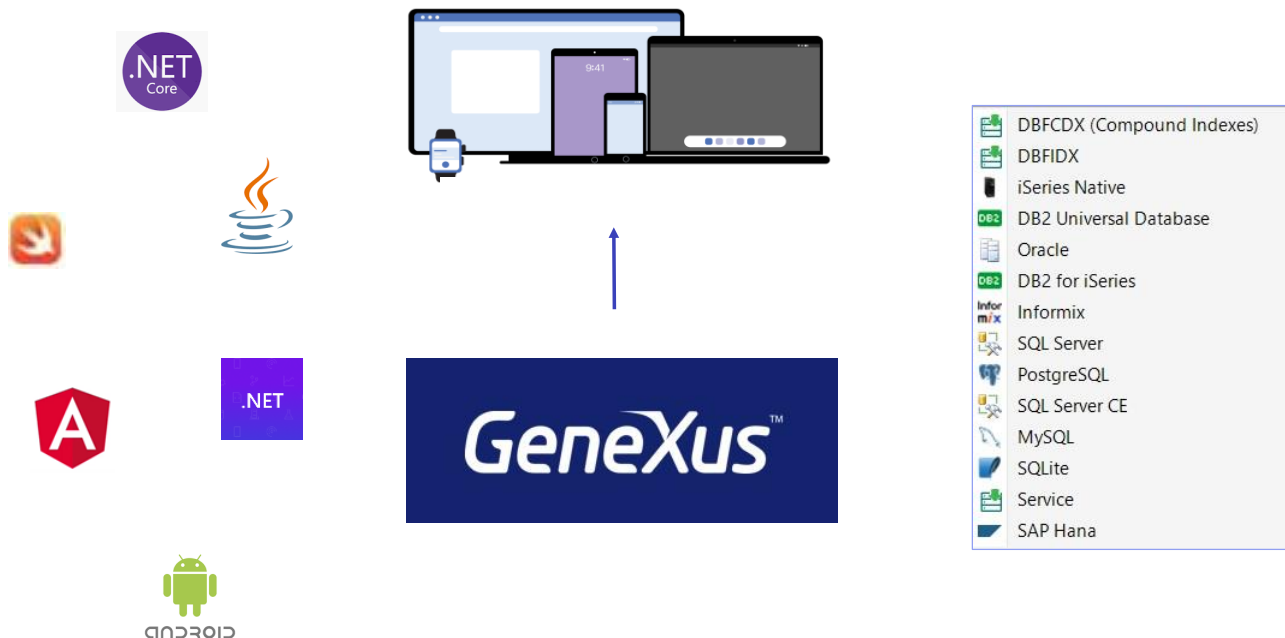


# Introducción a Environments

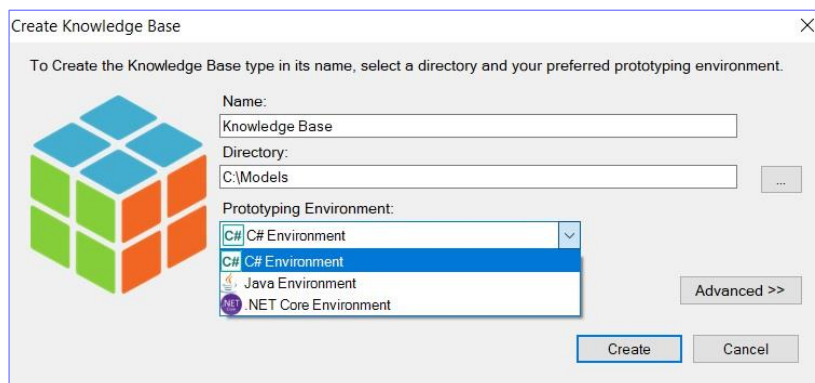
*GeneXus™*



Como ya sabemos, uno de los grandes beneficios de GeneXus es que permite generar aplicaciones para diferentes plataformas, con interface web o para dispositivos móviles, generando código en diferentes lenguajes de programación o almacenando los datos de la aplicación en diferentes bases de datos.

