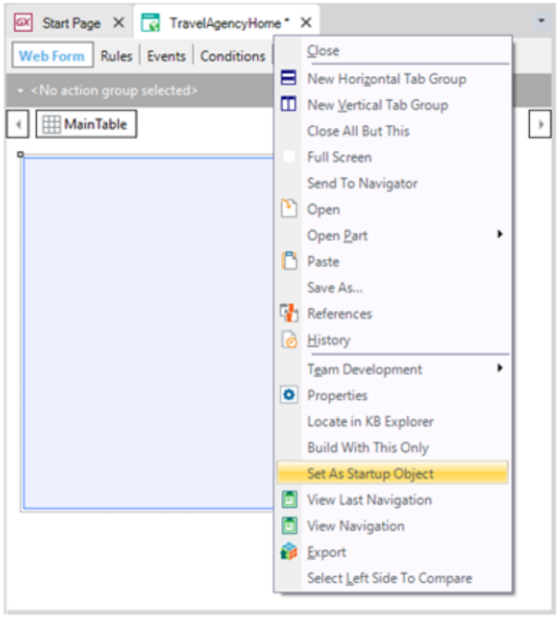
Proceso de BUILD

GeneXus 15

GeneXus

Startup object

- Objeto que queremos sea el que se ejecute cada vez que presionemos F5.
- Ejemplo: web panel que implementa el Home de Travel Agency
- Ejemplo: SD Panel que implementa el punto de entrada de la aplicación móvil



En algún momento del desarrollo de nuestra aplicación Web vamos a implementar el objeto que será el home de nuestra aplicación. A partir de allí, ya no necesitaremos el Developer Menu para prototipar.

El Startup object es el objeto de la Knolwledge Base que queremos sea el que se ejecute cuando presionemos F5, en lugar del Developer Menu.

Sólo puede haber un startup object. Si nuestra app implementa una parte web y otra nativa para SD, seguramente no queramos probar ambas a la vez. Elegimos como startup el objeto web para probar esa parte, y luego elegimos como startup el objeto SD, para probar la aplicación móvil.

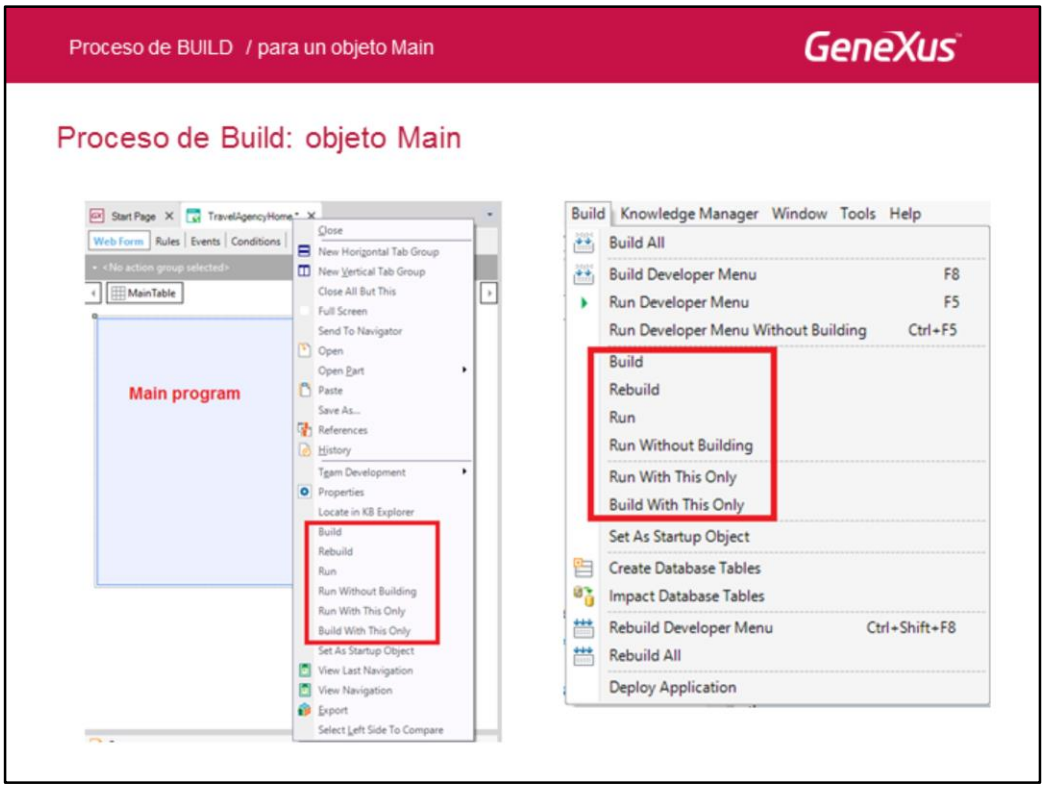
Pero… ¿qué opciones tenemos a la hora de probar la aplicación?

