vemos que tenemos aquí las dos atracciones de Beijing, que son de tipo Monument. Por tanto deberán quedar cambiadas, cuando ejecutemos el código del evento del Web panel, por la nueva categoría Tourist Site.

GeneXus Application Name

Attractions

INSERT

Id	Name	Country Id	Country Name	City Id	City Name	Category Id	Category Name	Photo	Address
9	Forbidden city	3	China	1	Beijing	5	Tourist site		
11	Great Wall	3	China	1	Beijing	5	Tourist site		

Selection List Category

Category Id

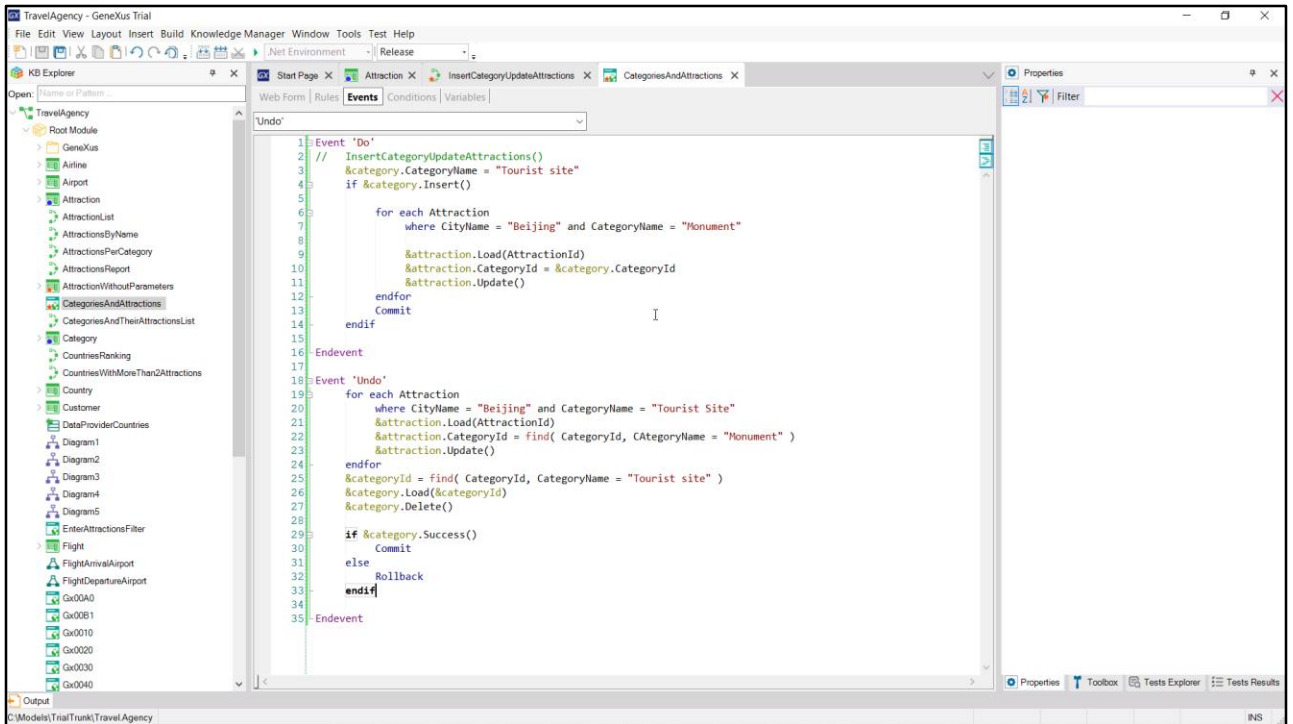
Category Name

- 1 [Museum](#)
- 2 [Monument](#)
- 3 [Famous Landmark](#)
- 4 [Tourist site](#)

CANCEL

Country Name [China](#)

Refrescamos la pantalla. Y vemos que le cambió efectivamente la categoría. Habiéndola creado previamente en la tabla.



Como mencionamos en el video anterior, las actualizaciones con business components pueden realizarse prácticamente en cualquier objeto que admita código. Así, podríamos haber escrito directamente aquí el código que teníamos en el procedimiento, sin necesidad de invocar a ningún otro objeto. Definimos las variables, simplemente, category y attraction, business components, y listo.

Vamos a programar el código asociado al evento Undo, que lo que va a querer hacer es deshacer las operaciones anteriores. Es decir, eliminar la categoría Tourist site, y a la vez volver a dejar las atracciones de Beijing que eran ahora Tourist Site con la categoría Monument.

