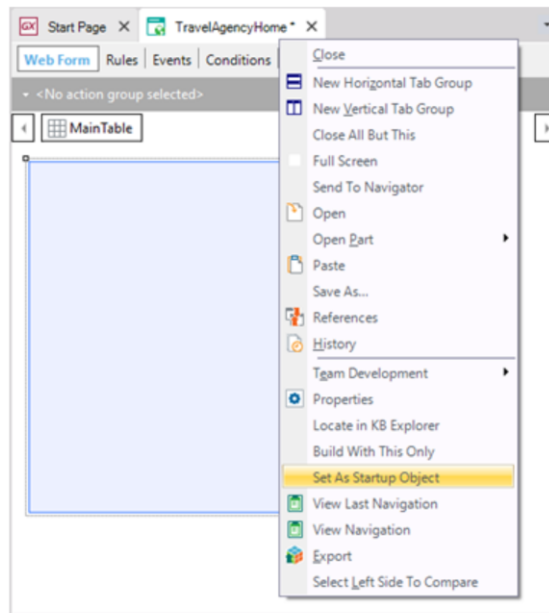


**BUILD Process**

*GeneXus 16*

### Startup object

- Object that we want to run every time we press F5.
- Example: web panel that implements the Travel Agency Homepage.
- Example: SD Panel that implements the entry point of the mobile application.



At some point during the development of our web application, we will implement the object that will be its homepage. From then on, we will no longer need the Developer Menu to prototype.

The Startup object is the Knowledge Base object that we want to have executed after pressing F5, instead of the Developer Menu.

There can only be one startup object. If our app implements a web part and another native part for smart devices, most likely we will not want to test both at the same time. We select the web object as startup object to test this part, and then select the SD object as startup to test the mobile application.

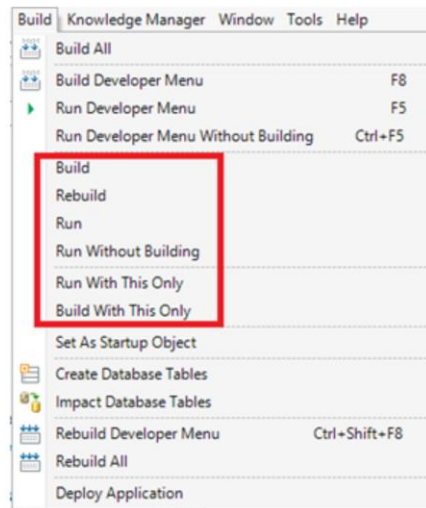
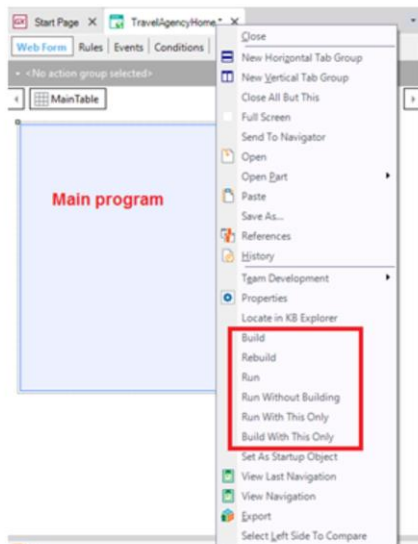
But... what options do we have to test the application?

## Build Process

- It performs the tasks required to leave everything ready for the next time the application is run (all or part of it).
- Actions involved:
  - Checking if changes are needed in the database, and if so, reorganizing it.
  - Specifying, generating and compiling all the objects/programs required.
- It can be applied to:
  - All the objects in the KB
  - Developer Menu
  - A Main object (for example, the startup object)
  - An option can also be applied to a non-main object.

The need to specify, generate and compile an object can be due to the fact that it has been changed directly or that it depends on another object that has been changed. For example, if a domain is changed from Numeric(4.0) to Numeric(8.0), all the objects/programs affected by this change will have to be rebuilt. If we have attributes based on that domain, it will be necessary to specify, etc., all the objects that contain those attributes. Another example: if a procedure's Parm rule is changed, all the objects that invoked it must be rebuilt; however, this doesn't apply if only its layout has been changed.

## Build process: Main object



Suppose that we're editing an object that has the Main Program property set to True. Right-clicking on the object tab displays the highlighted options. Another way to do so is to open the menu's Build option, making sure that we're positioned on the desired main object.