GeneXus

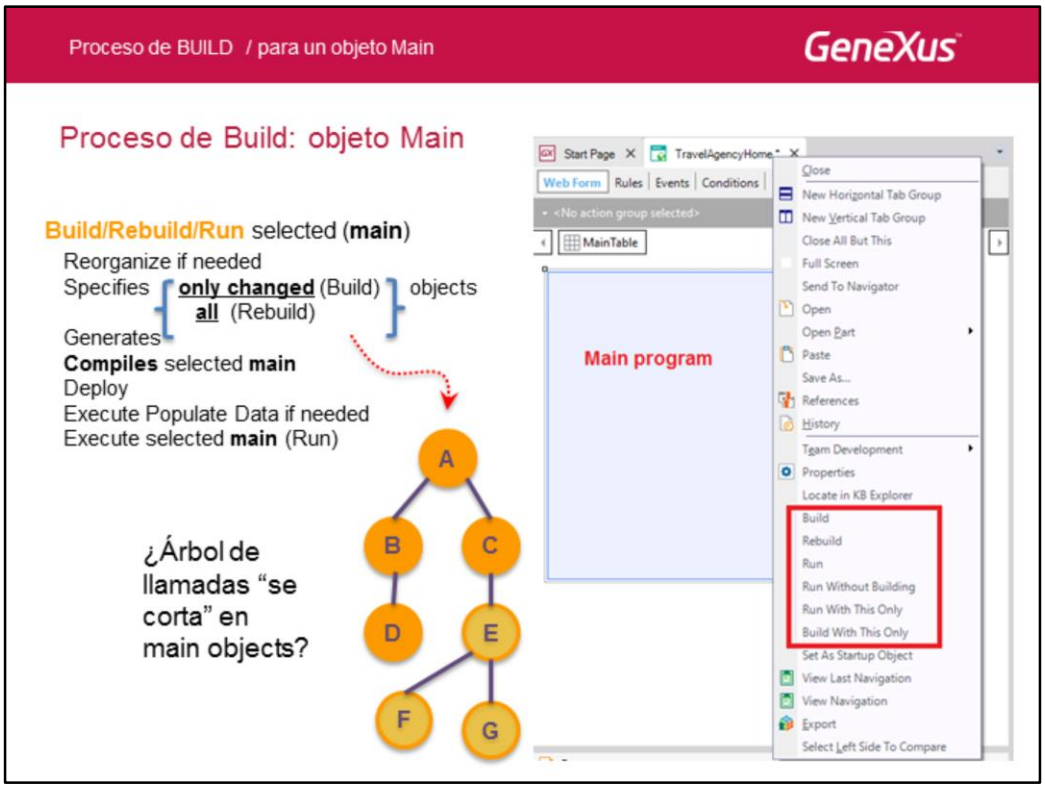
Proceso de Build

- Realiza las tareas requeridas para dejar todo listo para la próxima ejecución de la aplicación (entera, o parte de ella).
- Involucra:
  - Chequear si deben realizarse cambios en la base de datos, y si los hay, reorganizarla.
  - Especificar, generar y compilar todos los objetos/programas que sean necesarios.
- Puede ser aplicado a:
  - Todos los objetos de la KB
  - El Developer Menu
  - Un objeto Main (por ejemplo, el startup)
  - Alguna opción también puede ser aplicada a objeto no main

La necesidad de especificar, generar y compilar un objeto puede deberse a que éste sufrió algún cambio directamente, o a que depende de algún otro objeto que sufrió algún cambio. Por ejemplo, si cambia un dominio de Numeric(4.0) a Numeric(8.0) habrá que volver a construir todos los objetos/programas a los que ese cambio afecte. Por ejemplo, si teníamos atributos basados en ese dominio, habrá que especificar, etc., todos los objetos con esos atributos. Otro ejemplo: si cambia la regla parm de un procedimiento, todos los objetos que lo invocaban deben volverse a construir, pero si solamente cambia su layout, no.



