

Orders Searchs Conditions



19-Orders.Searchs.Conditions_sp



Behavior: orders, searchs and conditions

Cecilia Fernández | GeneXus Training

Orders & Searchs & Conditions

Invocations

Events

Caching

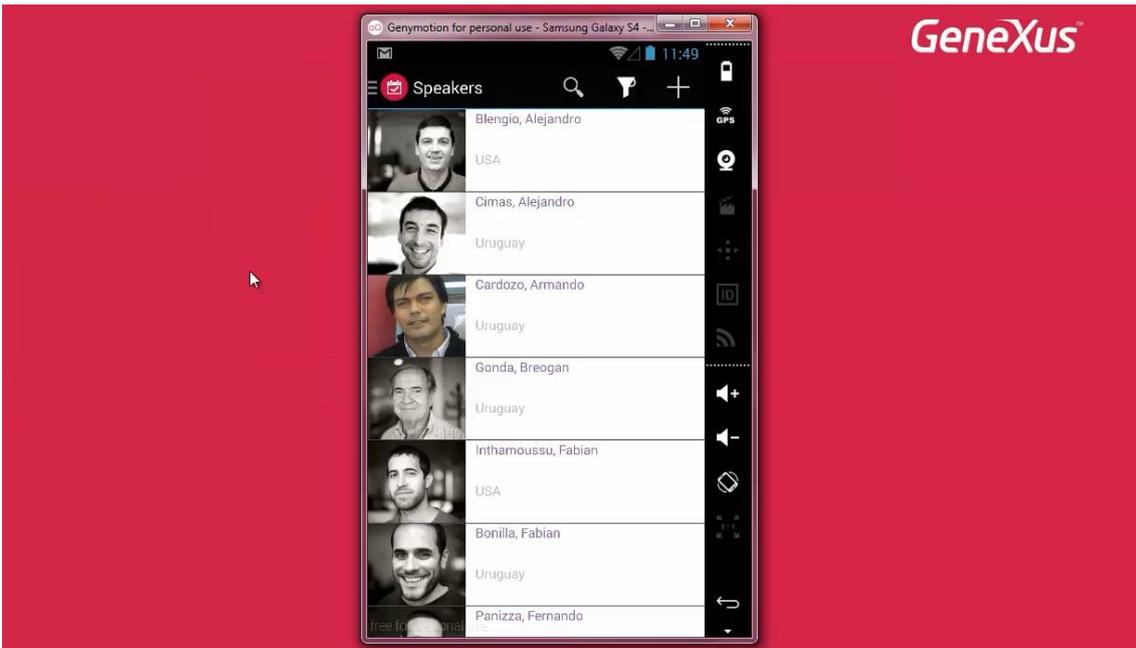
Dentro del comportamiento de las aplicaciones móviles, comencemos por estudiar el modo en el que podemos ordenar la información desplegada en un panel y cómo establecer criterios de búsqueda y filtros para los datos.

Orders & Searchs & Conditions

Invocations

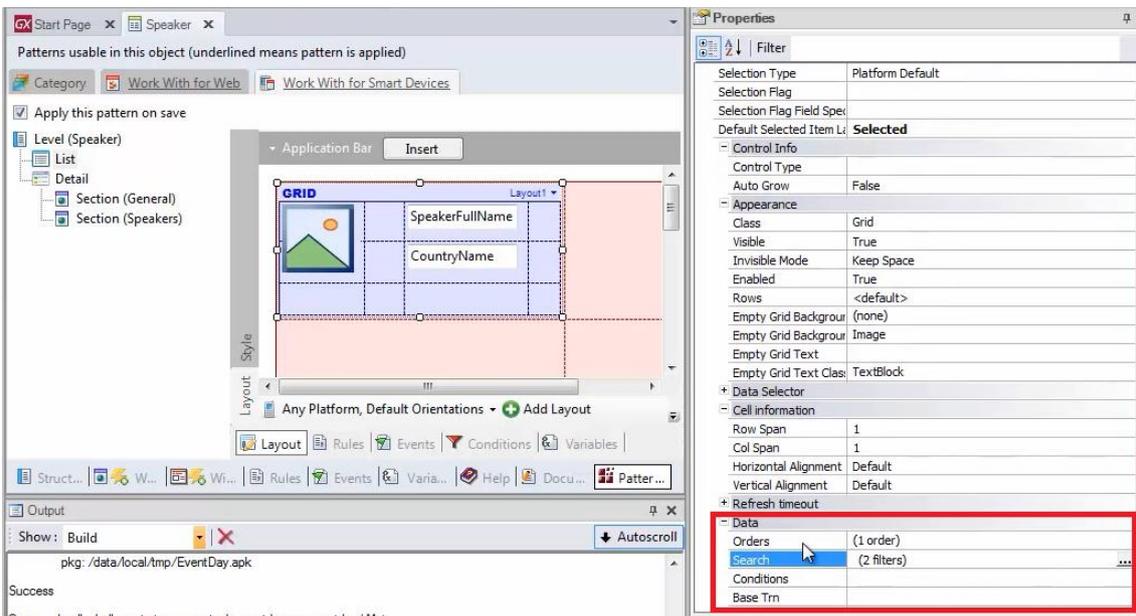
Events

Caching

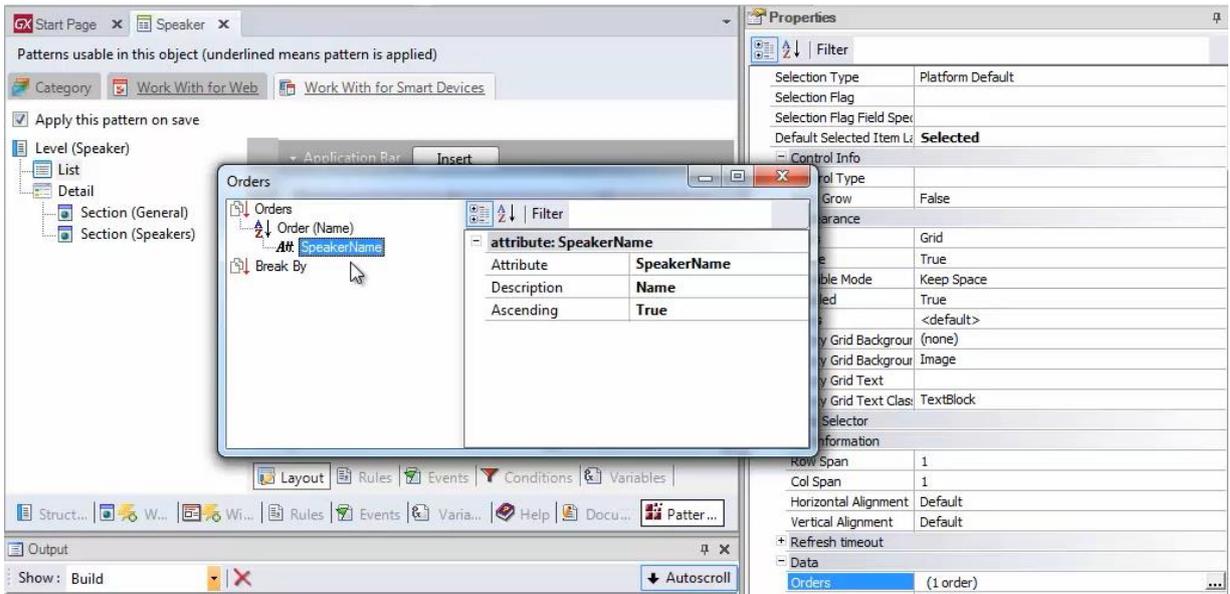


Si observamos el List de speakers, podemos apreciar que los oradores están saliendo ordenados por SpeakerName.

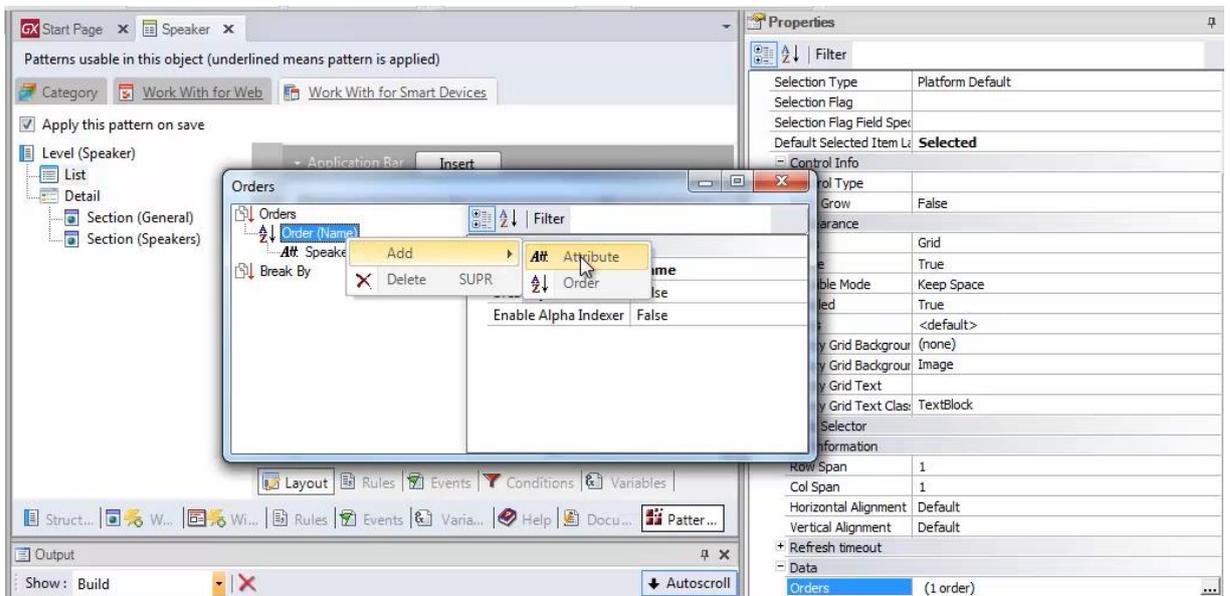
Esto está configurado en GeneXus, a nivel de las propiedades del grid, en las que están bajo el grupo: Data



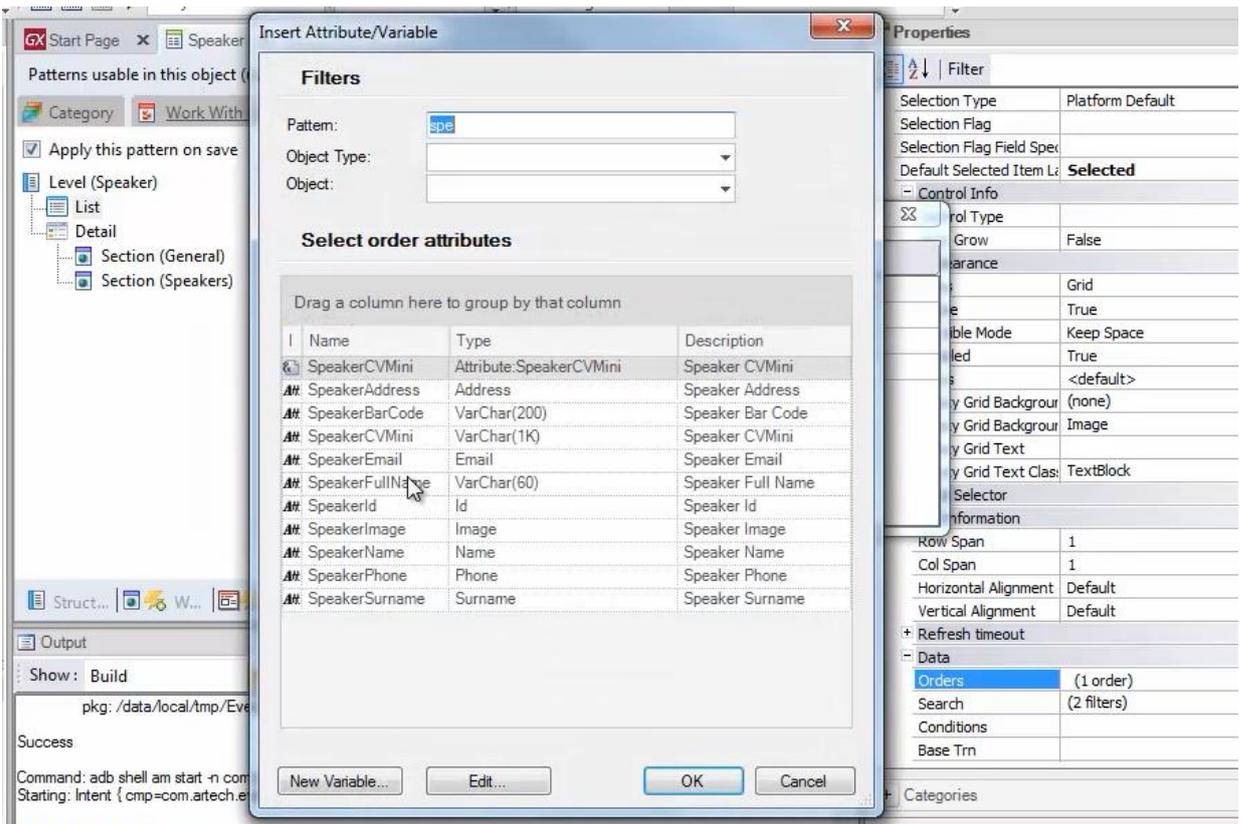
Así, si vamos a la propiedad: Orders, se nos abre esta ventana donde tenemos definido 1 orden, al que se le ha llamado Name, compuesto por el atributo SpeakerName



