

GX

GeneXus by Globant

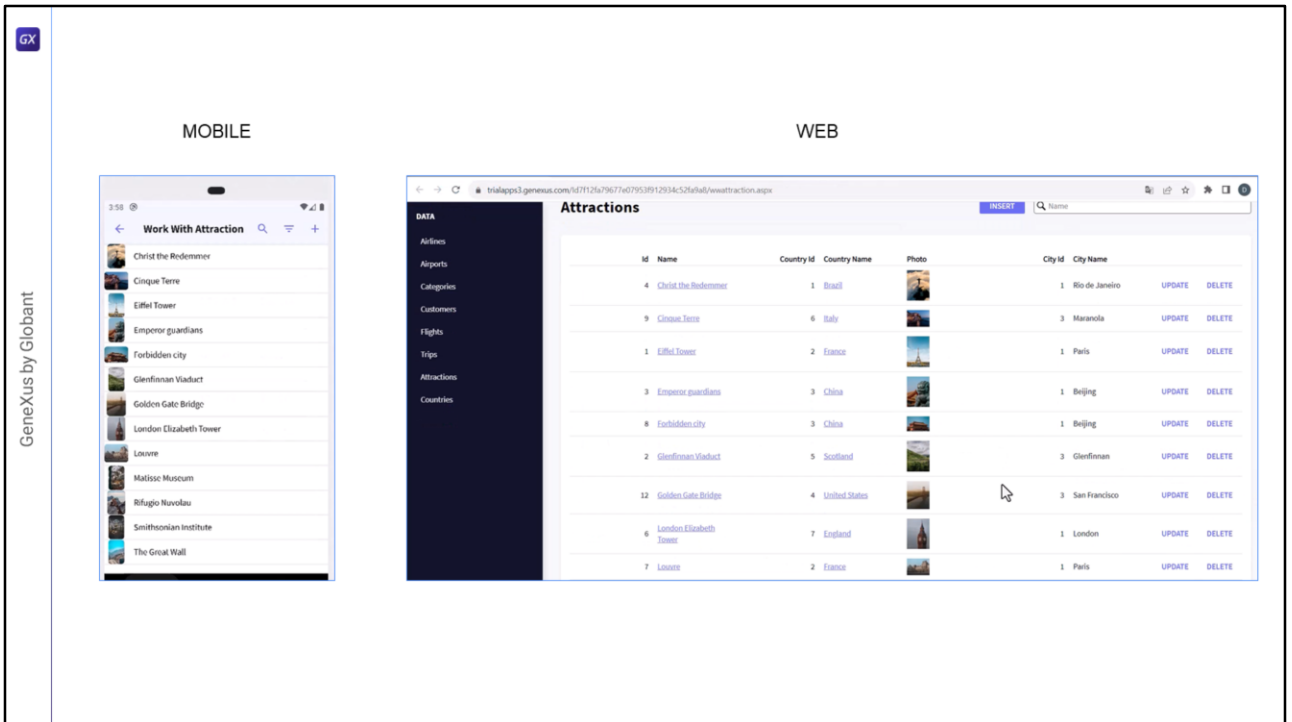
**GeneXus**<sup>™</sup>  
by Globant

[training.genexus.com](https://training.genexus.com)

# Offline Database Generation



Diego Marranghello



En este video veremos como convertir una aplicación Online en una aplicación Offline.

Primero vamos a ver como funciona actualmente nuestra aplicación Travel Agency, que actualmente es una aplicación Online.

Vamos a acceder a la aplicación mobile y a la aplicación Web al mismo tiempo, y vamos a tener ambas interfaces abiertas para ver como es la interacción.

A modo de poder mostrar un ejemplo claro y sencillo a la transacción Attraction le aplicamos ambos patrones Work With: para web y para mobile.