Supongamos que estamos editando un objeto que tiene la propiedad Main Program en True. Si hacemos botón derecho sobre la solapa del objeto, vemos las opciones que aparecen destacadas. Otra forma de hacerlo es ir a la opción Build del menú, estando seguros de estar posicionados sobre el objeto main deseado.



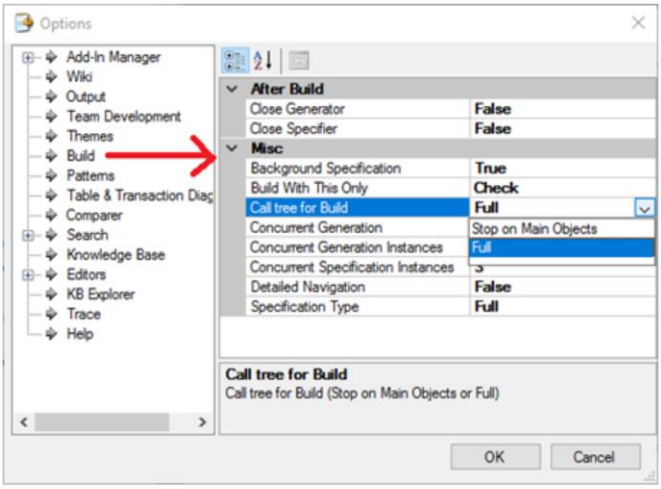
Si el objeto no está grabado, como primer paso se graba.

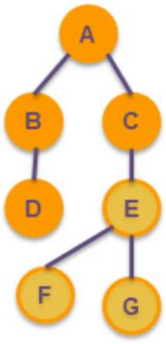
Dependiendo de la opción (si es Build o Rebuild) especifica sólo los objetos que hayan cambiado o todos los objetos que dependan de ese main. El objeto puede no haber cambiado, pero ser afectado por el cambio en otro objeto, como explicamos antes.

Nota: puede que algunos objetos hayan cambiado o ser afectados por el cambio de otro y sin embargo NO se especifiquen. ¿Por qué? Dependerá del valor de la propiedad **Call tree for Build** (ver página siguiente).

(vea aquí por más información: http://wiki.genexus.com/commwiki/servlet/wiki?5692,Build%2FRebuild%2FRun,)