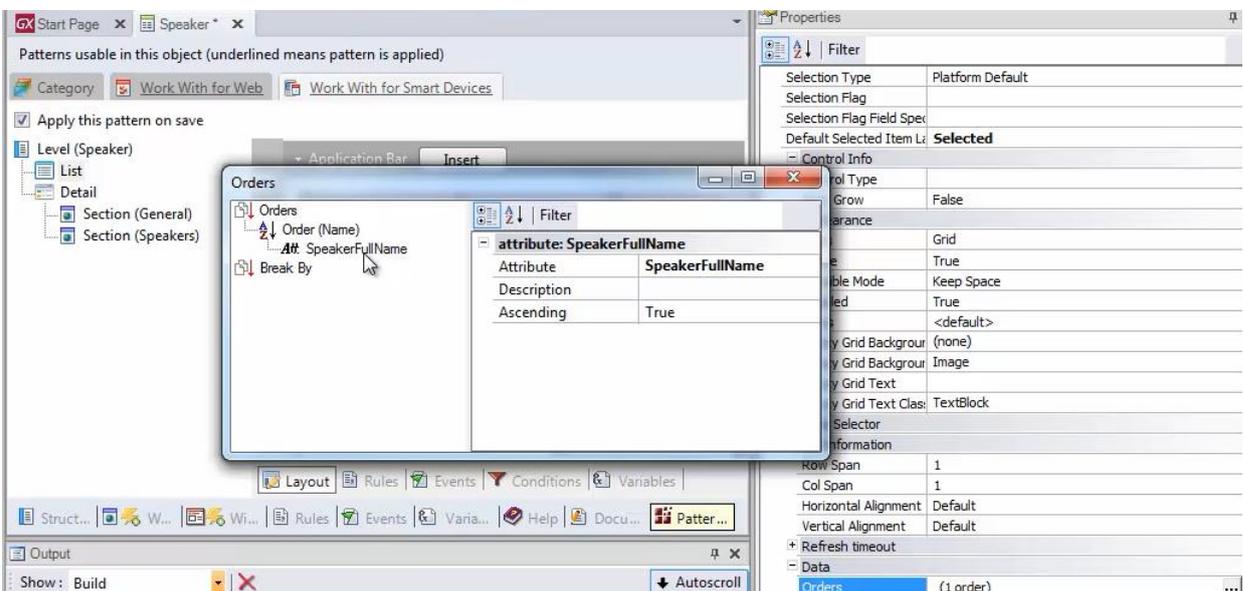
Vamos a modificarlo para que ahora esté compuesto



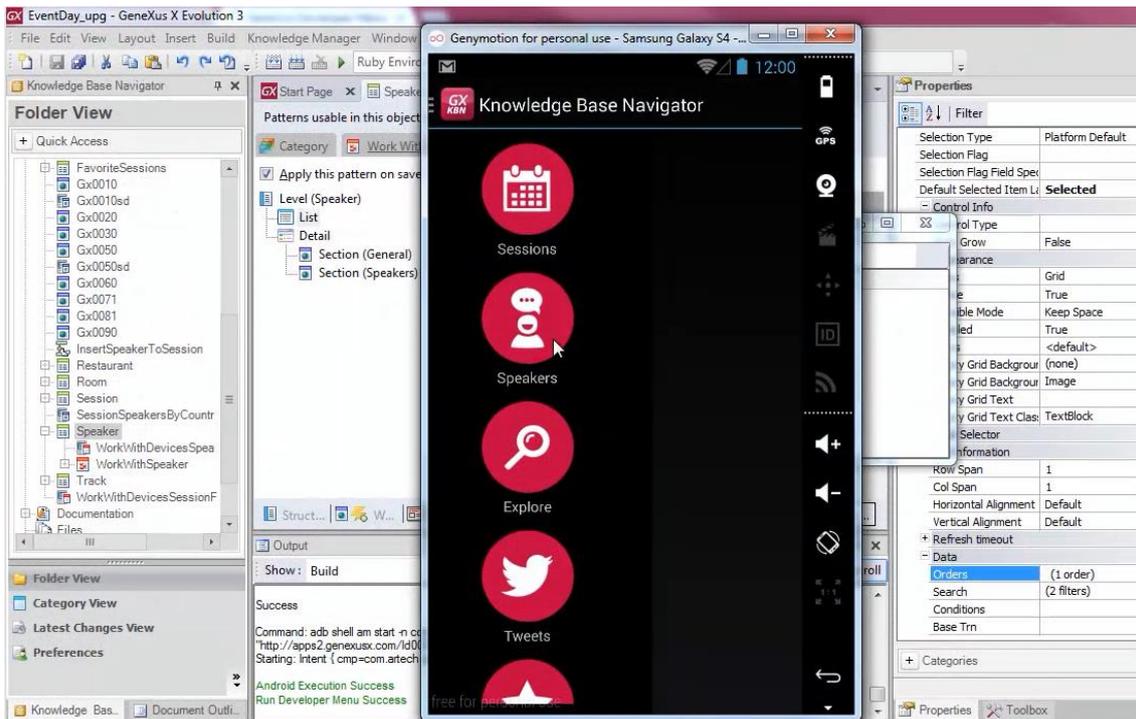
Por el atributo: SpeakerFullName



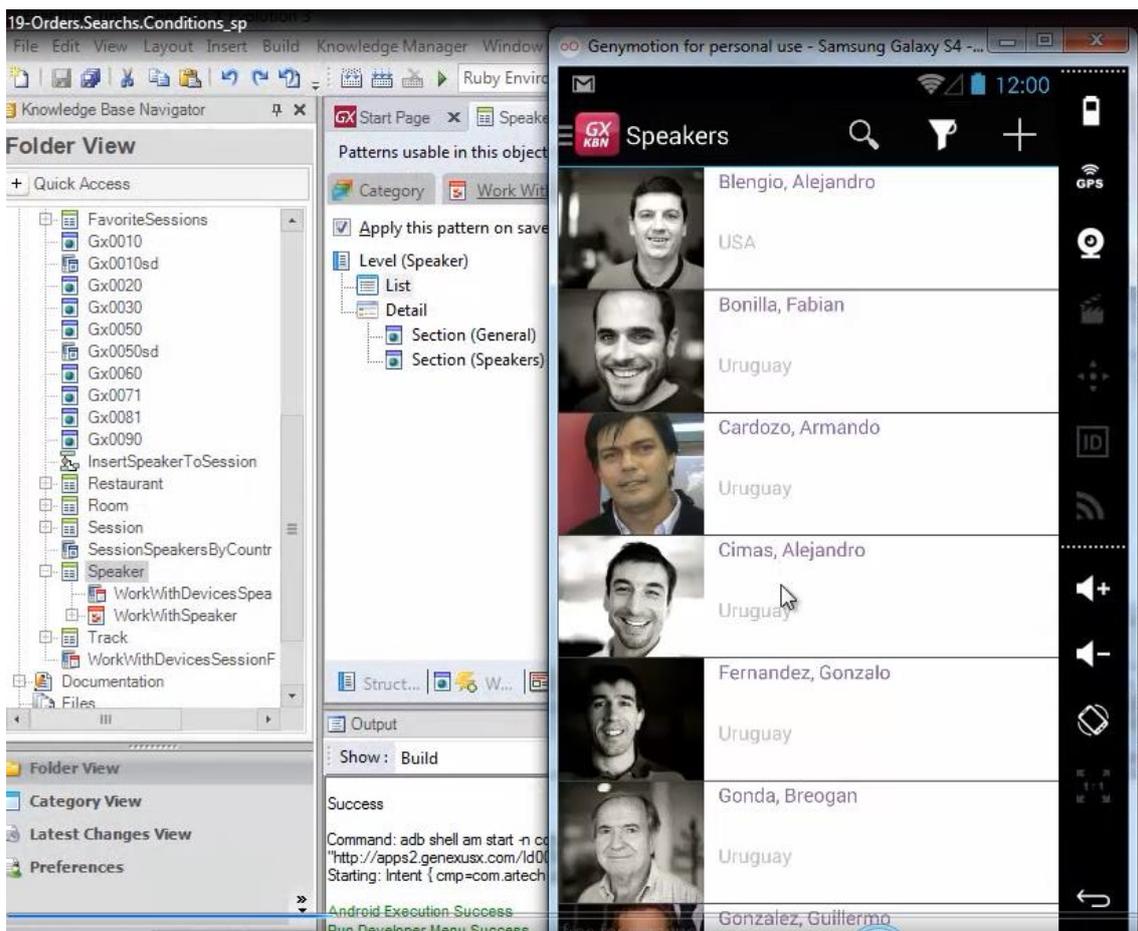
Y vamos a eliminar de su composición el SpeakerName:



Hagamos F5 y apreciemos el cambio.