Toda esta información se define en un Environment, un ambiente de ejecución.

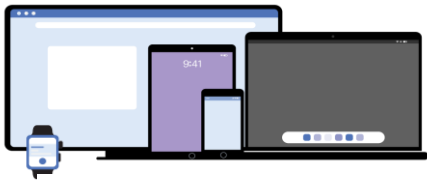
## GeneXus Full version - Environments



Recordemos que GeneXus cuenta con una versión full y una versión de prueba que tiene ciertas limitaciones, entre las cuales se encuentran los lenguajes de programación y base de datos disponible.

Al momento de creación de la base de conocimiento se crea un ambiente por defecto. Si contamos con una versión full de GeneXus, entonces el cuadro de diálogo solicita al usuario que seleccione el lenguaje de programación que será utilizado para el Back end de la aplicación. El resto de la información deberá configurarse a través de sus propiedades.

## Front end and Back end



FRONT-END

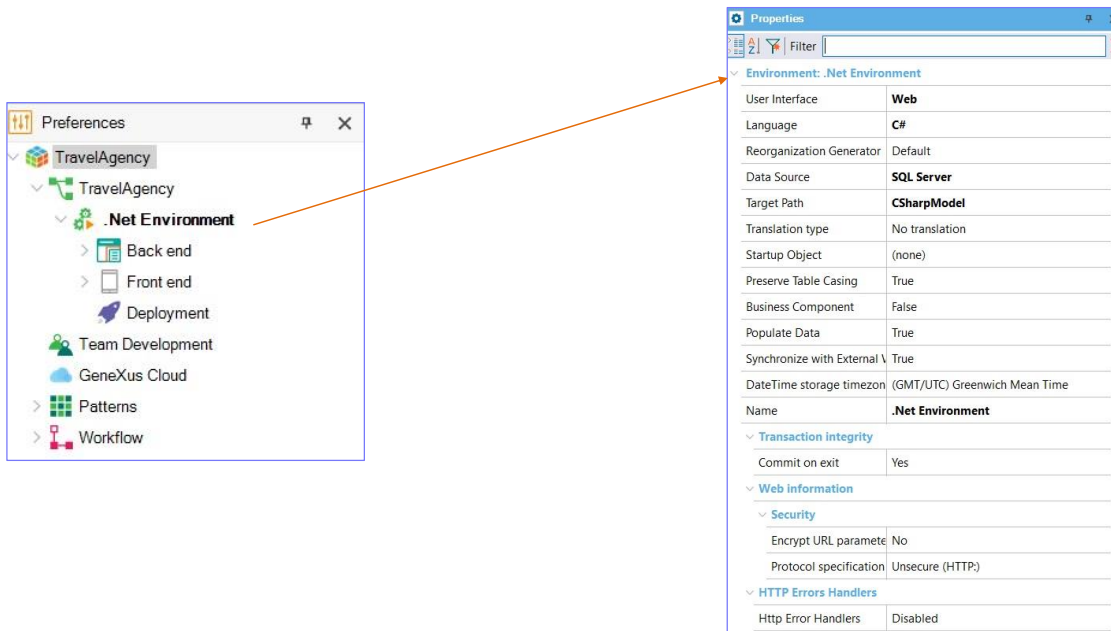


BACK-END



Recordemos que el Front end es la parte de la aplicación que interactúa con los usuarios, mientras que el Back end de una aplicación se encarga de toda su lógica y programas, así como también de la conexión con la base de datos, servidores, etc.