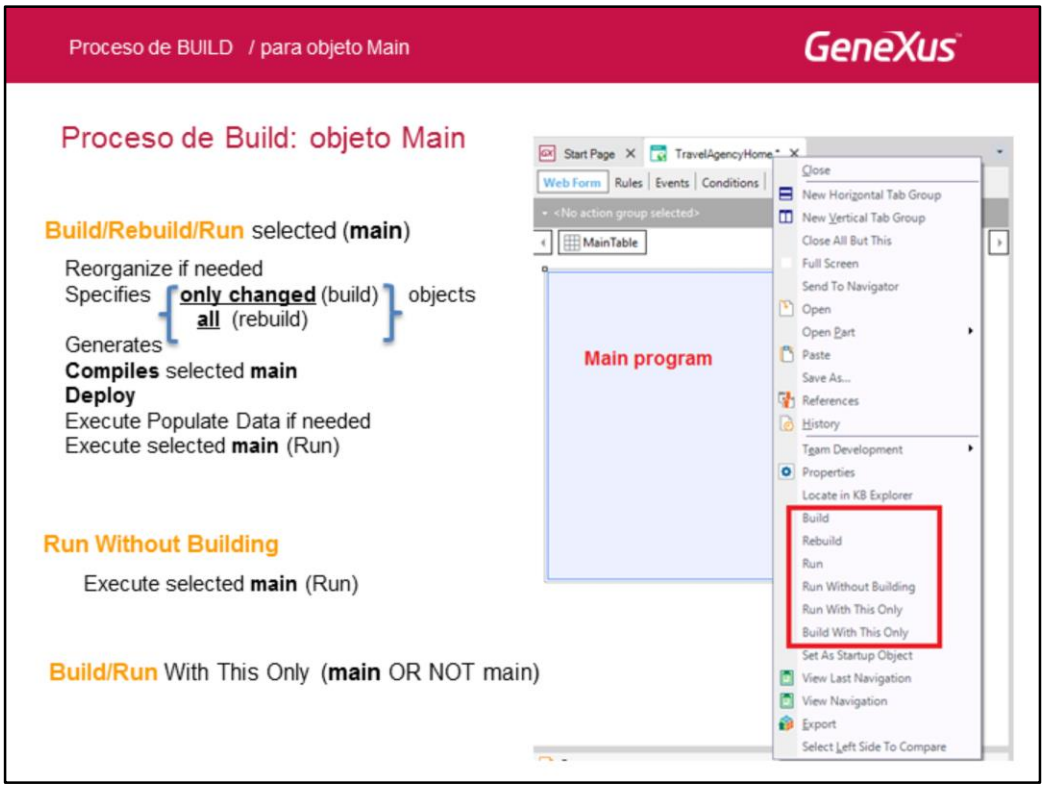
Proceso de Build





Si vamos a Tools/Options/Build vemos que la propiedad **Call tree for Build** tiene dos valores posibles: “Stop on Main Objects” y “Full”. ¿Qué diferencia hay?
Si la opción es “Stop on Main Objects”, cuando se elija hacer Build A, siendo A un objeto Main, se especificarán todos los objetos del árbol de invocaciones de A que se hayan visto afectados por algún cambio pero que **no sean main**. Si E es main, se cortará la especificación allí. Es decir, no se especificarán los objetos F y G. En cambio si la opción es “Full” se especificará todo.

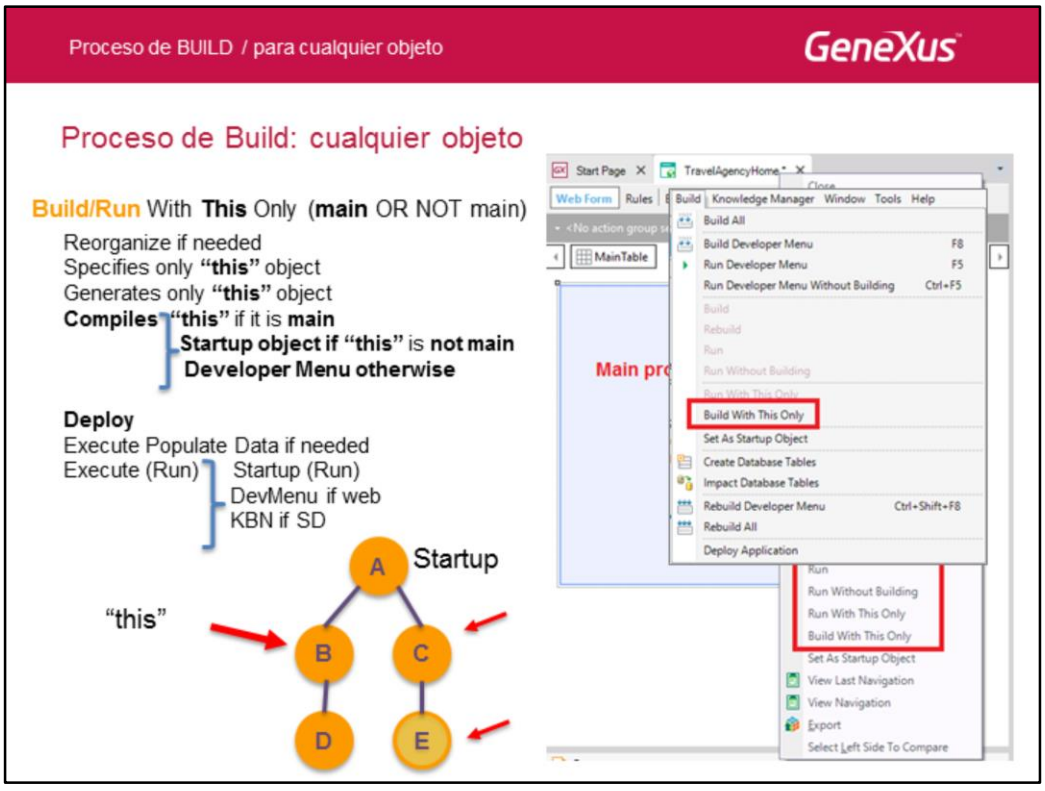
Esto vale para todas las opciones Build/Rebuild/Run.