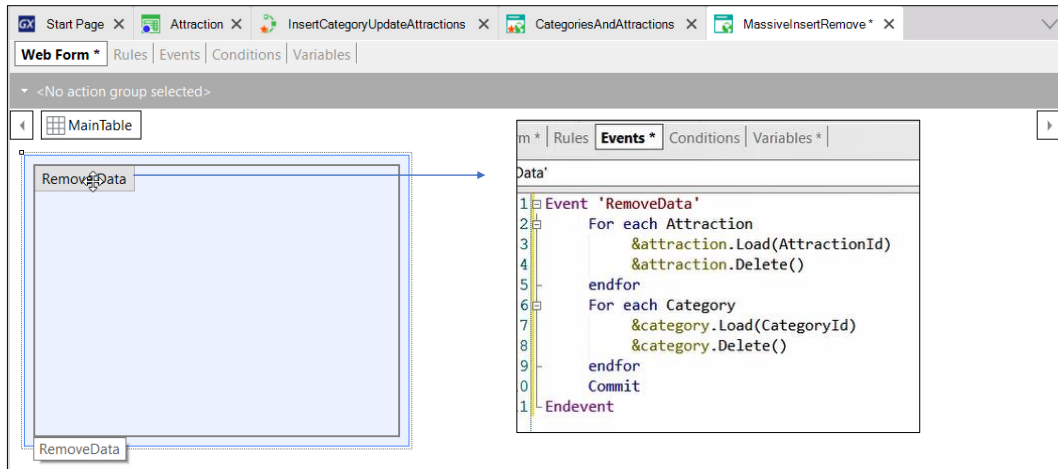
Así nos quedaría el código. Estamos recorriendo con un for each las atracciones de Beijing de categoría Tourist Site. Estamos cargando la variable attraction, business component, con cada atracción encontrada, y modificándole su id de categoría para que corresponda a Monument. Estamos dando la orden de actualizar.

Y luego es que vamos a eliminar la categoría. Definimos una variable, CategoryId, para recuperar el identificador de la categoría Tourist site, para poder cargar esa categoría en el Business component y eliminarla.

Si todo anduvo bien, entonces realizamos un Commit, y en caso contrario un Rollback.

¿Por qué definimos este orden, es decir, primero recorrer las atracciones para luego eliminar la categoría, y no al revés? ¿Qué pasaría si quisiéramos primero eliminar la categoría y luego modificar esa misma categoría de las atracciones turísticas? Evidentemente no nos lo permitirá. Fallará la integridad referencial, porque estaremos intentando eliminar una categoría que tiene registros asociados. Así que el orden no puede ser el inverso.

Probemos. Presionemos Undo. Vemos que la categoría fue efectivamente eliminada. Y si vamos a las atracciones y filtramos por China vemos que le restituyó la categoría Monument a las dos atracciones.



Ahora vamos a crear otro panel para eliminar toda la información de la tabla de categorías y toda la tabla de atracciones turísticas.

Otra vez nos aparece el web form vacío. Arrastramos un botón, le asociamos el evento RemoveData y lo codificamos de esta manera.

Otra vez el orden tiene que ser este. No podríamos recorrer las categorías e ir las eliminando, para recién después pasar a eliminar las atracciones. Fallaría la integridad referencial.

Probemos. Presionamos Remove Data. Y ahora si vamos a ver la información en la transacción Category vemos que está vacía. Lo mismo si vamos a ver las atracciones.

Comment

Category and Attraction Business Component in order to insert a new category ("Tourist Site") and change Beijing "monument" attractions to the new category

Pending Commits (7/7) Ignored Objects

<input checked="" type="checkbox"/>	Name	Type	Description	Modified On	Module	Local State	Last Synchronized	User
<input checked="" type="checkbox"/>	Attraction	Transaction	Attraction	10/7/2020 1...	Root Module	Modified	7/7/2020 16:36	ARTECHCFerna...
<input checked="" type="checkbox"/>	AttractionWith...	Transaction	Attraction Without Para...	7/7/2020 17...	Root Module	Inserted	3/7/2020 15:34	ARTECHCFerna...
<input checked="" type="checkbox"/>	CategoriesAn...	Web Panel	Categories And Attractio...	13/7/2020 1...	Root Module	Inserted	3/7/2020 15:34	ARTECHCFerna...
<input checked="" type="checkbox"/>	Category	Transaction	Category	13/7/2020 1...	Root Module	Modified	3/7/2020 15:34	ARTECHCFerna...
<input checked="" type="checkbox"/>	InsertCategor...	Procedure	Insert Category Update...	13/7/2020 1...	Root Module	Inserted	3/7/2020 15:34	ARTECHCFerna...
<input checked="" type="checkbox"/>	MassiveInsert...	Web Panel	Massive Insert Remove	13/7/2020 1...	Root Module	Inserted	3/7/2020 15:34	ARTECHCFerna...
<input checked="" type="checkbox"/>	WorkWithAttr...	Work With for Web	Work With Attraction	7/7/2020 16...	Root Module	Modified	3/7/2020 15:34	ARTECHCFerna...

Remind me to move changes to...

Por último enviemos los cambios realizados al server.

GeneXus[™]

training.genexus.com
wiki.genexus.com