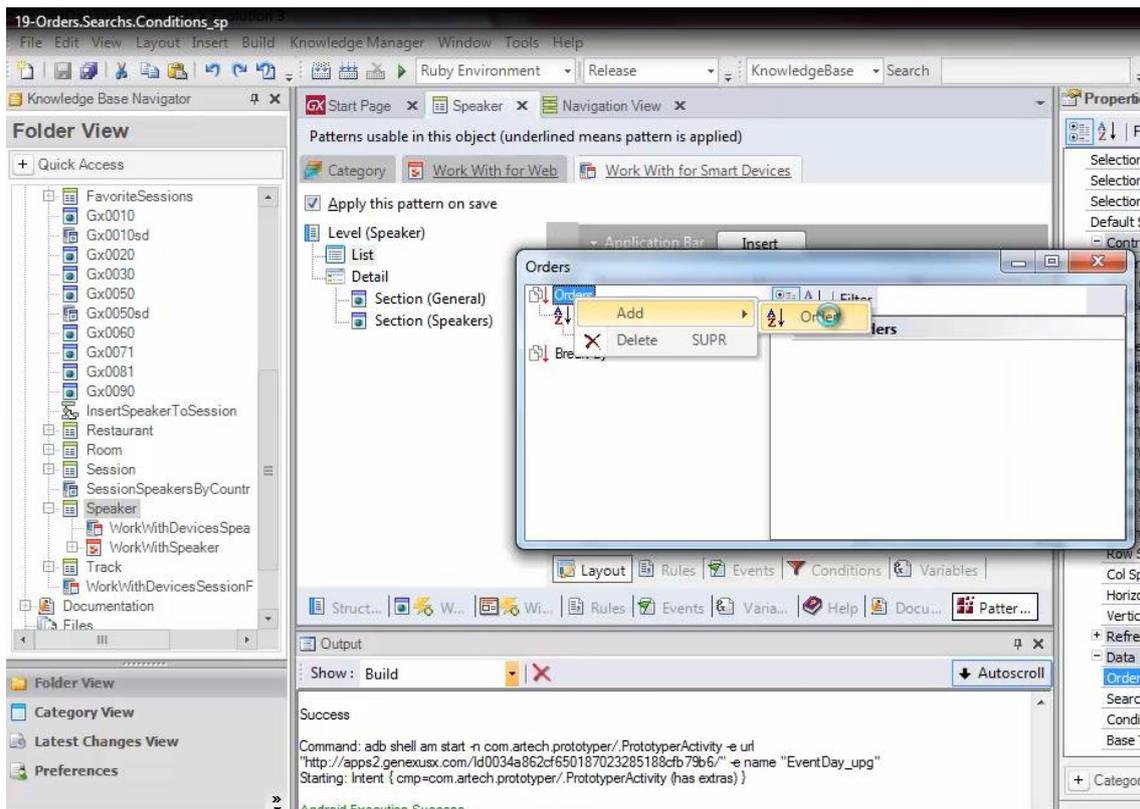


Y aquí lo vemos

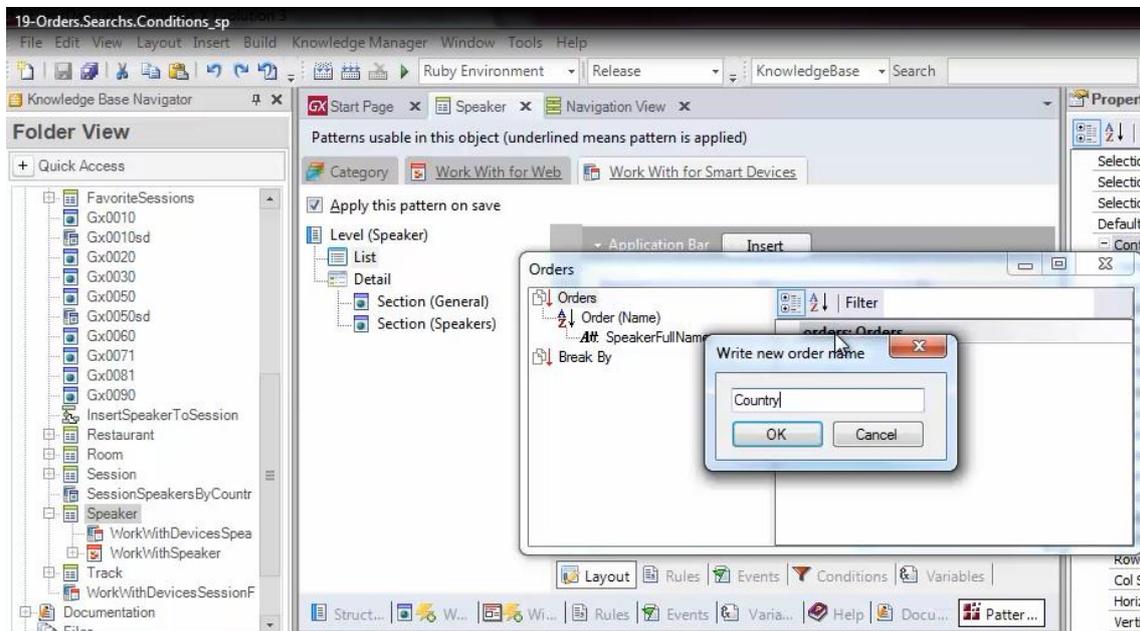


Supongamos ahora que queremos dar al usuario la posibilidad de elegir entre ordenar por nombre completo u ordenar por país del orador.

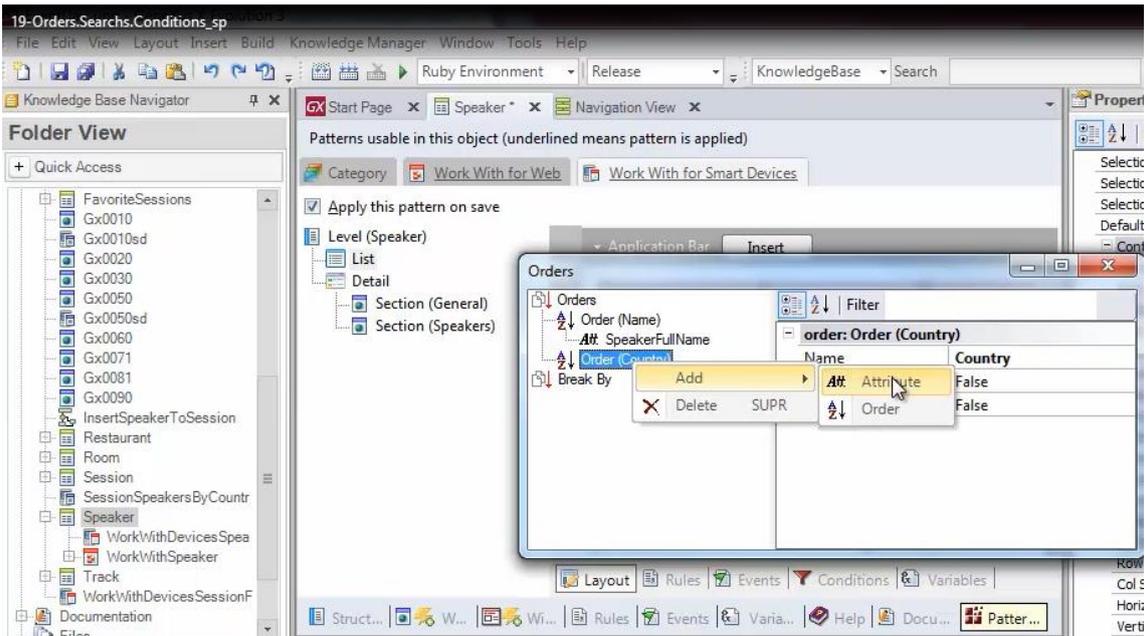
Para ello, volviendo a GeneXus, vamos a tener que definir otro orden