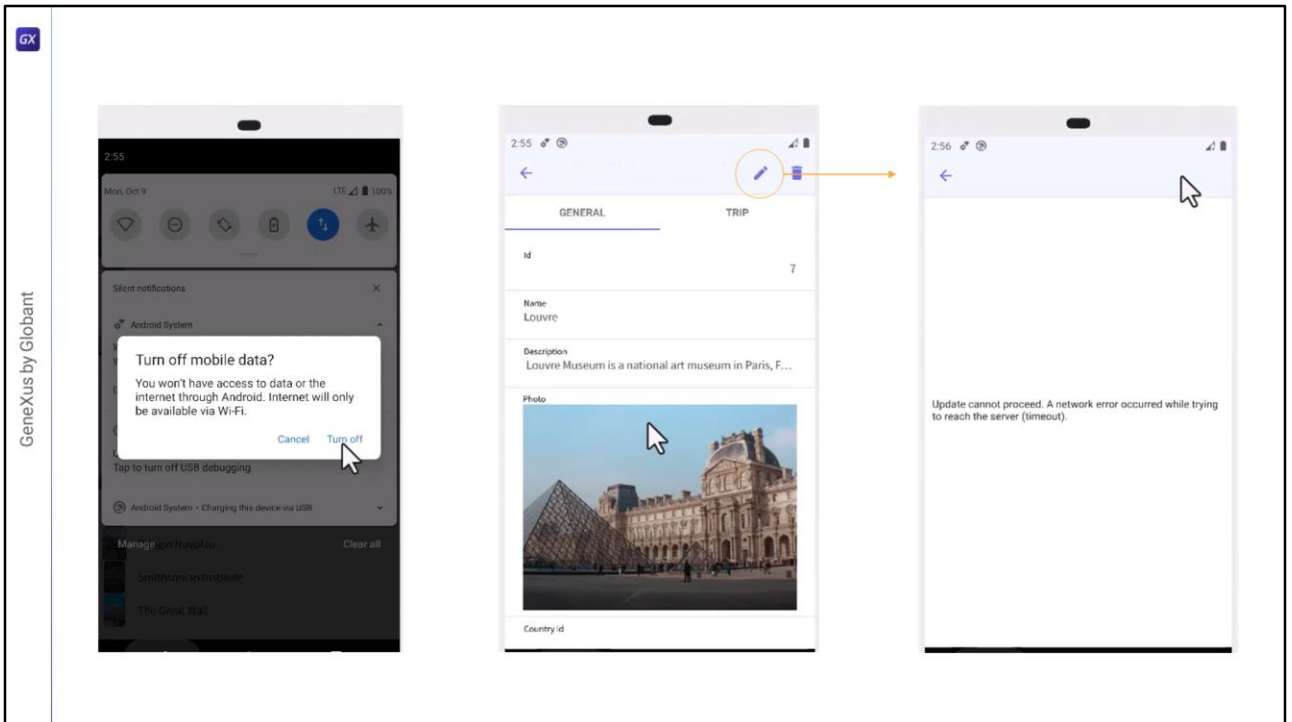
Ingresamos primero por web, ingresamos a una atracción, por ejemplo Louvre, editamos su nombre y grabamos.

Ahora vamos a la aplicación mobile, refrescamos y automáticamente vemos el cambio realizado.

¿Qué pasa si lo editamos desde el dispositivo?

Vamos a volver a editar el nombre para dejarlo como estaba en un principio, y grabamos. Refrescamos en el panel web y automáticamente vemos el cambio.

Esto es porque la aplicación trabaja en forma conectada.