## Build process: Main object

**Build/Rebuild/Run** selected (**main**)

Reorganize if needed

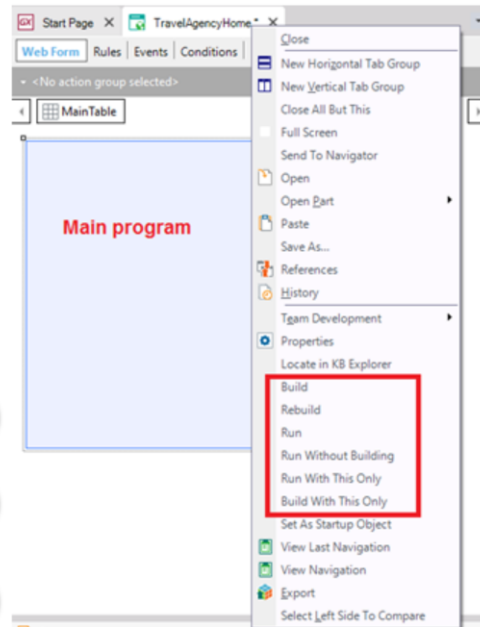
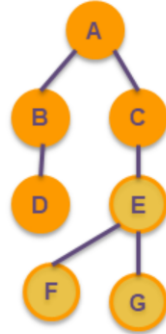
Specifies **only changed** (Build) **all** (Rebuild) objects

Generates

**Compiles** selected **main**

Deploy

Execute Populate Data if needed

Execute selected **main** (Run)Is the call  
tree "cut" in  
main objects?

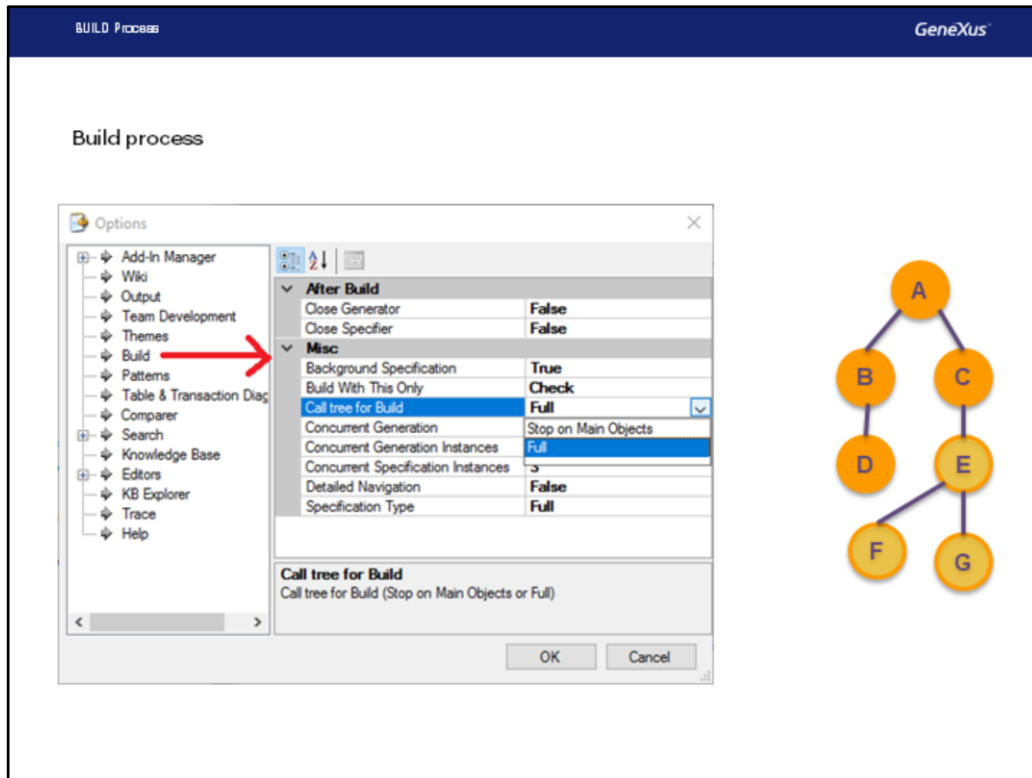
If the object has not been saved, first of all it is saved.

Depending on the option selected (Build or Rebuild), the objects specified will be only those that have been changed or all the objects associated with that main object. That is to say, the object may have not been changed, but it may be affected by a change made to another object, as we explained earlier.

Note: some objects may have been changed or may be affected by the change made to another object, but they are NOT specified. Why? This will depend on the value of the **Call tree for Build** property (see the following page).

Here you will find more information:

<http://wiki.genexus.com/commwiki/servlet/wiki?5692,Build%2FRebuild%2FRun>,



If we open the Tools/Options/Build option, we see that the property **Call tree for Build** has two possible values: "Stop on Main Objects" and "Full". What's the difference between them?

