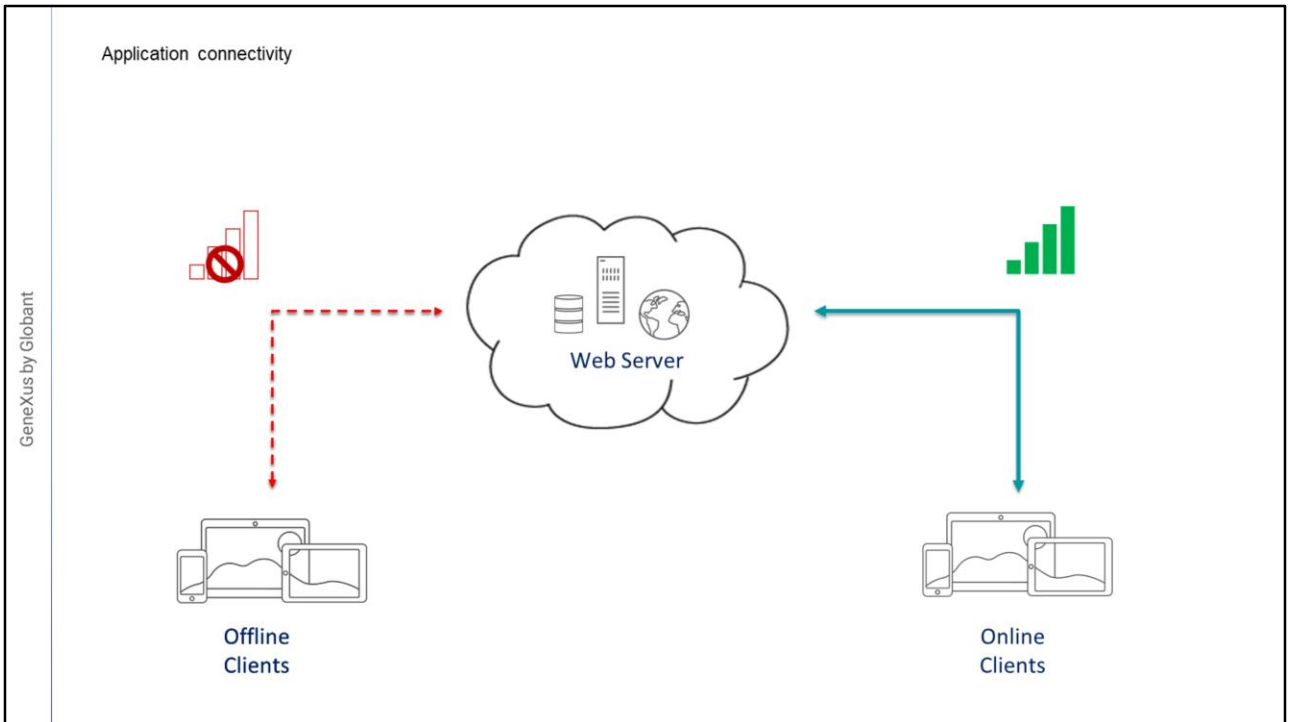# Introduction to Offline Applications

Diego Marranghello

GeneXus by Globant

GX

So far, we've seen mainly applications that have to be connected to the web server in order to work, accessing the REST services and through them the database hosted on the server.

However, GeneXus also allows us to create mobile applications that can work partially connected or even completely disconnected from the web server.

Next, we will study these cases.

Online Applications

Web Server

Online Clients

GeneXus by Globant
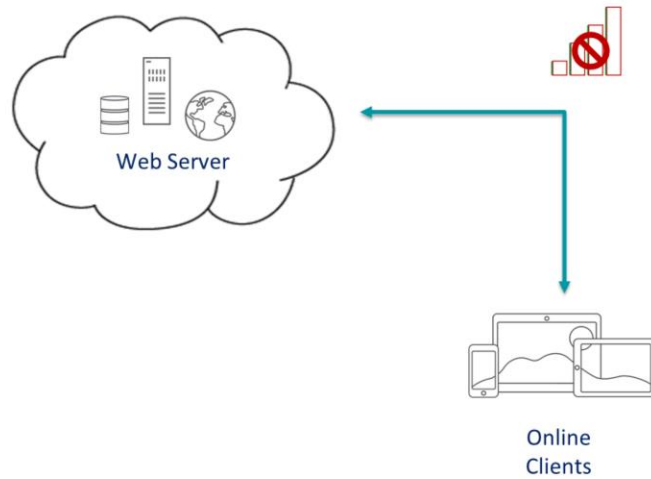
Let's review the concepts around connected applications.

Online applications are those that must always be connected to the Internet in order to retrieve data and work with it.

When the device is offline, the app will only be able to work with cached data, and it will not be able to browse new screens or update the data.

The application needs to access the web server's service layer in order to run the data providers that return the data and the business components that perform CRUD operations over the database.

The data cannot be obtained without an Internet connection because there isn't a local database, and all the data is on the server.