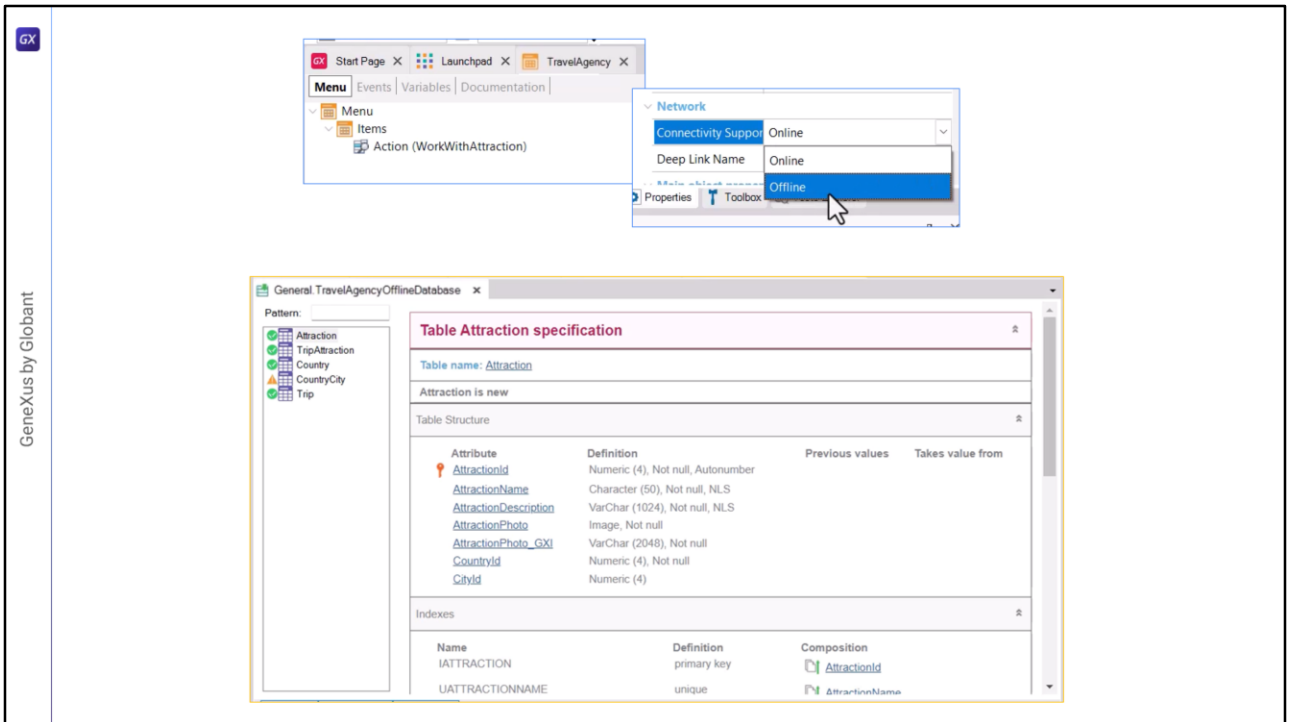
¿Qué pasa por ejemplo si ahora desconecto la aplicación?  
Vamos a simular esto, apagamos el acceso Wifi y apagamos los datos. Ahora el emulador no posee conexión a internet, por lo que no puede acceder a la capa de servicios REST.

Si tratamos de ingresar a la atracción, vamos a poder porque estos datos están cacheados del acceso anterior, pero si intento editar nos da un error.

Si tratamos a una atracción que no hayamos accedido antes no podemos, ya que no tenemos conexión y tampoco tenemos sus datos guardados.

Cuando estamos sin conexión en una aplicación Online, solo tenemos acceso a datos están cacheados en la aplicación, pero no podemos acceder a nuevas pantallas.