## GeneXus Full version - Environments

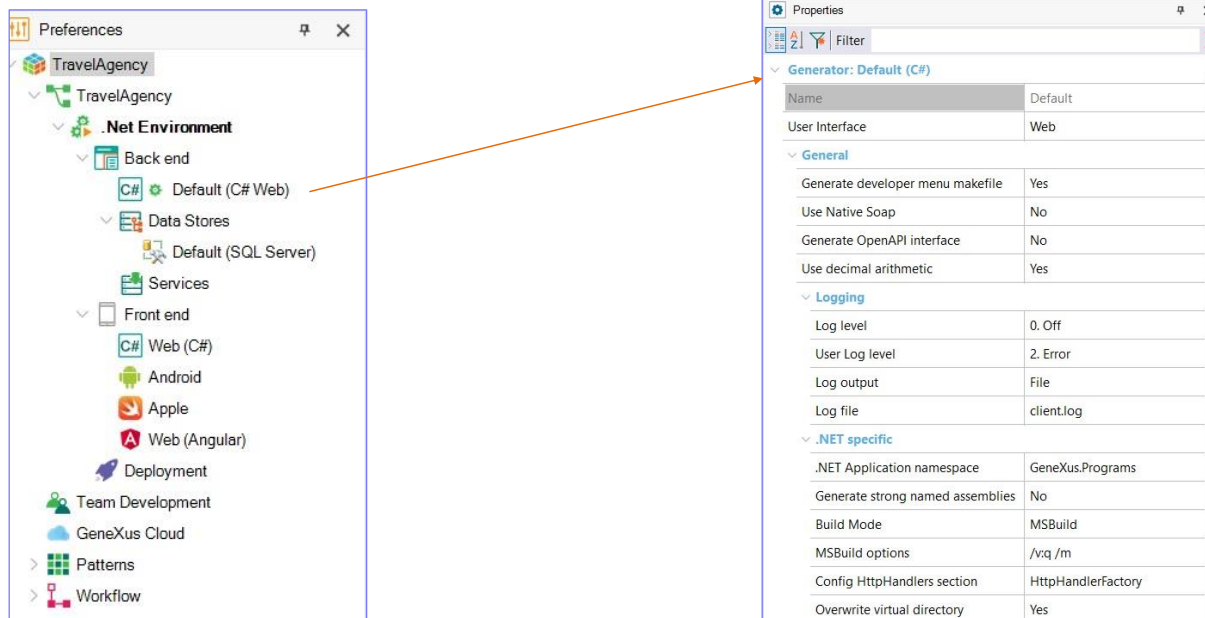


Ya en el IDE de una versión full de GeneXus, toda la información del Environment se encuentra en la ventana de Preferences. Si observamos, el Environment tiene definidas varias propiedades que determinan la plataforma de ejecución y sus características.

Un Environment está compuesto por los siguientes nodos:

- Back end
- Front end
- Y Deployment

## GeneXus Full version – Environments

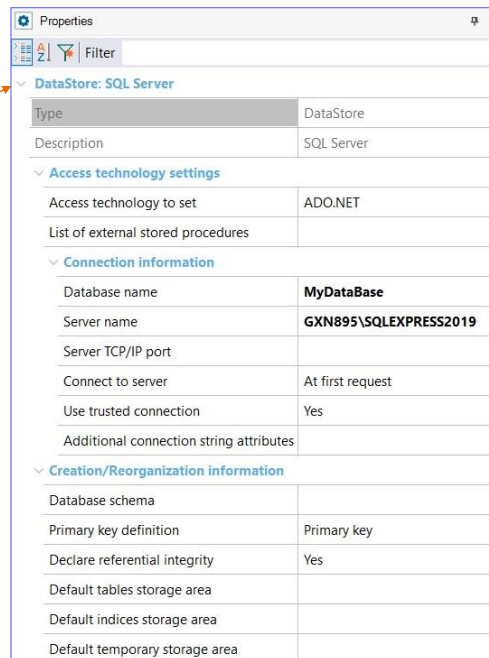
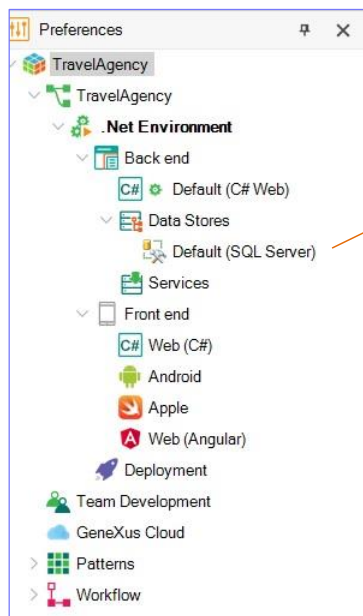