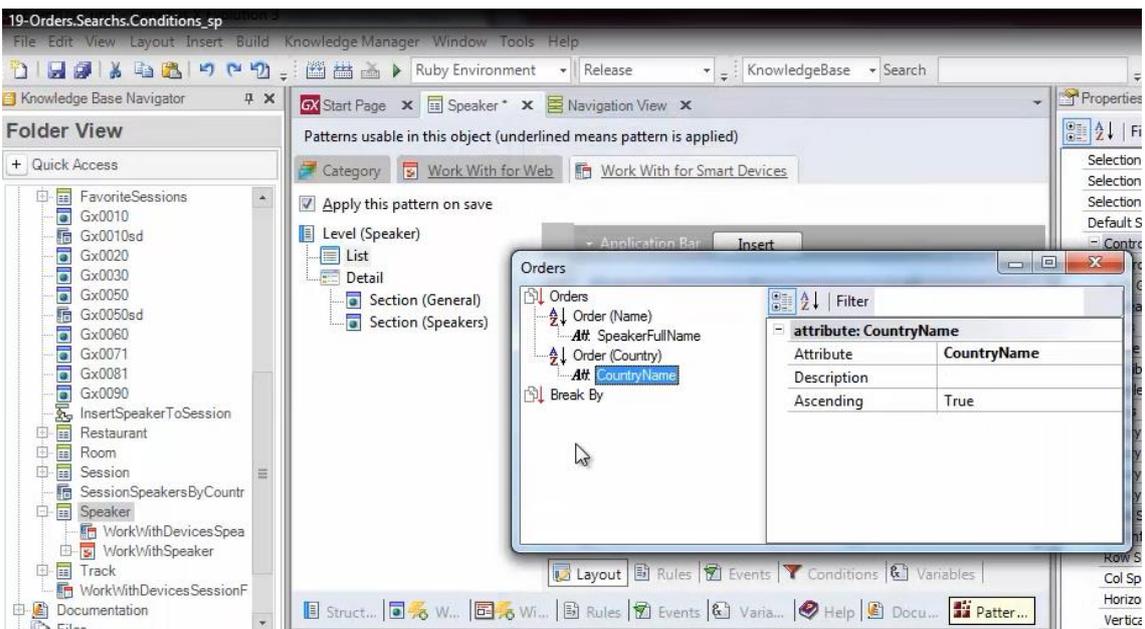
Al que le vamos a llamar: Country



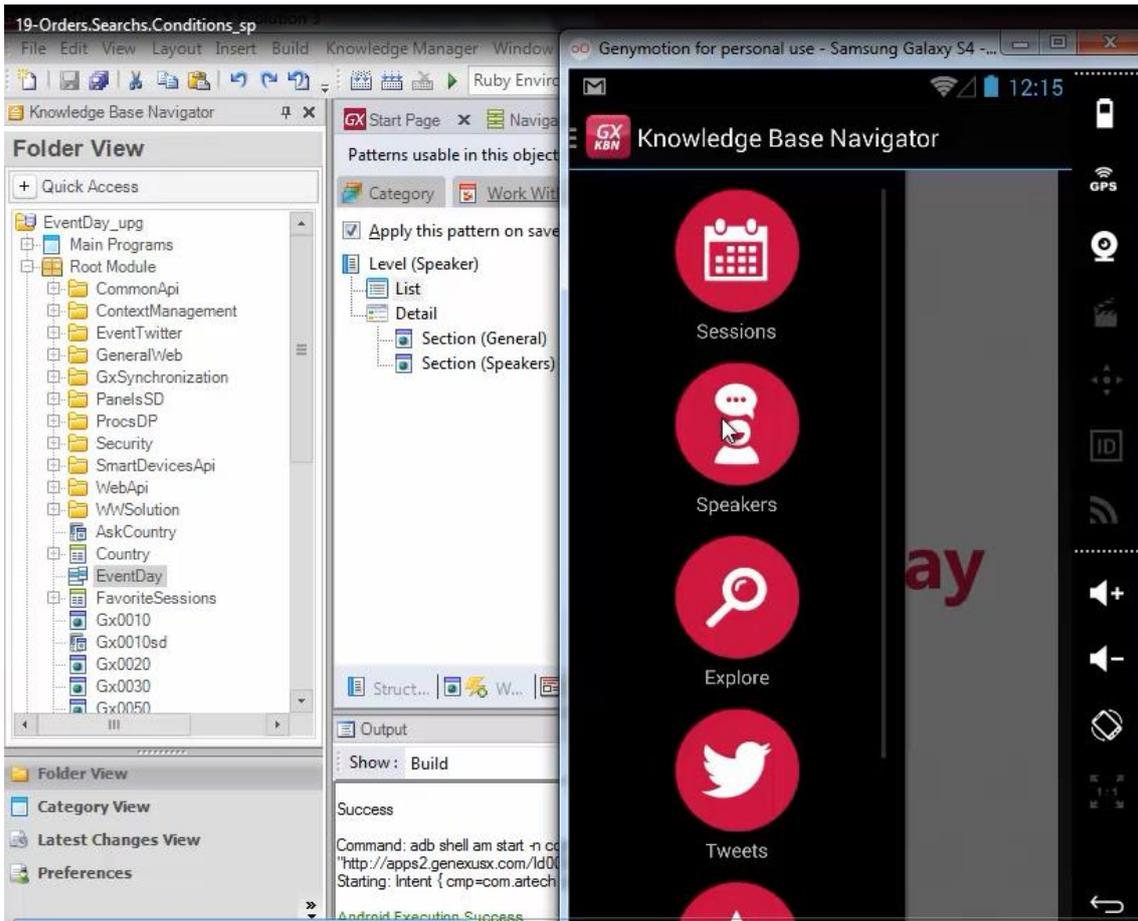
y donde vamos a definir su composición



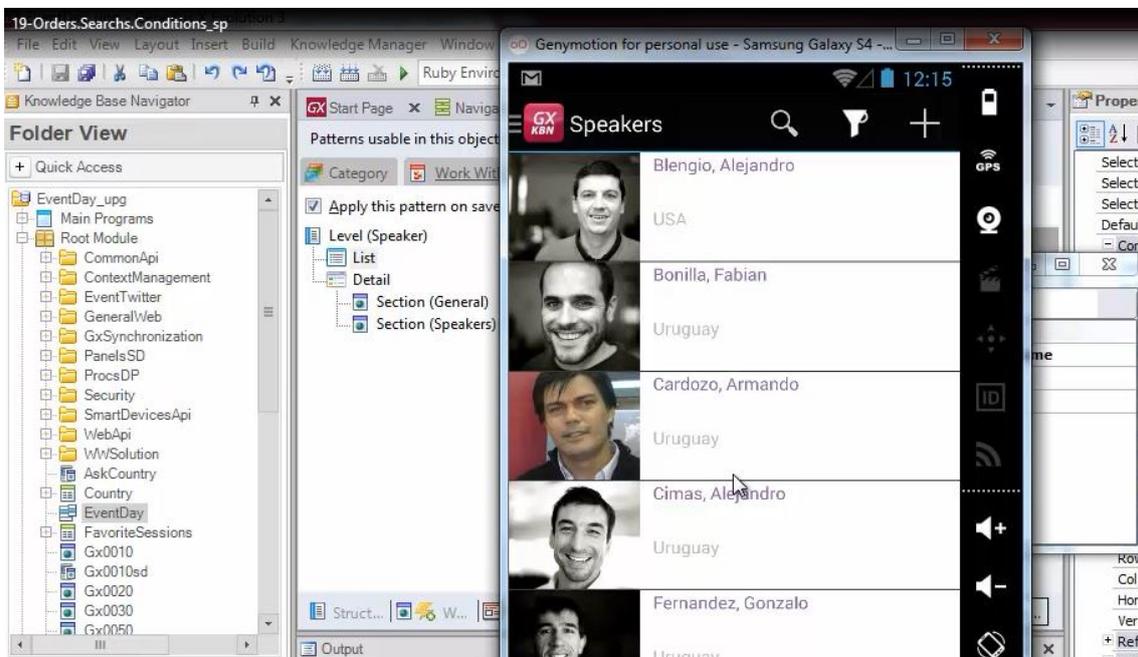
Por el atributo: CountryName



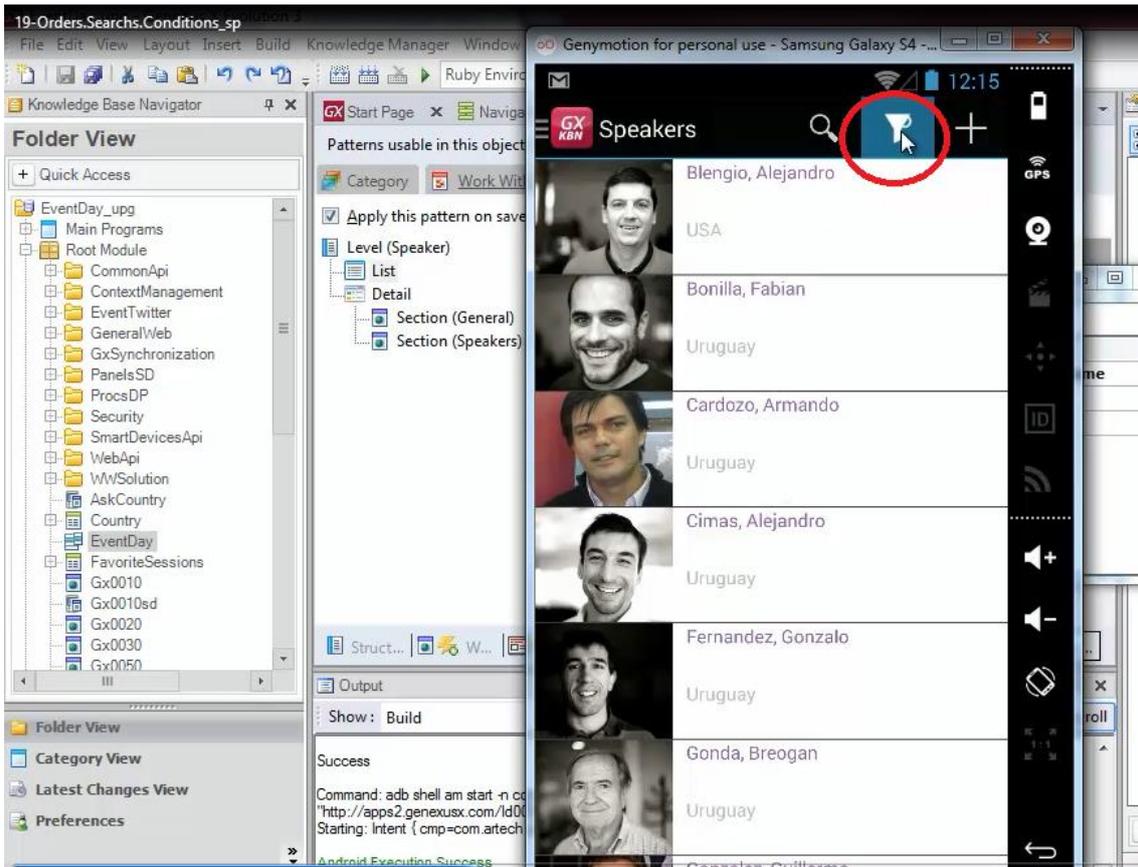
Damos F5



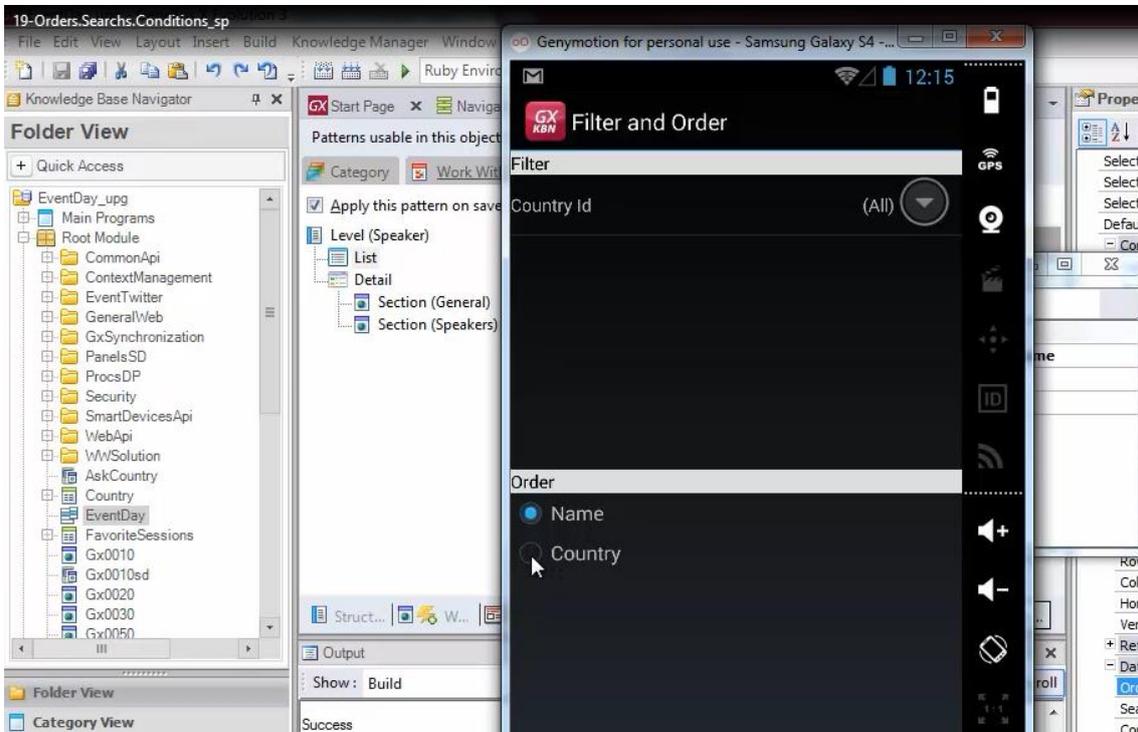
y vemos que por defecto se sigue ordenando por el SpeakerFullName