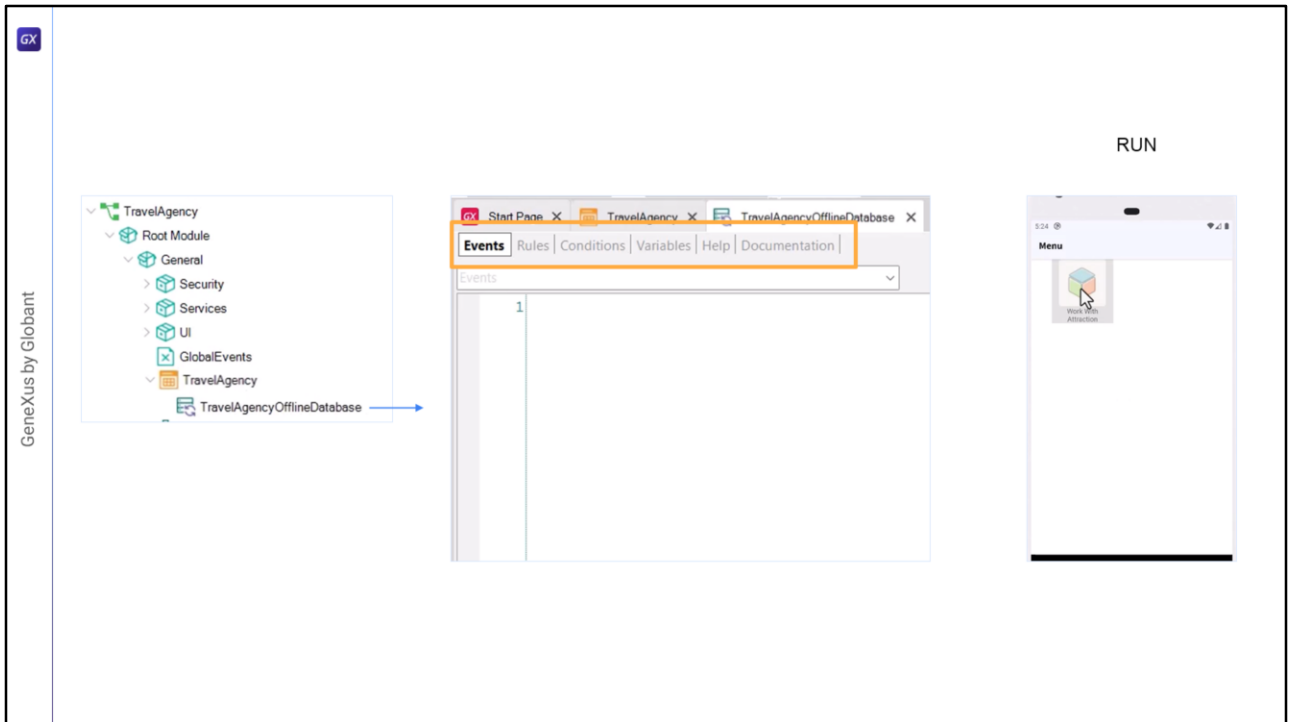
Bien, vamos ahora a GeneXus así convertimos la aplicación en una aplicación Offline.



En nuestra aplicación tenemos un objeto Menu de nombre TravelAgency, que es el objeto main de la aplicación. Y dentro, sólo tenemos, al momento, el work with de atracción.

Al ir a las propiedades del objeto menu, vemos una de nombre Connectivity Support la que por defecto tiene el valor Online, y cambiaremos a Offline. Guardamos los cambios y hacemos un build de la aplicación.

Una de las cosas que veremos a continuación es un análisis de impacto de la base de datos Offline, estas serán las tablas que se llevaran y se crearan en el dispositivo.