For the option "Stop on Main Objects", selecting Build A, where A is a Main object, will specify all the objects in the invocation tree of A that have been affected by some change but which are **not main objects**. If E is a main object, the specification will stop there. That is to say, objects F and G will not be specified. Otherwise, if the option selected is "Full", everything will be specified.

This is valid to all the Build/Rebuild/Run options.

BUILD Process / for a Main object
GeneXus®

**Build process: Main object**

**Build/Rebuild/Run selected (main)**

Reorganize if needed  
 Specifies only changed (build)  
all (rebuild) objects  
 Generates  
**Compiles** selected **main**  
**Deploy**  
 Execute Populate Data if needed  
 Execute selected **main** (Run)

**Run Without Building**  
 Execute selected **main** (Run)

**Build/Run With This Only (main OR NOT main)**

The screenshot shows the GeneXus IDE interface. On the left, a window titled 'Main program' is visible. On the right, a context menu is open, displaying various actions. The options 'Build', 'Rebuild', 'Run', 'Run Without Building', 'Run With This Only', and 'Build With This Only' are grouped together and highlighted with a red rectangular box. The 'Run With This Only' option is currently selected.

Once the objects have been specified, they are generated in the environment/generator language, the main object is compiled and the Deployment is made. If it has been affected by a change in the Data Provider used to Populate a transaction with data, it is run.

If the option selected was Build or Rebuild, the process ends here. Otherwise, if the option selected was Run, all of the above is performed, and the main object selected is run.

**Run Without Building.** It saves all the objects that haven't been saved and runs the main object without doing anything else, regardless if the object or some other object in its invocation tree has been changed.

<http://wiki.genexus.com/commwiki/servlet/wiki?20689,Run+Without+Build>,

**Build /Run With This Only.** The objective of this option is to specify and generate only the object selected, regardless if it is a main object or not. If it isn't a main object, the only option enabled will be Build, not Run.

Build process: any object

### Build/Run With This Only (main OR NOT main)

Reorganize if needed

Specifies only **"this"** object

Generates only **"this"** object

**Compiles** **"this"** if it is **main**

**Startup object if "this" is not main  
Developer Menu otherwise**

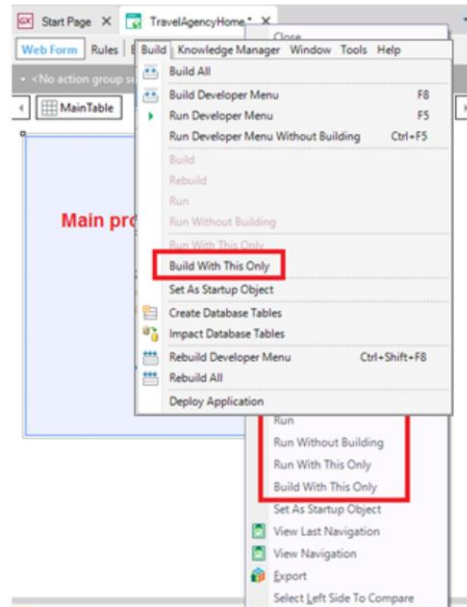
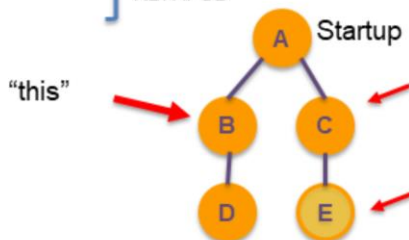
**Deploy**

Execute Populate Data if needed

Execute (Run) Startup (Run)

DevMenu if web

KBN if SD



**Build /Run With This Only.** The objective of this option is to specify and generate only the object selected, regardless if it is a main object or not. If it isn't a main object, the only option enabled will be Build, not Run. The Startup Object is executed later (or the Developer Menu if there isn't a startup object), incorporating only the changes in **"this"** object which was indicated.

Suppose that A is the Startup Object and that changes have been made to objects B, C and E. But we only want to test the changes made to B, fast. We don't want to specify and generate the programs of the other objects that have been changed. So, selecting **"Build with this only"** over object B will only specify and generate this object. Selecting then **"Run without building"** over object A will compile startup (A) that will only see changes in B. The deployment is made if B is a transaction with a DP used to Populate Data and this DP has been changed; it is run to populate the transaction again, and the Startup is also run for us to test object B. We need to be careful when using this option so as not to get confused and test other objects such as C, because these changes will not have been incorporated.