Online Applications

GeneXus by Globant

Web Server

Send

Receive

Local DB

Clients
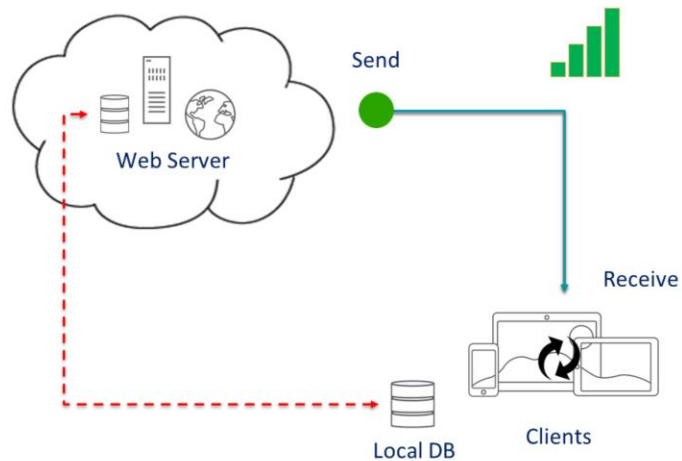
Now, let's see the case of partially connected applications.

It may happen that part of the application keeps running when it is disconnected from the Internet, while the other part must stay connected in order to work.

Suppose we have an application for our GeneXus annual event, and we will want users to be able to see the entire conference schedule, and all related information, even if they lose connection.

Then, when the connection is restored, the application will automatically update the data (which will be stored in a local database in the device), synchronizing with the server to receive updated data.

One of the functionalities of this application is that users can select conferences as favorites, and in this case it will also send this information to the server when synchronized, updating the central database.

However, some tasks will require accessing the web server, whether due to their sensitivity, or due to the frequency with which the data is changed.

These tasks must be executed online.

Therefore, we can select which objects of the application can be executed offline and which cannot.

Applications of this type will be called Partially Connected or Standalone Applications. They provide access to local data and the ability to execute complex logic on the same

device.

Here, note that the structure of the centralized database on the server that is managed by the mobile application is mirrored on the device. That is to say, a SQLite local database will be created on the device, with the same tables.
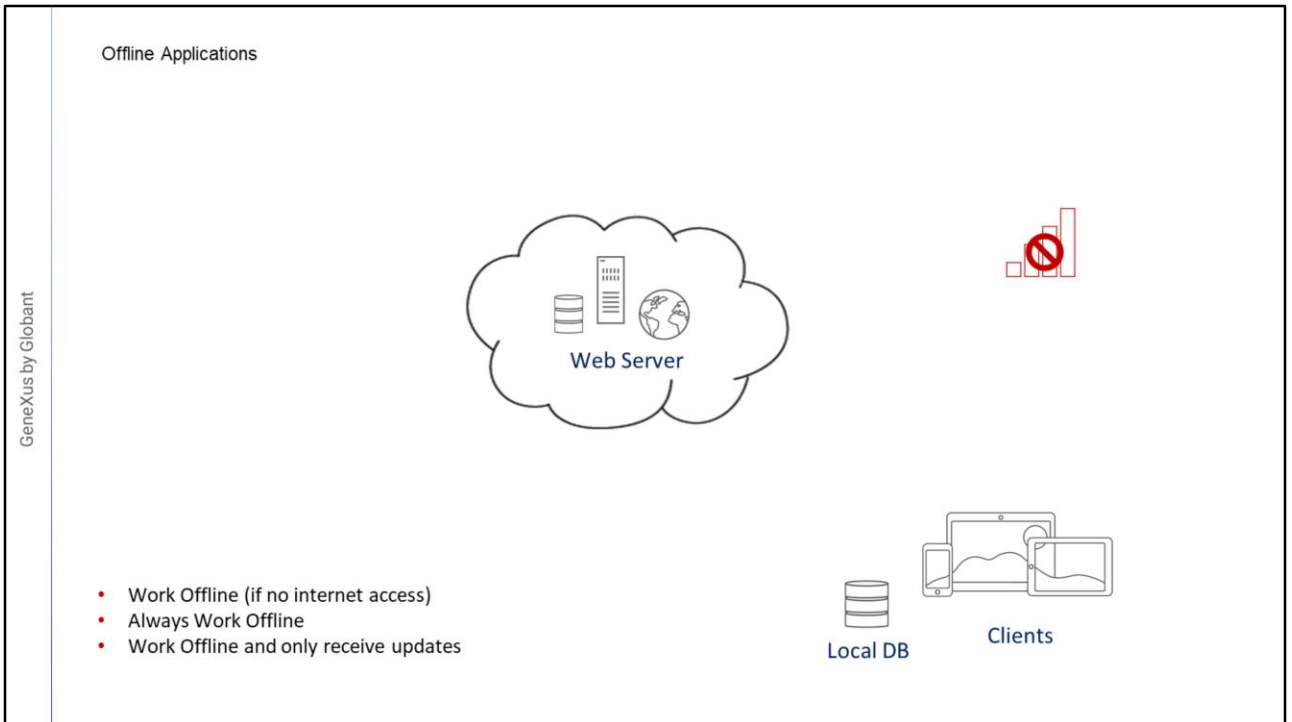
Now, let's talk about applications that can work completely disconnected from the Internet.