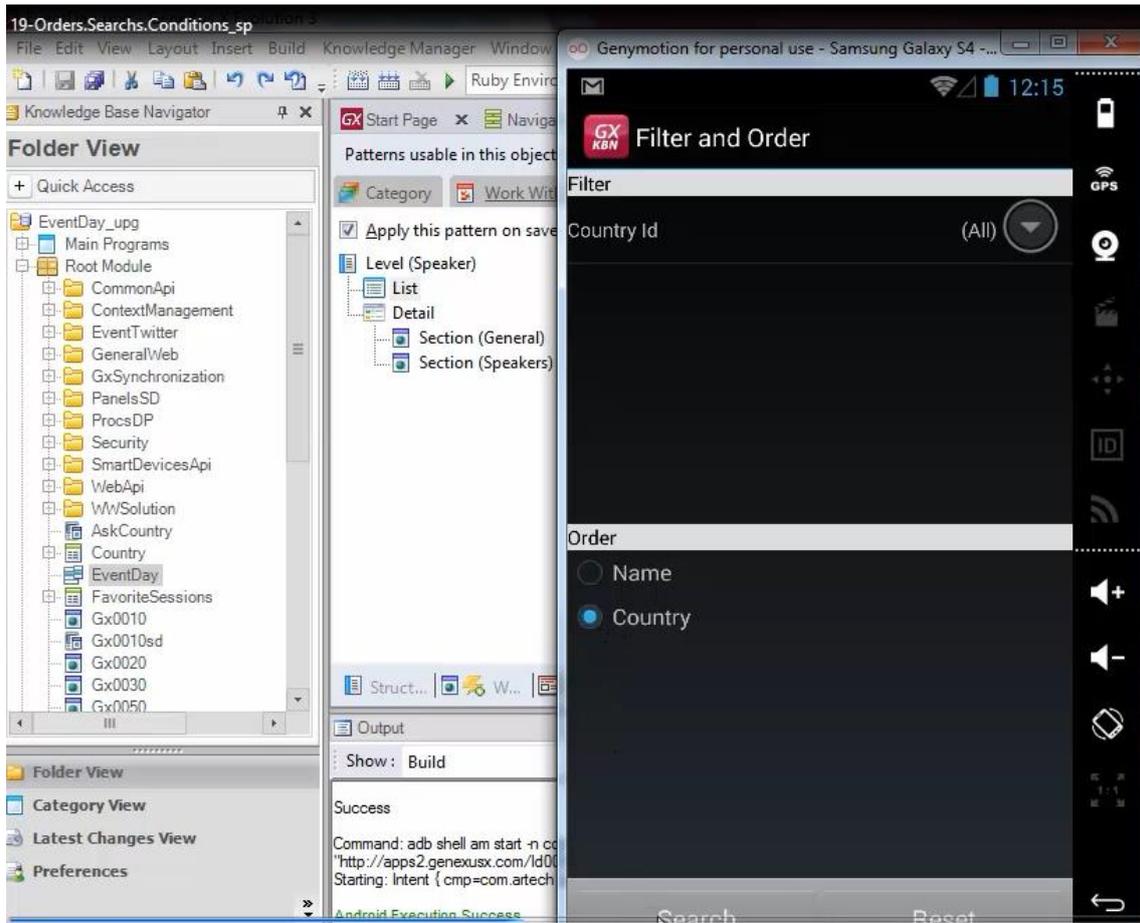


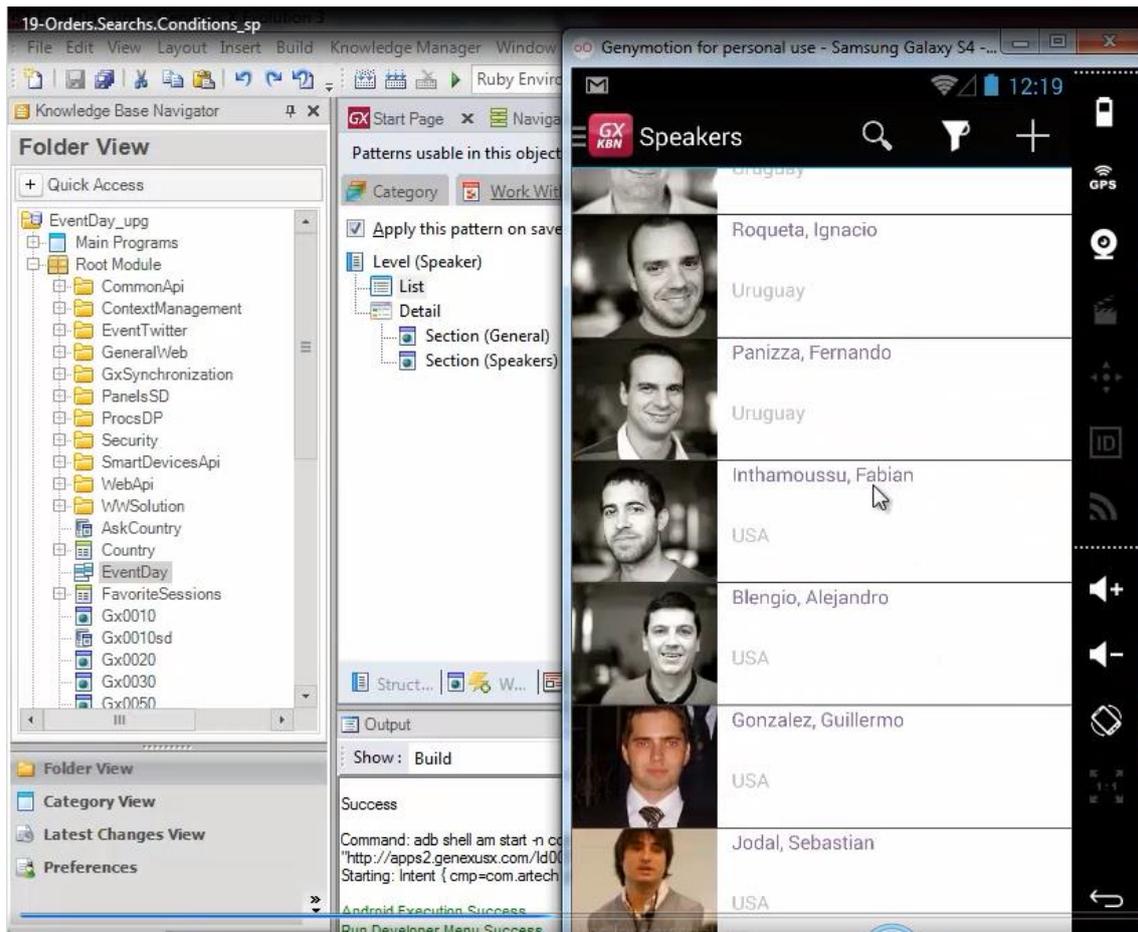
pero si vamos a observar los filtros



encontramos la posibilidad de ordenar por Country:

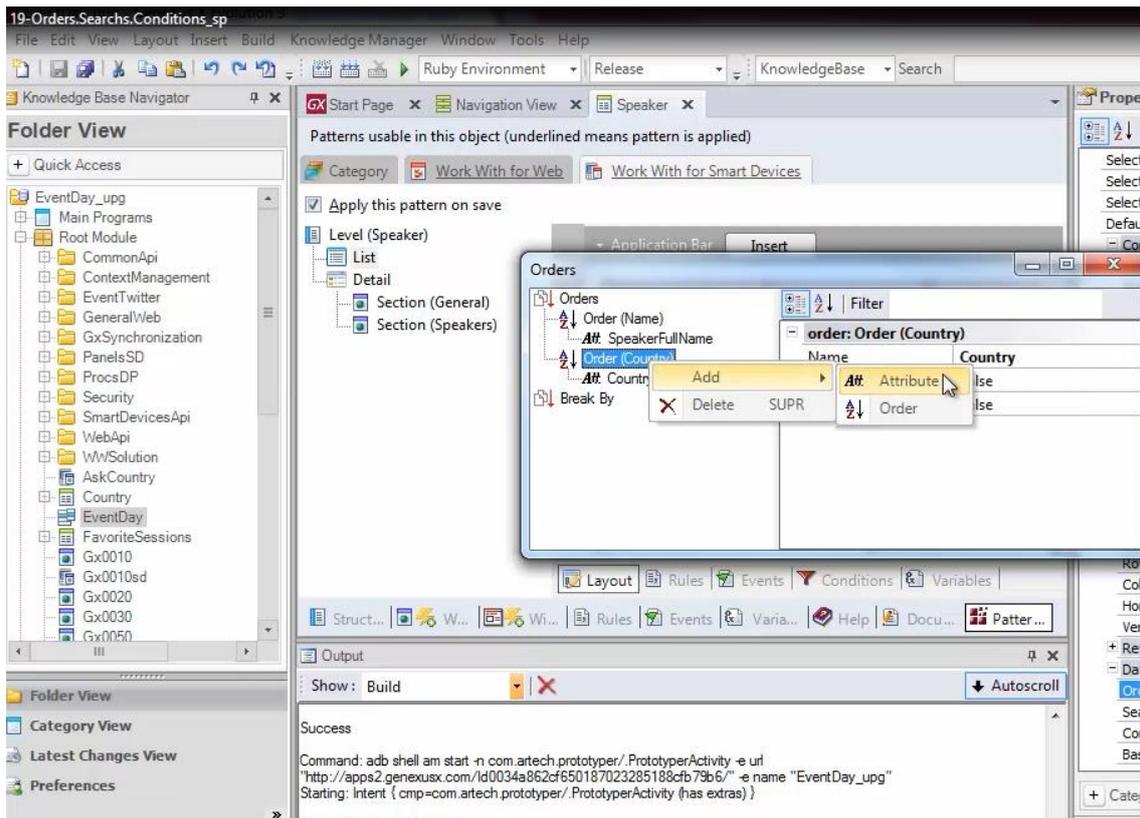




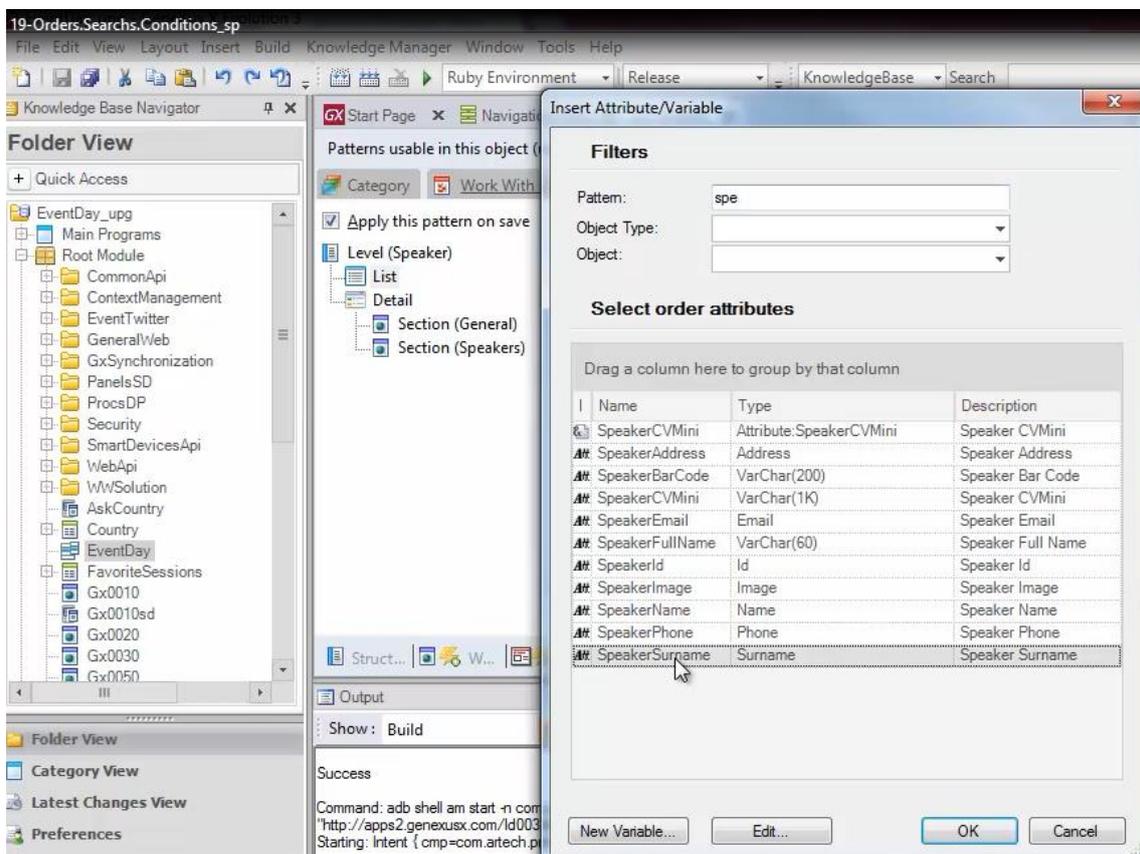


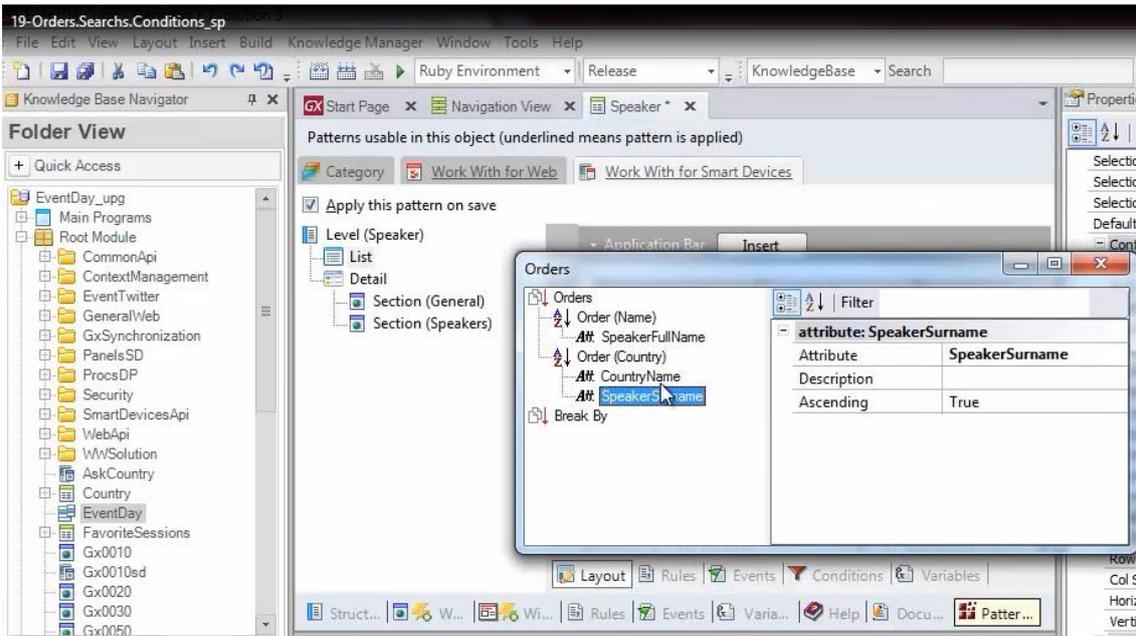
Uruguay primero.. y luego: USA

Si queremos que dentro del país, se ordene por apellido, volvemos a GeneXus, y agregamos..