In general: only the selected object is specified and generated, regardless of the changes made to other objects. If the object is a main object, it is also compiled and run. Otherwise, it compiles and run the startup object if it exists; if it doesn't exist: the Developer Menu if **"This"** is a web object, or the KBN if **"This"** is a Smart Device object.

<http://wiki.genexus.com/commwiki/servlet/wiki?5693,Build%2FRun+With+This+Only>



BUILD Process
GeneXus®

### Build process

#### Build/Rebuild All

Reorganizes if it is necessary  
 Specifies only changed (build) objects  
                   all (rebuild)

Generates  
**Compiles ALL main** objects (Web + SD) + DevMenu  
 Deploy  
 Execute Populate Data if needed

#### Build/Rebuild/Run Developer Menu (not startup object)

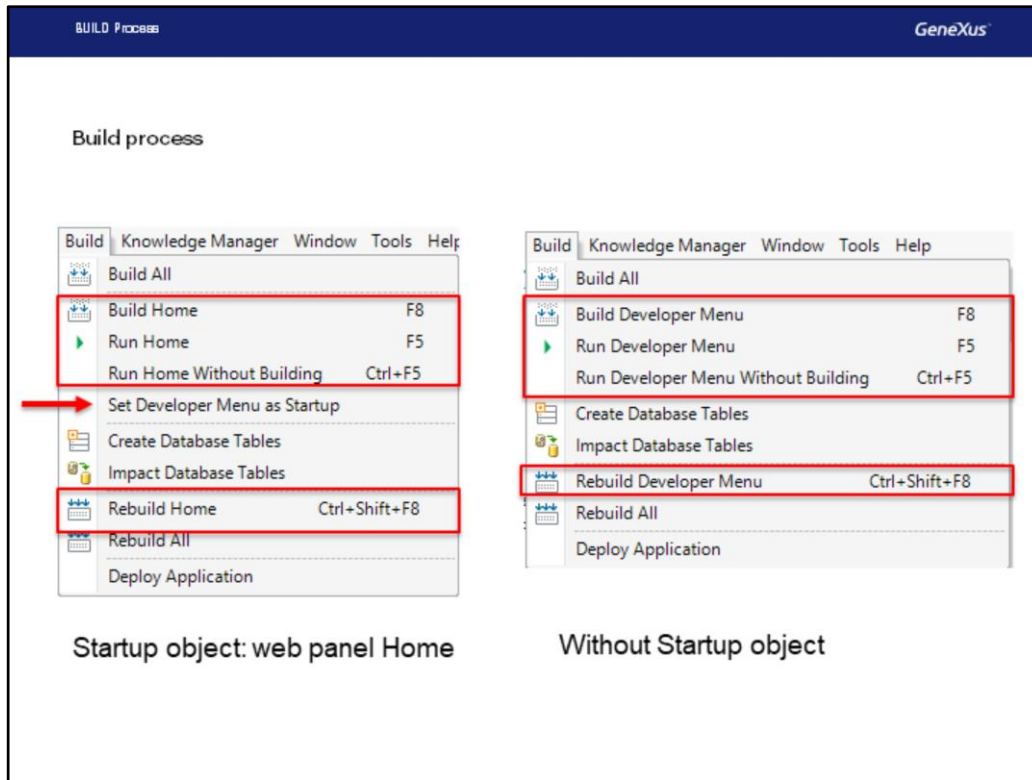
Reorganizes if it is necessary  
 Specifies only changed (build) objects  
                   all (rebuild)

Generates  
**Compiles** DevMenu – **Not compiles** SD mains!  
 Deploy  
 Execute Populate Data if needed  
 Run DevMenu + Run KBN

Here we see the Build menu option, when we're not editing any object in particular. The options Build All and Rebuild All are used when we don't know the exact impact of the changes and we need to have everything updated. The Build All option does everything necessary according to the changes made, while the Rebuild All option builds everything again; therefore, depending on the size of the KB it may take considerable time.

<http://wiki.genexus.com/commwiki/servlet/wiki?5691,Build+All%2FRebuild+All>,

Note that the options that have shortcuts (F8, F5, Ctrl+F5, Ctrl+Shift+F8): are those related with the Developer Menu.  
 What happens if a startup object has been configured?



If there is a Startup Object, instead of the options Build Developer Menu, Run Developer Menu, Run Developer Menu Without Building, Rebuild Developer Menu, the same options will be displayed but for the Startup object.



Videos

Documentation

Certifications

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

[wiki.genexus.com](http://wiki.genexus.com)

[training.genexus.com/certifications](http://training.genexus.com/certifications)