En el nodo Back end se definen los lenguajes de programación que se utilizarán para su generación. Podría ser C#, Java, .Net Core. Vemos ya el generador por defecto, que fue el indicado al momento de creación de la base de conocimiento.

Es posible definir más generadores bajo este nodo, simplemente haciendo click derecho y eligiendo New generator.

En las propiedades del generador se definen característica de los objetos generados, del servidor o del lenguaje de programación a utilizar.

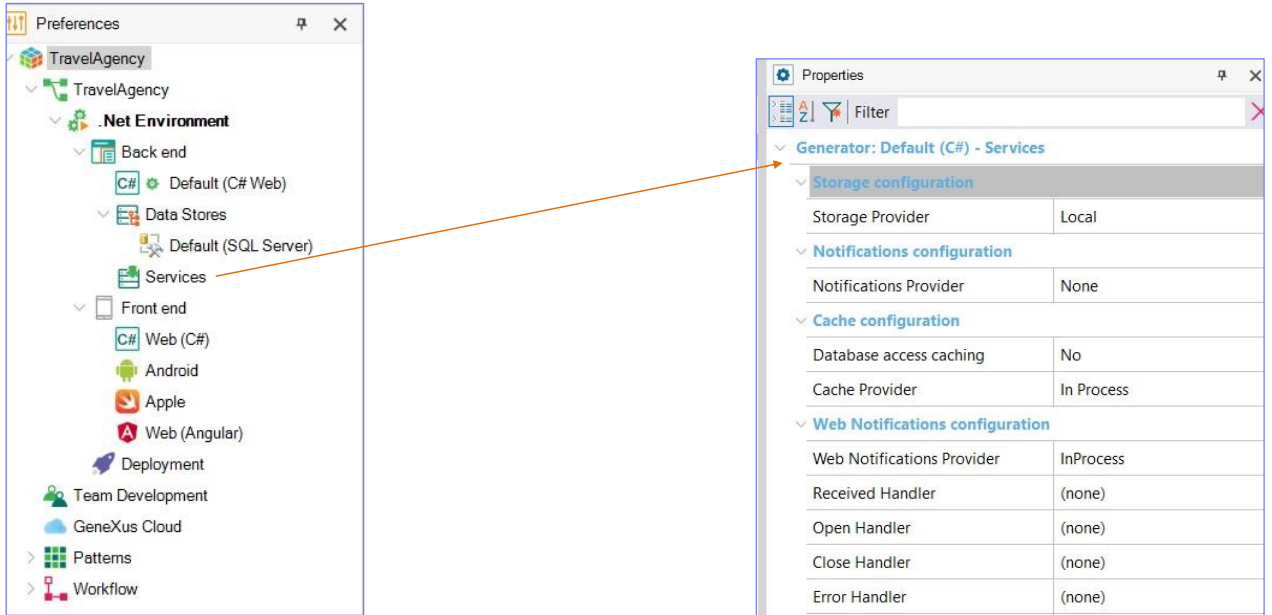
## GeneXus Full version – Environments



Pasemos ahora al nodo Data Stores. Aquí se define el tipo de base de datos donde guardaremos los datos de nuestra aplicación, y también es posible definir otras bases de datos a las cuales acceder para consultar información.

Si observamos las propiedades del Data Store vemos que podemos definir los datos de conexión a la base de datos, como ser la dirección del server, usuario, contraseña, versión de la base de datos, etc.