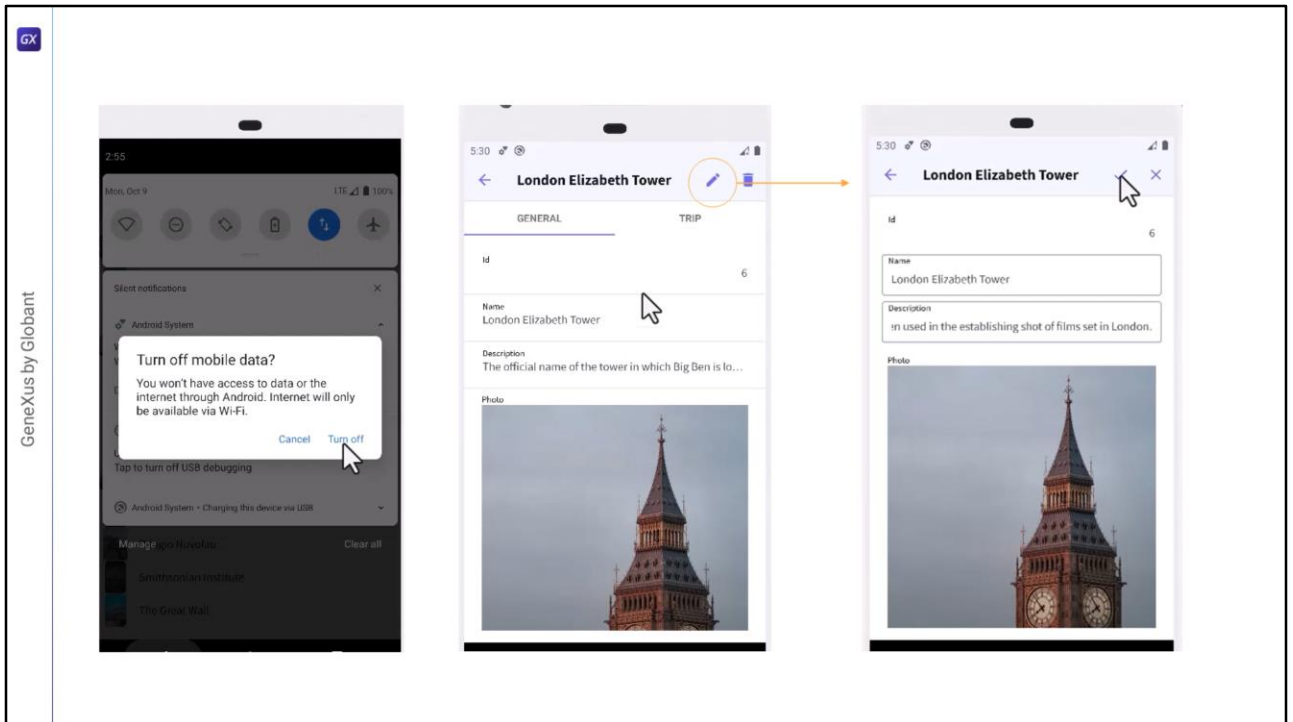
Vamos a ver que ahora tendremos un objeto nuevo, TravelAgencyOfflineDatabase que queda asociado al objeto menu TravelAgency.

Este objeto nos permitirá configurar ciertas cuestiones relativas a la base de datos Offline, la que se generará en el dispositivo, como por ejemplo la granularidad. Luego estudiaremos estas propiedades mas detalladamente.

Además este objeto tiene Eventos, Reglas, Conditions y variables, luego veremos como se utilizan, por ahora dejaremos todo default.

Veamos que ahora cuando se lanza la aplicación mobile, podemos ver que se esta ejecutando la sincronización.

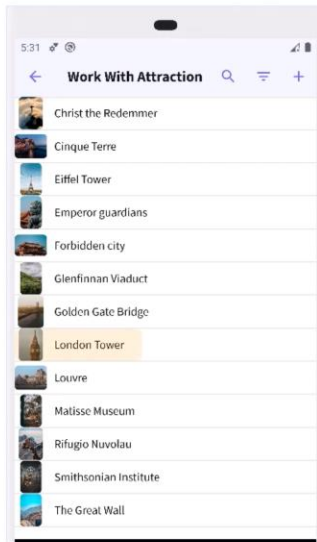
Una vez que termina el proceso, podremos utilizar la aplicación normalmente.



Ingresamos a ver las atracciones, veremos la lista.  
Vamos a desconectar el dispositivo.

Y ahora intentemos ingresar a una atracción, y no tenemos el error que nos daba antes, cuando estábamos conectados y perdíamos la conexión. Esto es porque ahora los datos están en el dispositivo.  
Incluso puedo editar los registros.

Para el usuario no hay ninguna diferencia con respecto a estar conectado.



### Attractions

Id	Name	Country Id	Country Name	Photo
4	<a href="#">Christ the Redemmer</a>	1	<a href="#">Brazil</a>	
9	<a href="#">Cinque Terre</a>	6	<a href="#">Italy</a>	
1	<a href="#">Eiffel Tower</a>	2	<a href="#">France</a>	
3	<a href="#">Emperor guardians</a>	3	<a href="#">China</a>	
8	<a href="#">Forbidden city</a>	3	<a href="#">China</a>	
2	<a href="#">Glenfinnan Viaduct</a>	5	<a href="#">Scotland</a>	
12	<a href="#">Golden Gate Bridge</a>	4	<a href="#">United States</a>	
6	<a href="#">London Elizabeth Tower</a>	7	<a href="#">England</a>	
7	<a href="#">Louvre</a>	2	<a href="#">France</a>	