In the first case, when the connection is re-established, synchronization with the central database takes place.

In the second case, we may not be interested in running the app's synchronization because we want it to be completely independent from the data on the server. Everything would be managed from the device, as the app installed on it wouldn't communicate with the server.

The third case is when we want the device to be able to receive the changes made on the centralized database, but never to send its own changes, which will be saved in its local database.

These applications that can work without an Internet connection are called offline applications.
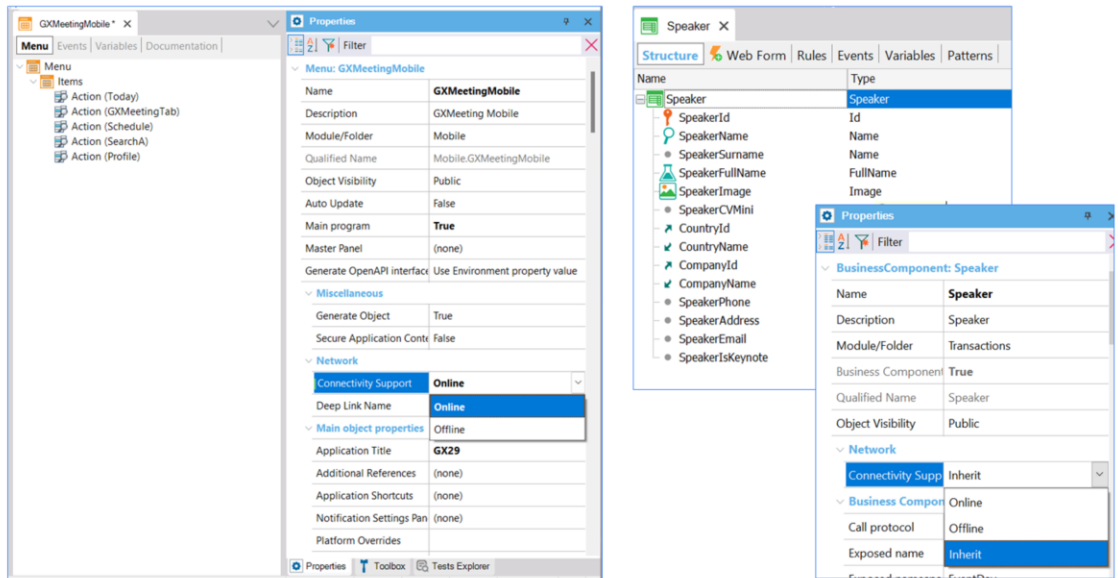
Connectivity Support
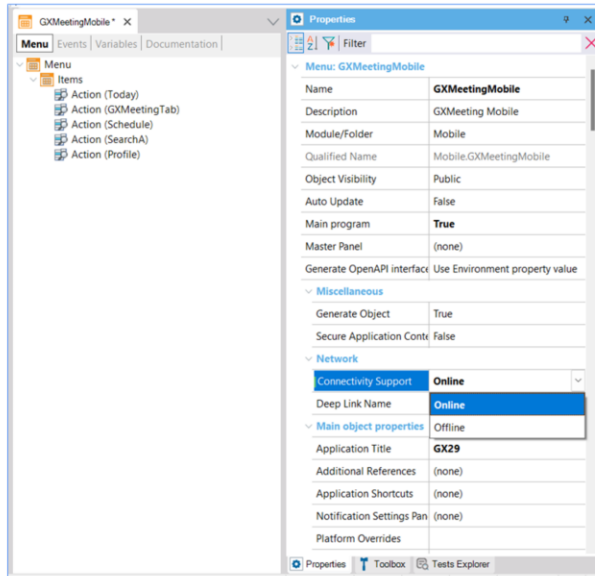
How do we build an offline application using GeneXus?

If we want an application to run offline, we must set the Connectivity Support property of its main object to Offline. This value makes it possible to generate the application offline. Also, the necessary native code will be generated for the application to run without invoking the REST services of the web server.

Once a Main object's Connectivity Support property is set to Offline, when the application is installed on the device, the local database is created and the necessary tables are created there.

Connectivity Support

The Connectivity Support property is also at the level of objects not set as main.
Its value can be Online, Offline, or Inherit.

All the tables that have the Connectivity Support property set to Inherit, used in SD objects
invoked from the Main object (directly or indirectly), will be created in the local
database of the device. If they are set to be Online, the table will not be created in the
local database because the server table will be accessed through REST services.

The Inherit value is displayed only in objects not set as main.

In addition, the Connectivity Support property set to Offline can be assigned to an object.
In this case, the tables will be created in the local database and the object will work
offline, even if the main object works online.