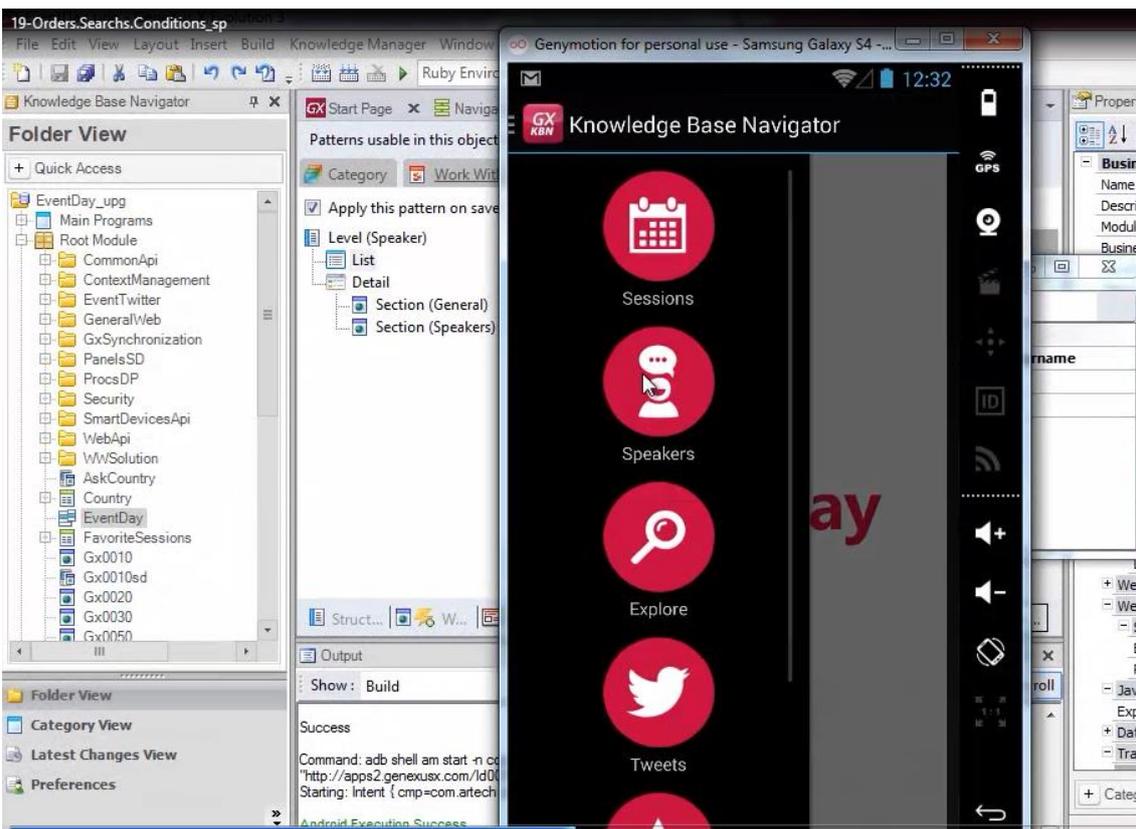


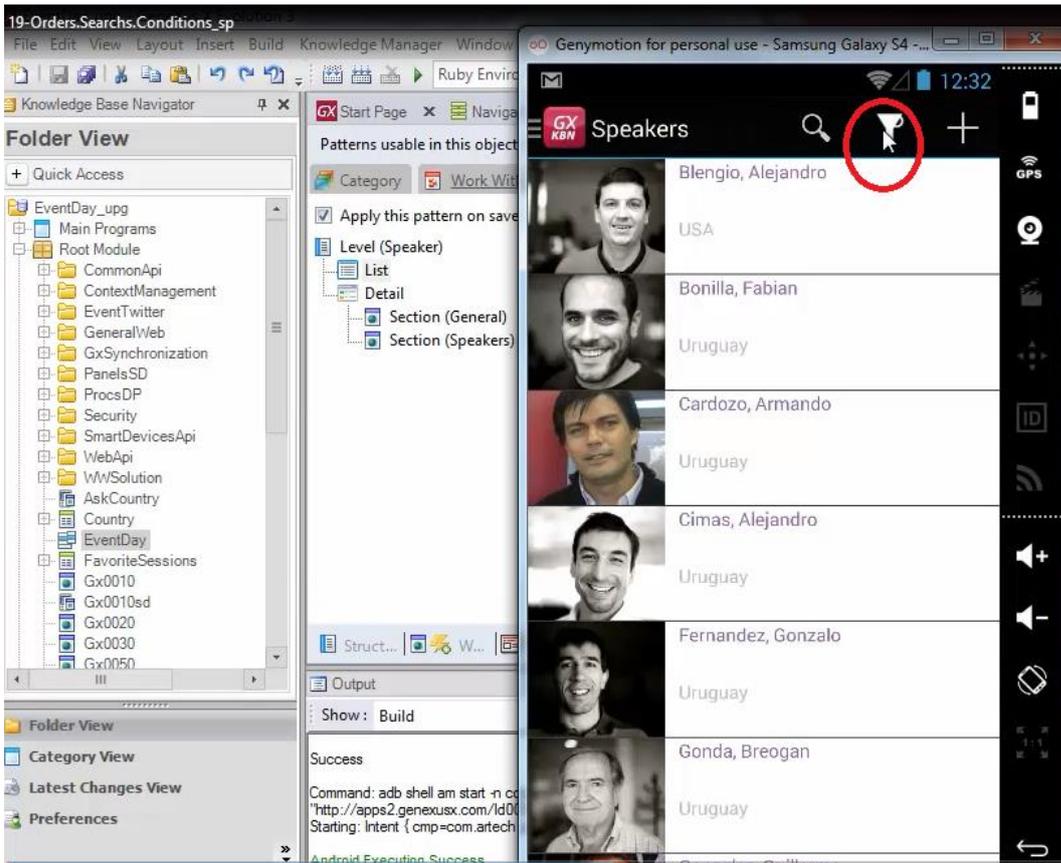
El segundo atributo: SpeakerSurname



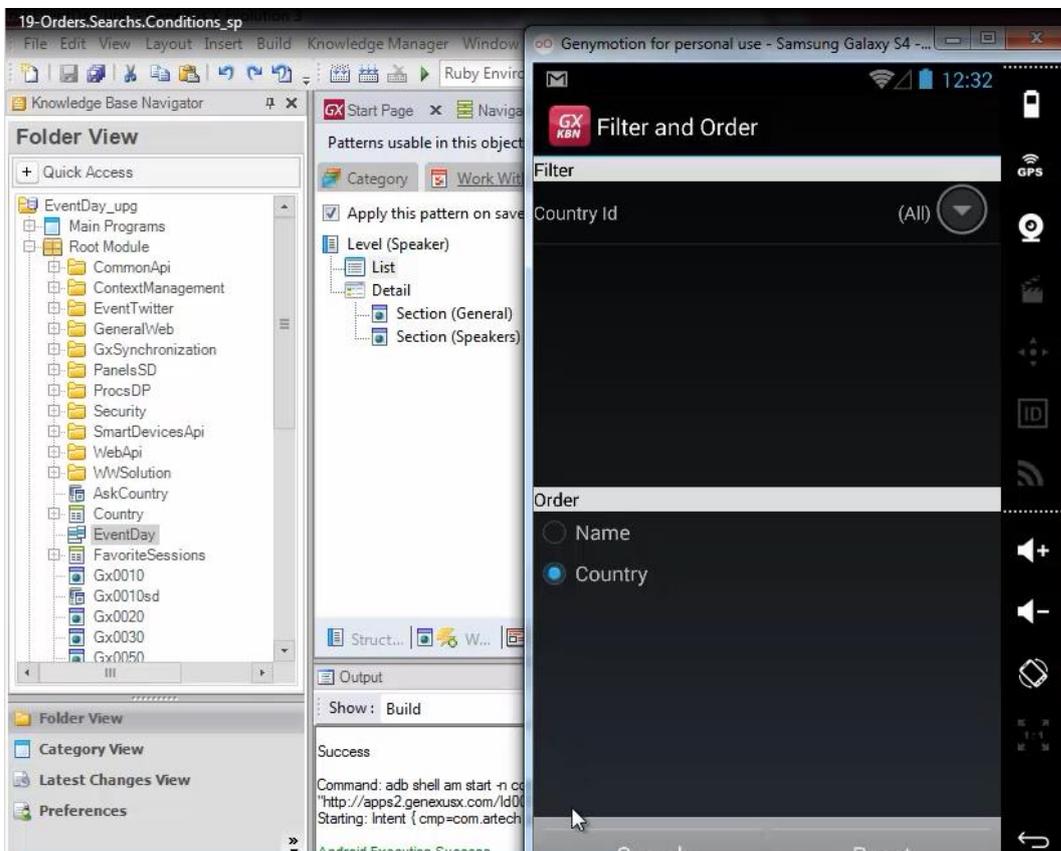


por ejemplo.. damos F5..