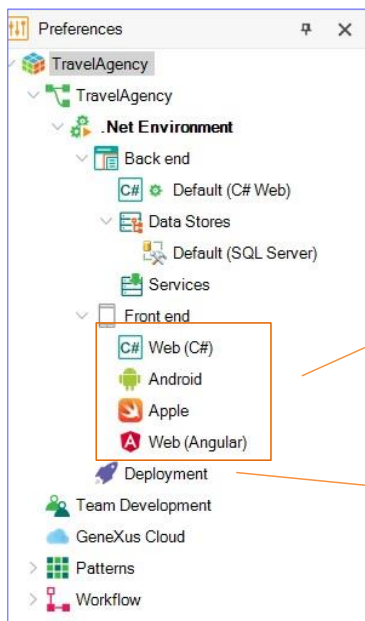
## GeneXus Full version – Environments



En cuanto al nodo Services, en sus propiedades podemos configurar el manejo de servicios, configuraciones de almacenamiento, notificaciones, etc.



## GeneXus Full version – Environments



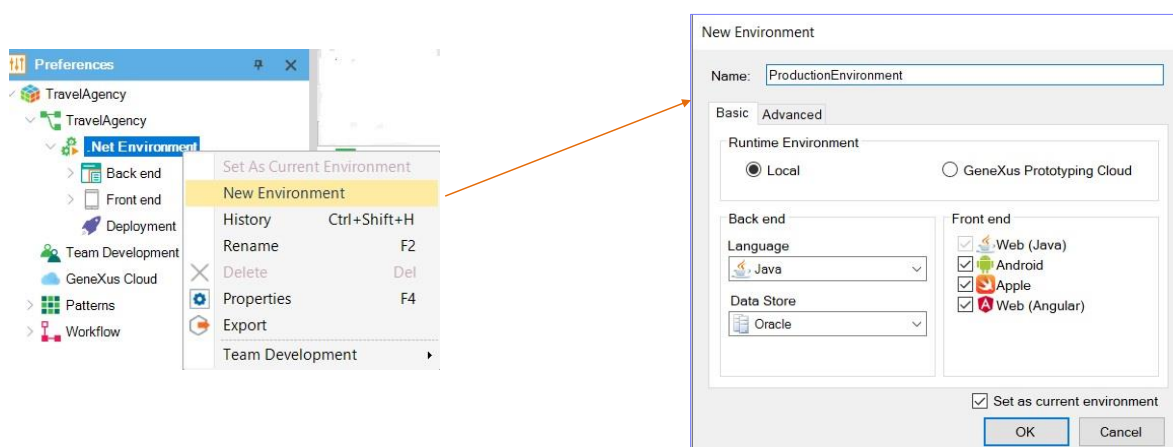
Generadores disponibles para el diseño del Front end.

Unidad de Deployment: indica el conjunto de objetos a integrarse al hacer el deploy.

Pasemos ahora al nodo Front end.

Aquí vemos el conjunto de generadores disponibles para el diseño de este componente de la aplicación, y finalmente el nodo Deployment, donde se pueden definir distintas unidades de deployment. Vale mencionar que una unidad de Deployment indica el conjunto de objetos a integrarse al hacer el deploy.

## GeneXus Full version – How to create a new Environment



GeneXus nos permite definir diferentes ambientes de ejecución para la misma base de conocimiento.

Por ejemplo, es posible definir un ambiente de ejecución para la etapa de desarrollo donde nos conectamos a una base con datos de prueba, y luego otro ambiente de producción donde se define el servidor y base de datos para la aplicación terminada.