Luego de especificados los objetos, se generan en el lenguaje del environment/generador y se compila el main y se hace el Deploy. Si se vio afectado por un cambio el Data Provider usado para “Populate data” de una Transacción, se ejecuta.
Si la opción elegida fue Build o Rebuild el proceso termina aquí. Si la opción en cambio fue Run, se hace todo lo anterior y además se ejecuta el main seleccionado.

Run Without Building. Graba todos los objetos que no se hayan salvado y ejecuta el objeto main sin hacer más nada, independientemente de que el objeto o alguno de los de su árbol de invocaciones haya cambiado.
http://wiki.genexus.com/commwiki/servlet/wiki?20689,Run+Without+Build,

Build /Run With This Only. La idea de esta opción es especificar y generar solamente el objeto seleccionado, sea o no main. Si no es main, la única opción habilitada será Build, no Run.

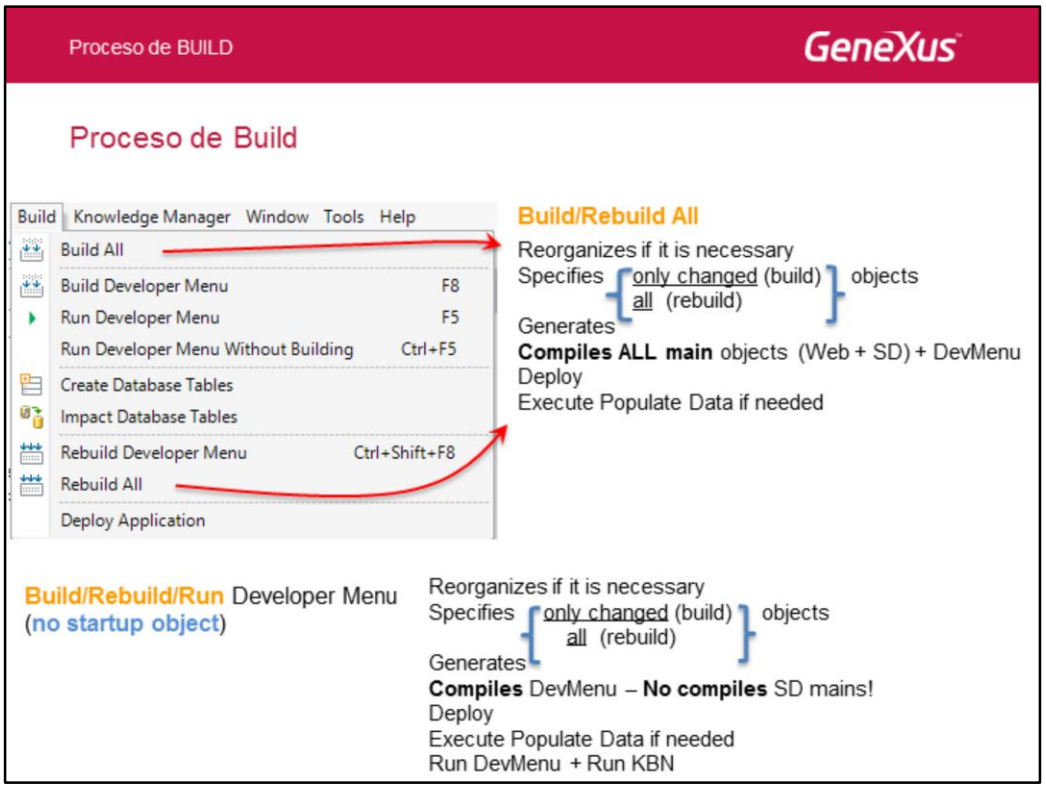


Build /Run With This Only. La idea de esta opción es especificar y generar solamente el objeto seleccionado, sea o no main. Si no es main, la única opción habilitada será Build, no Run. La intención es ejecutar luego el Startup Object (o el Developer Menu si no hay startup object) , únicamente incorporando los cambios en "este" objeto, el indicado. Supongamos que A es el Startup Object y que se hicieron cambios en los objetos B, C y E. Pero queremos probar únicamente los cambios que hicimos a B, de forma rápida. No queremos que se especifiquen y generen los programas de los demás objetos que cambiaron. Entonces haciendo "Run with this only" sobre el objeto B, solamente se especificará y generará ese objeto y se compilará el startup (A) que solamente verá cambios en B. Se hará el deploy, si B era transacción con DP usado para Populate Date y se modificó ese DP, se ejecuta para volver a poblar de datos la transacción y se ejecutará el Startup, para que probemos el objeto B. Hay que tener cuidado, utilizando esta opción, de no confundirnos probando otros objetos como el C, pues sus cambios no estarán incorporados.