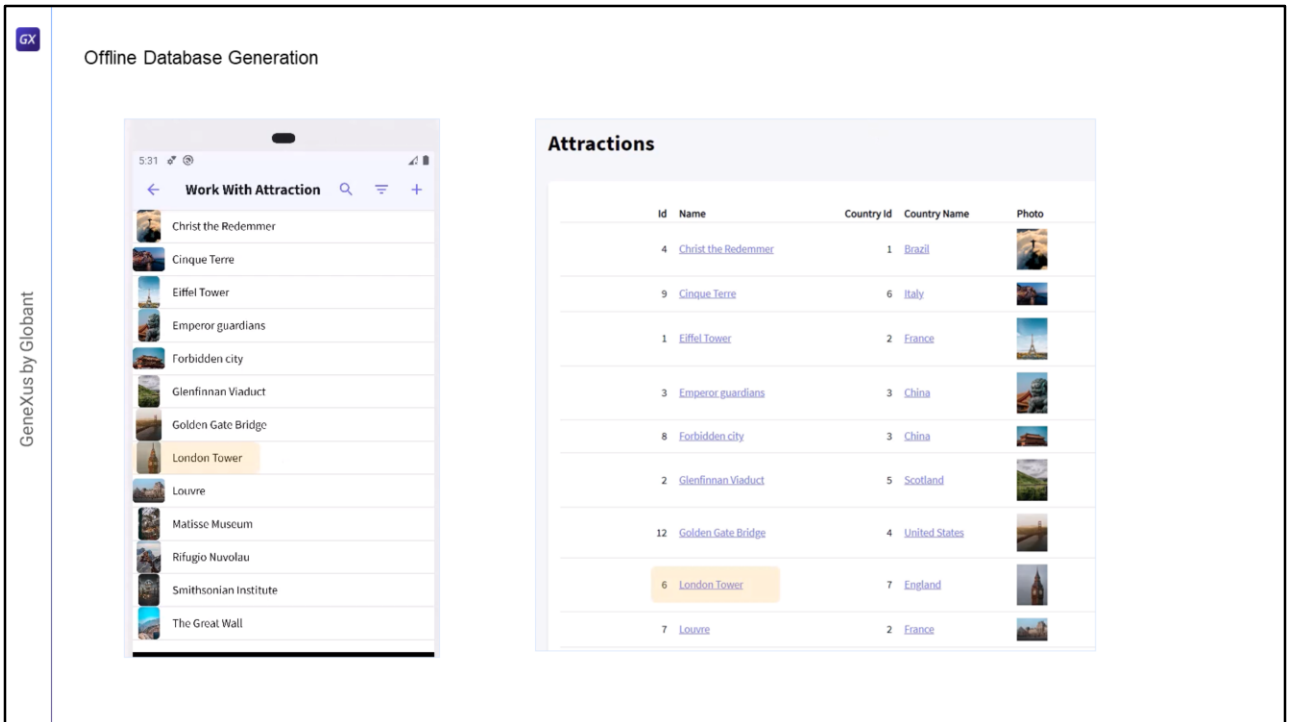
Volvamos a editar el nombre de una atracción, grabamos.

Noten que no tenemos ningún problema por estar offline. los datos son grabados siempre en el dispositivo.

Y si vamos a la interfaz web y refrescamos no veremos ese cambio. Porque el dispositivo esta offline.

D





Vamos a volver a conectar el emulador a la red.