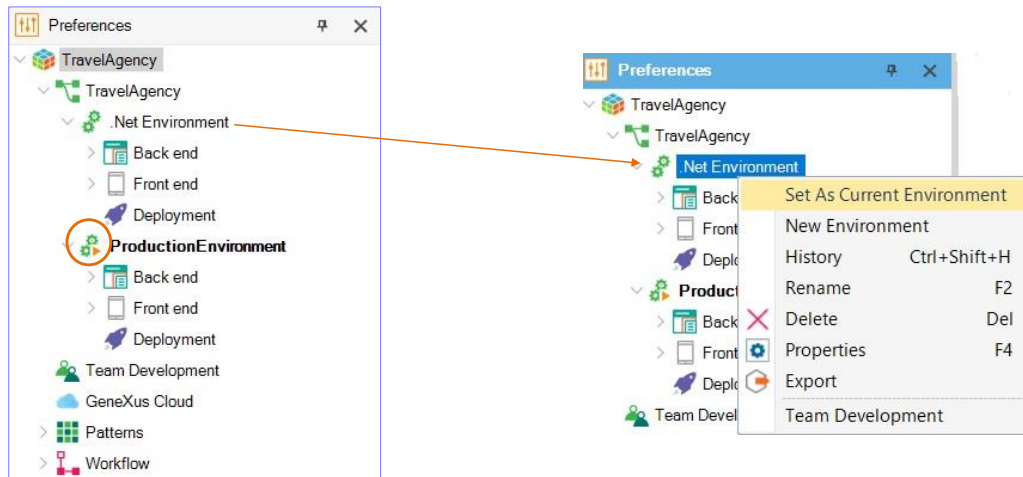
Para eso nos posicionamos sobre el nombre del ambiente ya definido, hacemos click derecho y seleccionamos New Environment.

Debemos indicar:

- El nombre del nuevo ambiente. Vamos a indicar que es un ambiente de producción
- Luego debemos indicar si se va a prototipar en forma local o en la nube de GeneXus.
- Elegir el lenguaje para el Back end, e
- Seleccionar el Data Store que se va a utilizar,
- y los generadores para el Front end.

En caso de seleccionar que se va a prototipar en la nube de GeneXus, entonces queda deshabilitada la posibilidad de elegir el Data Store.

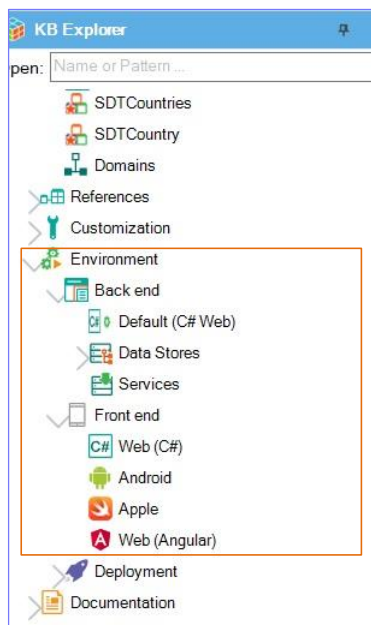
## GeneXus Full version – Current Environment



Una vez definido el nuevo ambiente, observemos que el que tiene el nombre resaltado, y el ícono de play, será el ambiente activo y el que se aplicará al presionar F5.

En caso de querer construir la aplicación en el otro ambiente alcanza con posicionarnos en el nombre de dicho ambiente, hacer click derecho, Set as current environment.

## GeneXus Trial version – Default Environment



Finalmente, en el caso de estar utilizando la versión de prueba de GeneXus, no es posible definir nuevos Environments.

Al momento de creación de la base de conocimiento se define por defecto el único ambiente de ejecución que prototipará en la nube de GeneXus, tendrá C# como generador para el Back end, y C#, Android, Apple y Angular para la generación del Front end.

Si pasamos ahora al IDE de la versión de prueba de GeneXus, vemos que el nodo de Environments se encuentra integrado en la ventana KBExplorer.

Podemos ver las opciones del Back end y también del Front end. Si hacemos click derecho vemos que no tenemos disponible la posibilidad de definir un nuevo ambiente.

# GeneXus™

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

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

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