En general: solamente especifica y genera el objeto seleccionado, independientemente de los cambios que hubiera en otros objetos. Si el objeto es main, además lo compila y ejecuta. En caso contrario, compila y ejecuta el startup object si existe, y si no existe:

- el Developer Menu si "This" es un objeto web, o
- el KBN si "This" es un objeto Smart Device.

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

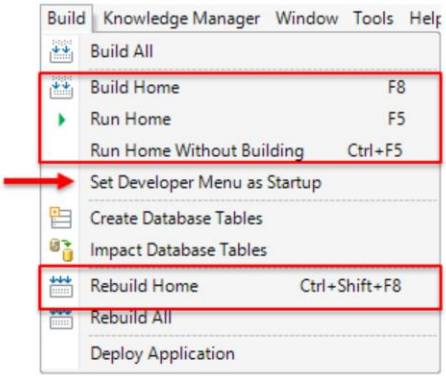


Aquí vemos la opción de Build del menú, cuando no estamos editando ningún objeto en particular.
Las opciones Build All y Rebuild all se utilizan cuando no sabemos exactamente el impacto de los cambios y necesitamos que nos quede todo actualizado. La opción Build All hará lo que se necesite en base a los cambios que se hayan efectuado, mientras que la opción Rebuild All vuelve a construir todo de vuelta, y por lo tanto dependiendo del tamaño de la KB puede insumir un tiempo considerable.

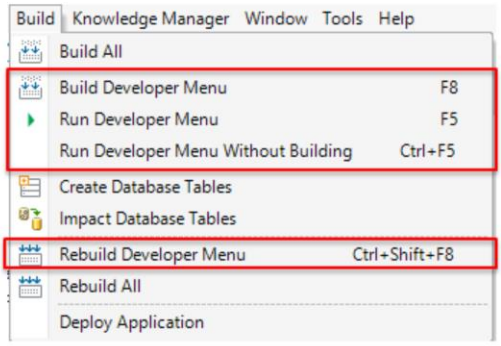
http://wiki.genexus.com/commwiki/servlet/wiki?5691,Build+All%2FRebuild+All,

Obsérvense las opciones que tienen shortcuts (F8, F5, Ctrl-F5, Ctrl+Shift+F8): son las relacionadas al Developer Menu.
¿Pero qué pasa si tenemos startup object configurado?

Proceso de Build



Startup object: web panel Home



Sin Startup object

Si existe Startup Object, en lugar de las opciones Build Developer Menu, Run Developer Menu, Run Developer Menu Without Building, Rebuild Developer Menu, aparecerán las mismas opciones, pera para el objeto Startup.

GeneXus™

Videos	training.genexus.com
Documentación	wiki.genexus.com
Certificaciones	training.genexus.com/certificaciones