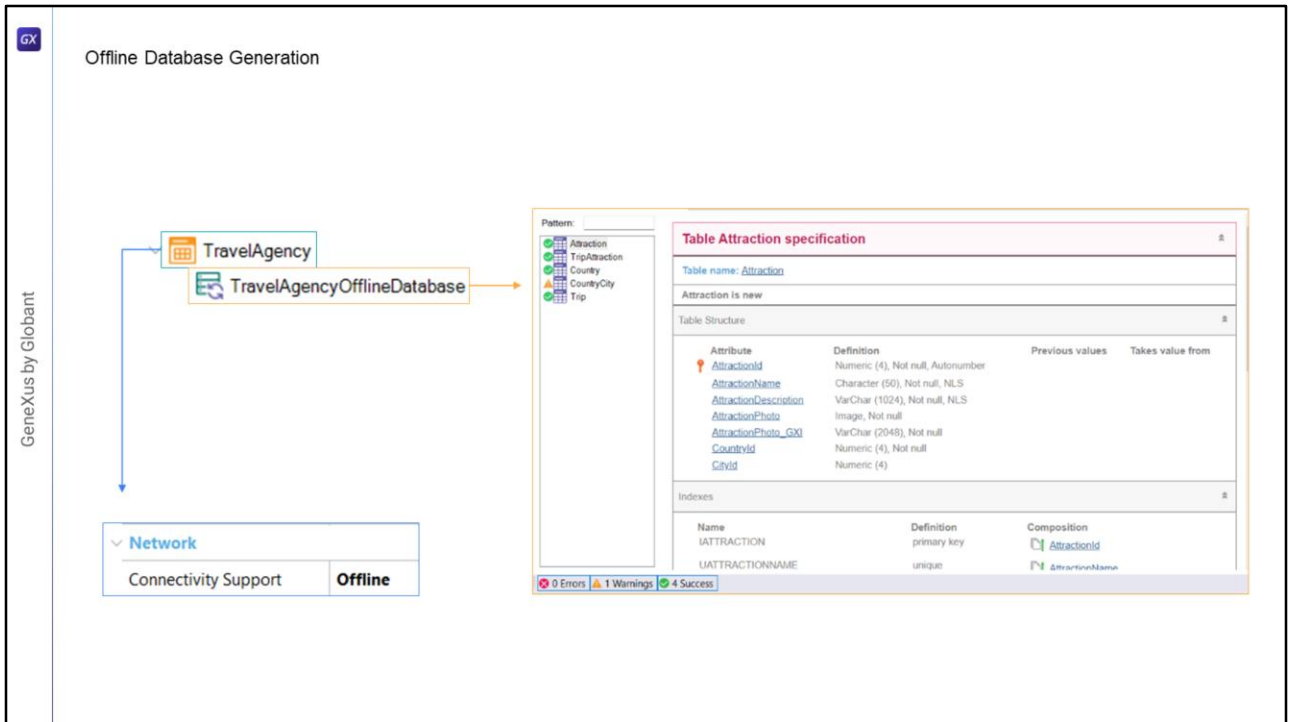
Y para forzar la sincronización voy a cerrar y abrir la aplicación nuevamente, esto es porque aun no configuramos nada del objeto OfflineDatabase.

Al abrir nuevamente ejecuta la sincronización.

Y ahora si, vamos a la web y actualizamos, y ahí vemos el cambio.

Terminamos aquí esta Demo básica, donde vimos lo simple y fácil que es crear una aplicación Offline en GeneXus.

Además hemos visto el proceso de sincronización de los datos entre el dispositivo trabajando en forma Offline y el servidor web, y las operaciones de CRUD que siempre se realizaran sobre la base de datos local.