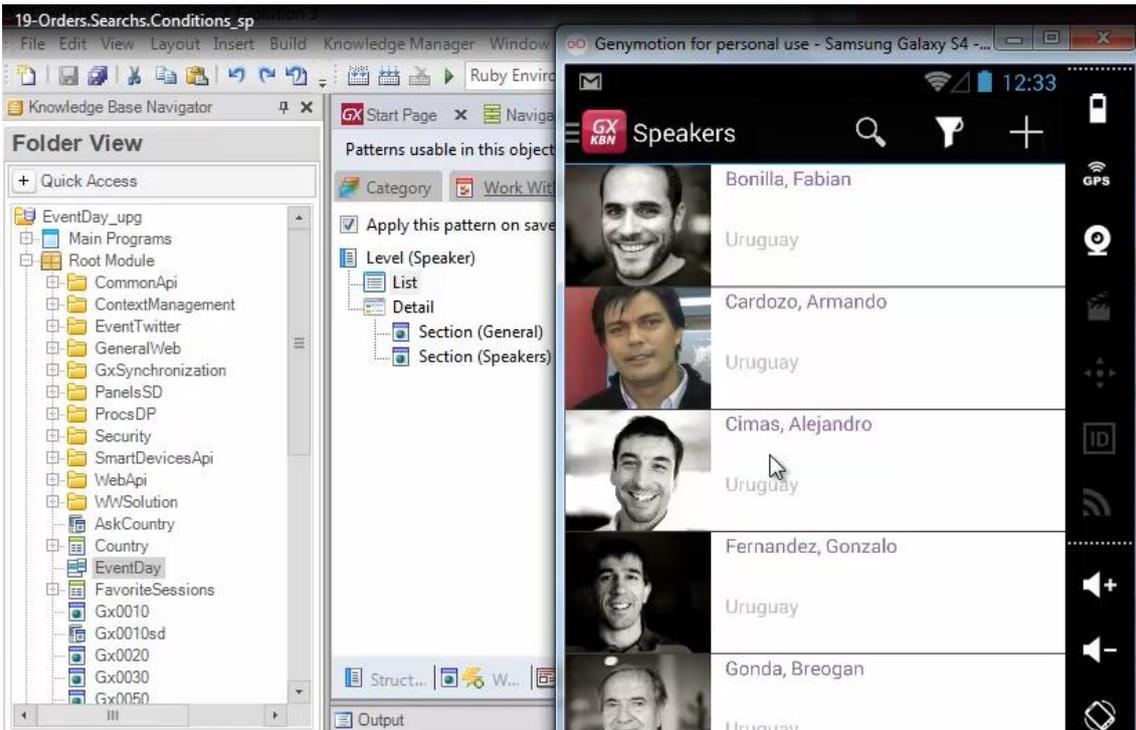




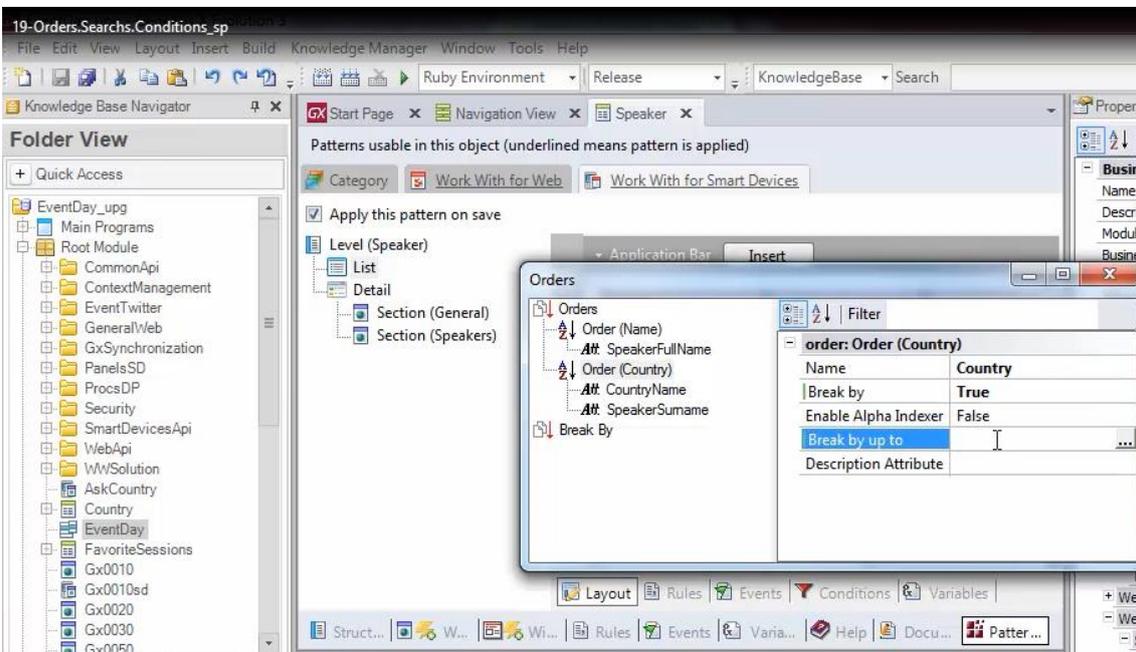
Ordenamos por Country



y vemos cómo está saliendo ordenado además dentro de país, por SpeakerFullName

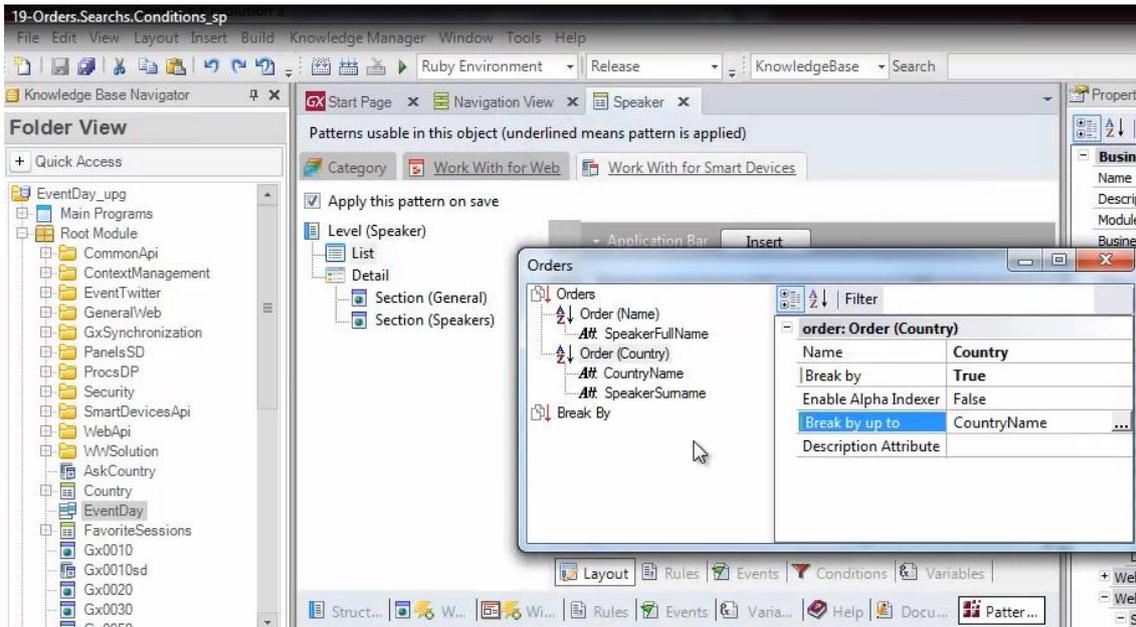


Si ahora queremos que además de ordenarse por CountryName, se agrupe la información por CountryName, es decir, que queden separados de alguna manera los oradores de Uruguay de los oradores de USA, volvemos a GeneXus... y a nivel de la definición del orden.. establecemos que se va a hacer un Break by



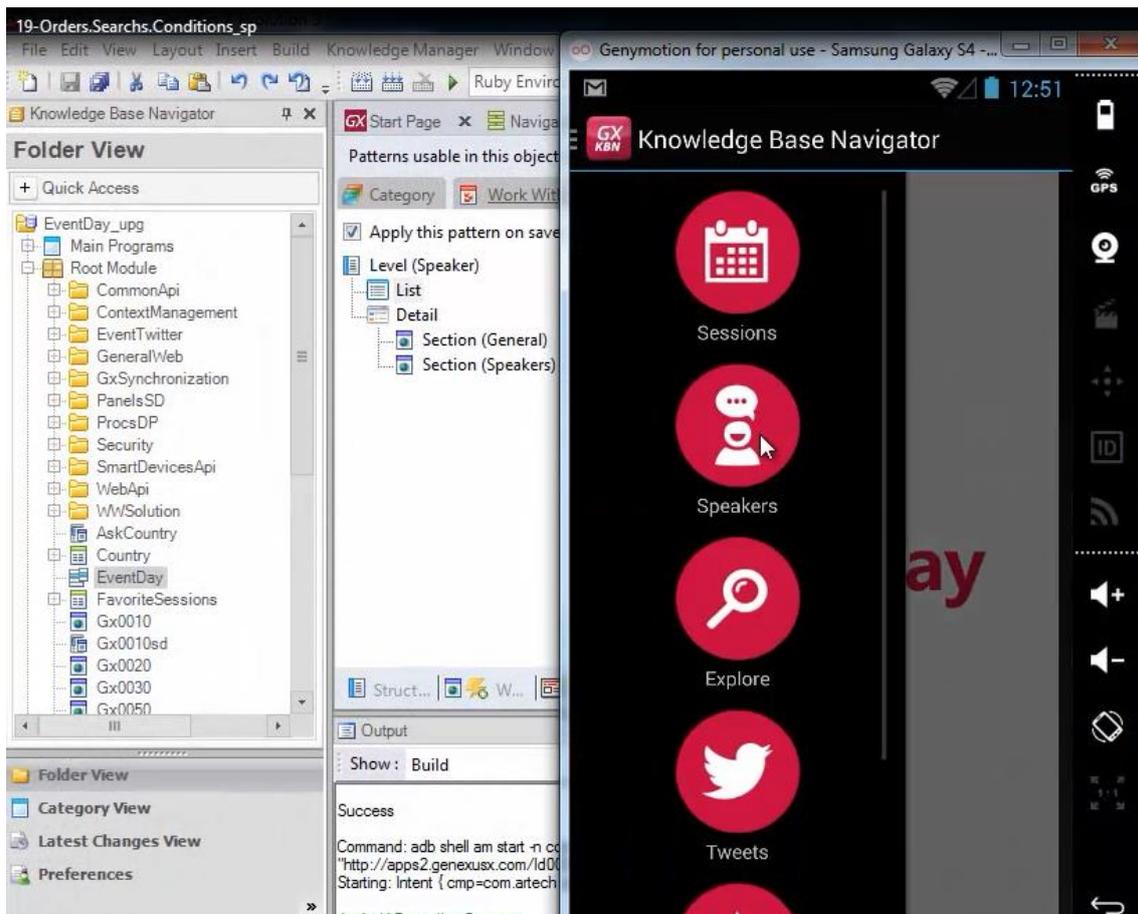
¿por qué atributo?

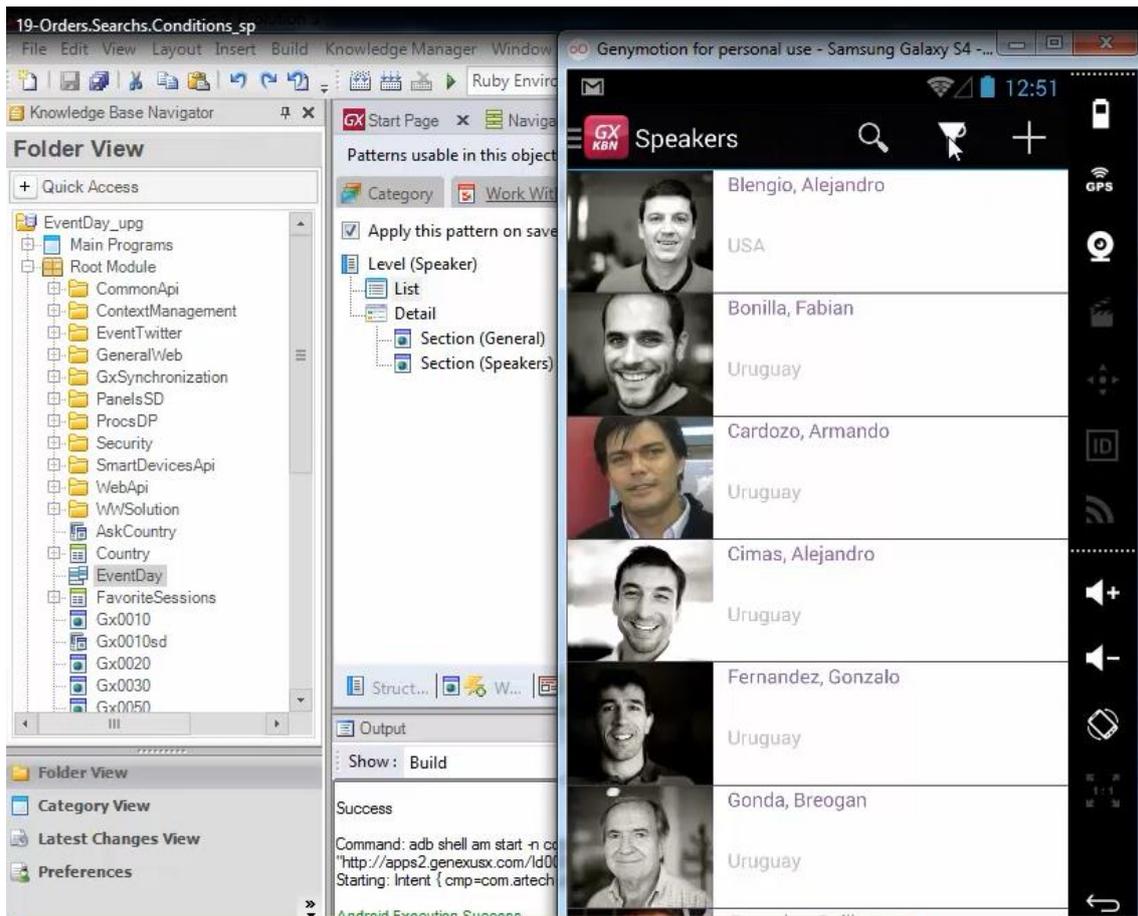
por todos desde el principio de la lista del orden, hasta: **CountryName**