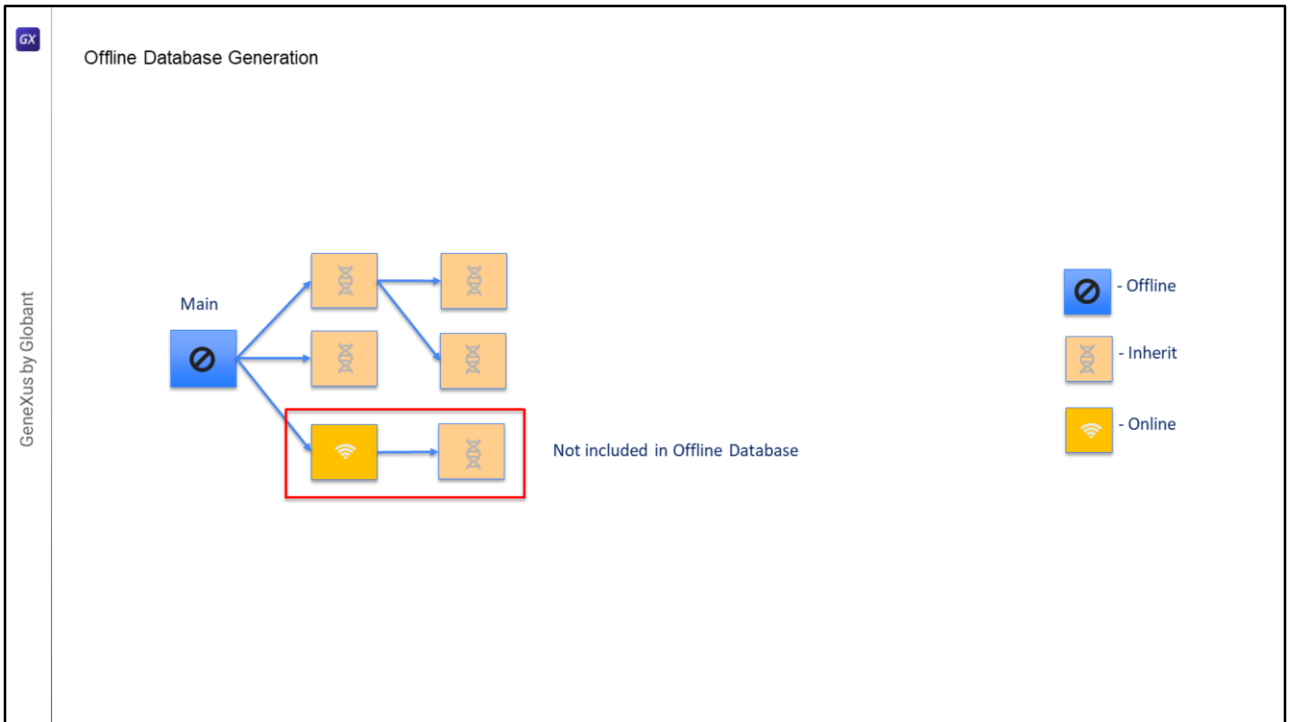


Como pudimos ver recién, cuando indicamos que el objeto main era Offline y realizamos el Build, se generó automáticamente un objeto OfflineDatabase asociado al objeto Main. Vimos también que nos presenta un análisis de impacto, similar a cuando hacemos el create de la base de datos, pero estas tablas que se muestran son las tablas que analizó GeneXus que debe llevar a la base de datos SQLite en el dispositivo.

Al crear el objeto OfflineDatabase, se crean también en el lenguaje nativo del dispositivo, los programas para crear esta base de datos local.

Este objeto es el encargado de determinar cuándo se produce la sincronización, cuáles son las tablas que se crearán en la base de datos local y también cuáles son los datos que se llevan a las mismas cuando se sincronizan con las tablas del server.

En el objeto OfflineDatabase además es donde se configuran las propiedades que vimos para el Send y el Receive.



Ahora bien, ¿Qué tablas efectivamente se llevan cuando una aplicación pasa a ser offline?

Supongamos que tenemos este árbol de llamadas desde el objeto Main, podemos tener objetos con la propiedad Connectivity Support Offline, como el Main, otros con el valor default que es Inherit, que hereda el tipo de acuerdo al objeto que lo llame y otros que pueden ser Online.

Si tenemos este árbol de invocaciones, entonces las tablas que se van a llevar al dispositivo son todas las de los objetos Offline y los que heredan la conectividad offline del que los invocó.

Pero también se va a contemplar la integridad referencial y los atributos mencionados, tanto en paneles como en prompts.

Este objeto que es Online que llama a este otro que hereda el tipo de conexión, no se incluirían, van a ser online.

En los próximos videos veremos mas detalles sobre la sincronización y las propiedades del objeto OfflineDatabase

GX

GeneXus by Globant

**GeneXus**<sup>™</sup>  
by Globant

[training.genexus.com](https://training.genexus.com)