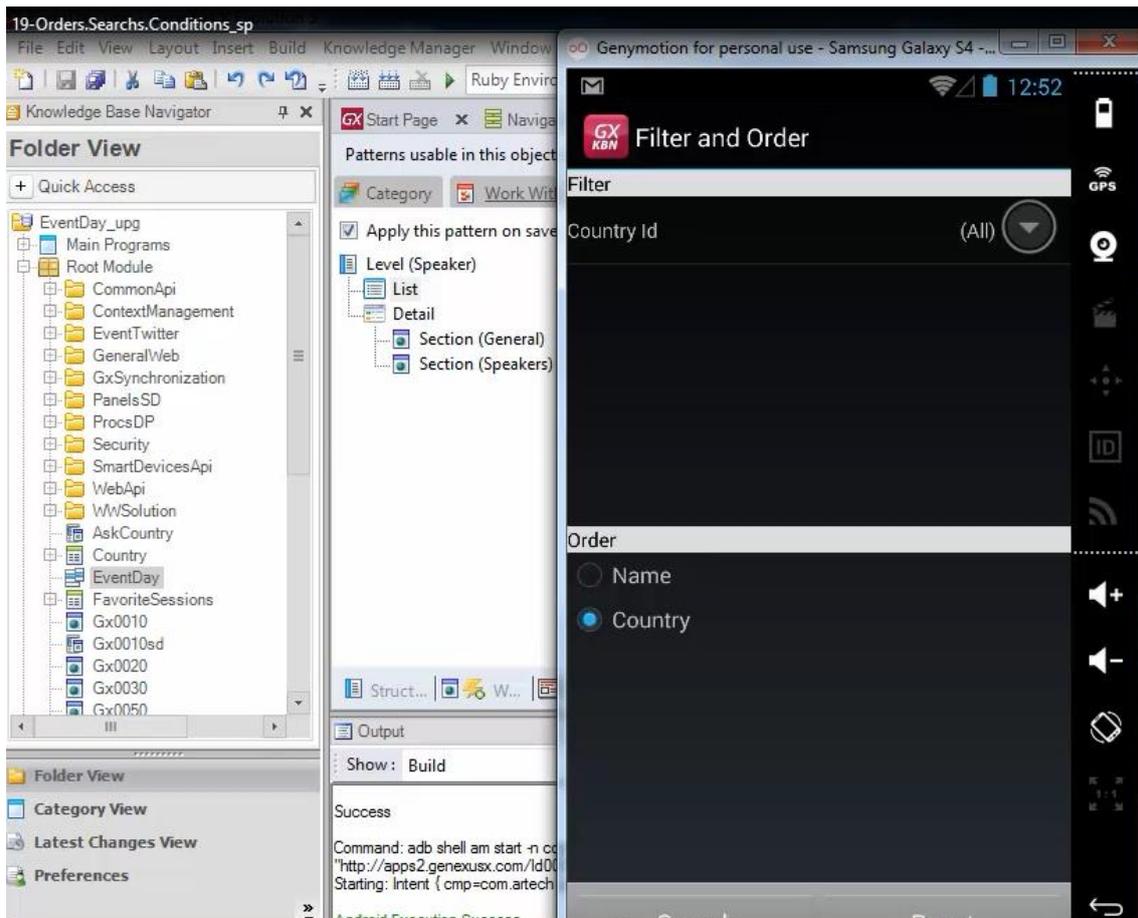
es decir, los atributos que vengan después del que especificamos aquí en la lista, van a servir para ordenar la información pero no para cortar por ella.

Hagamos F5..

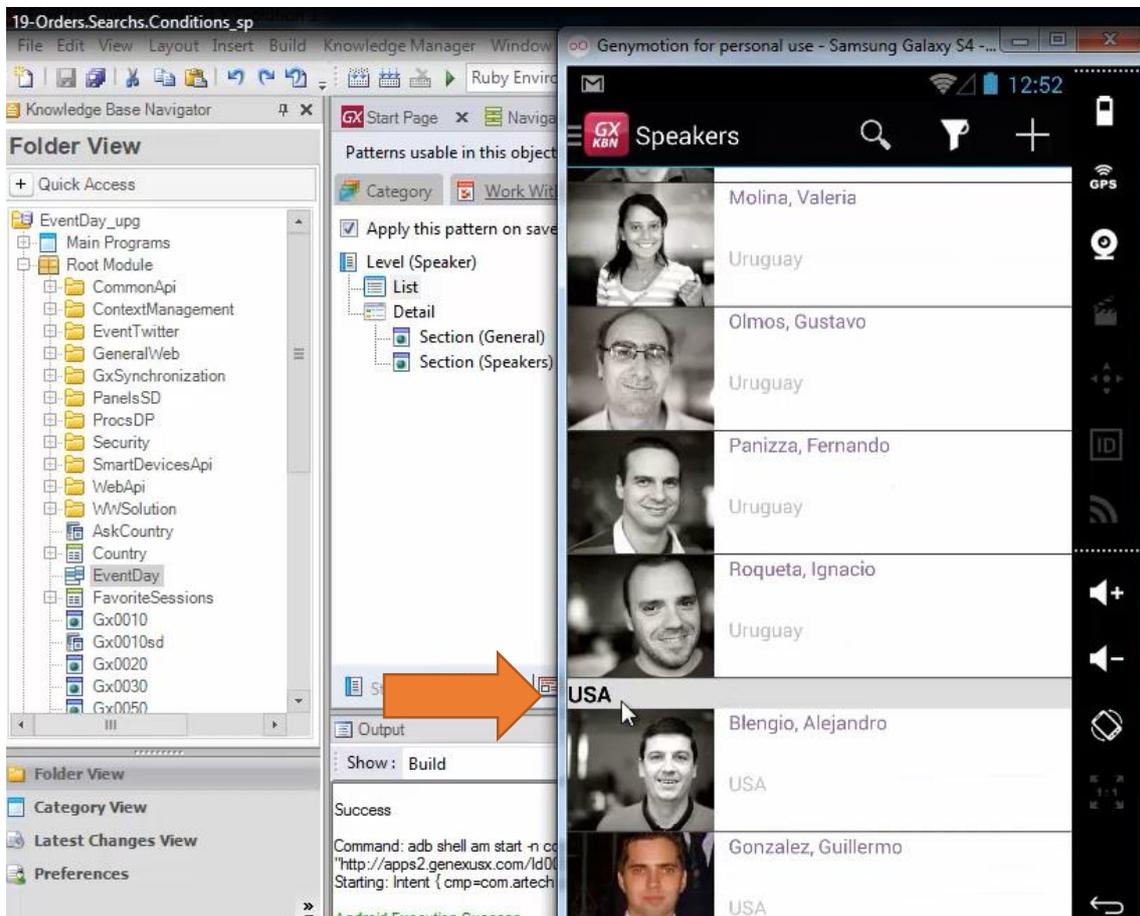
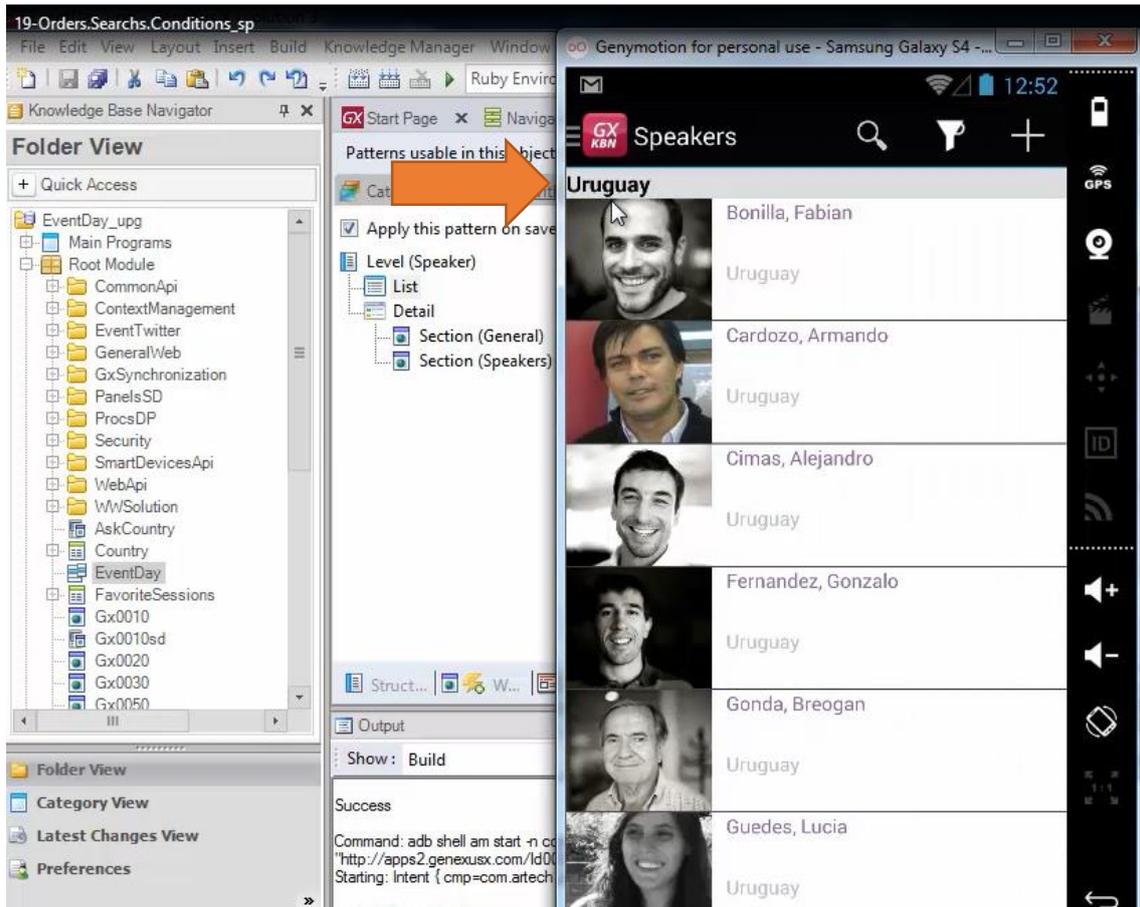




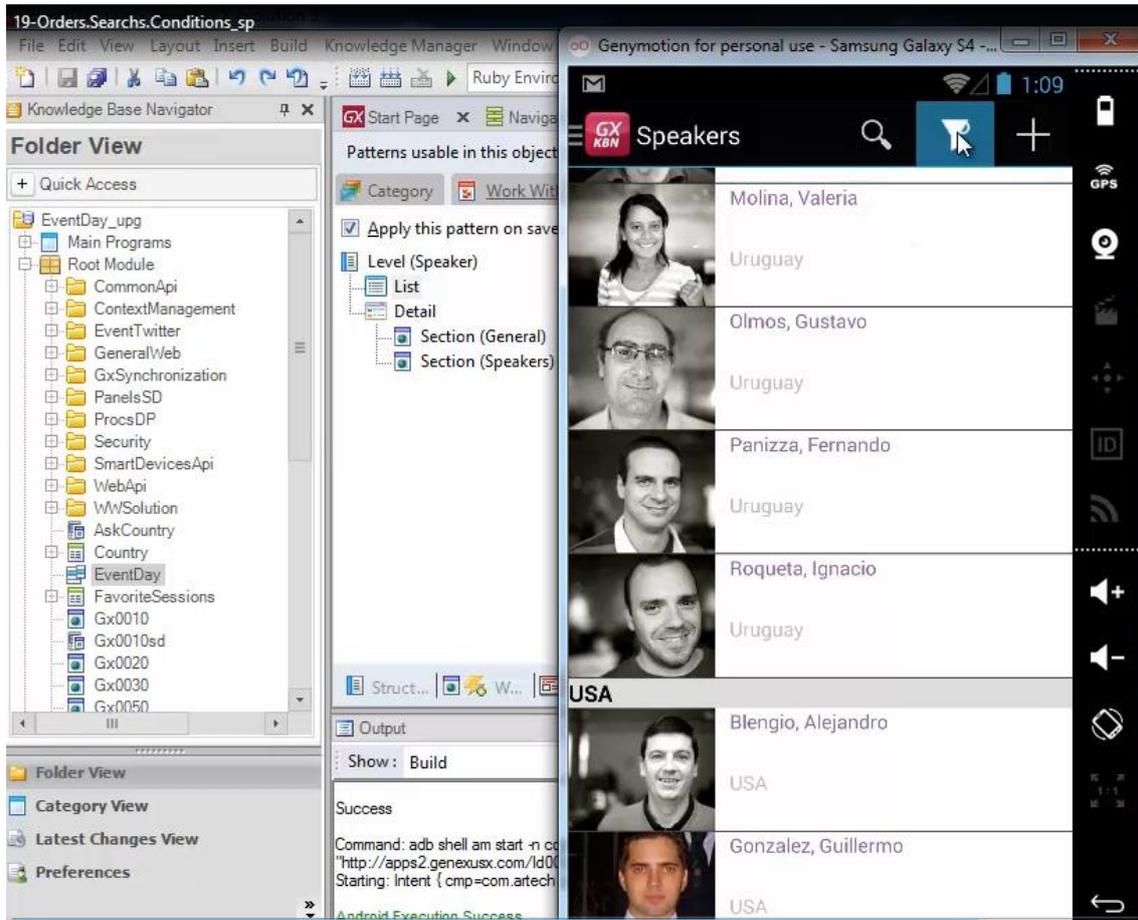
elijamos otra vez ordenar por país



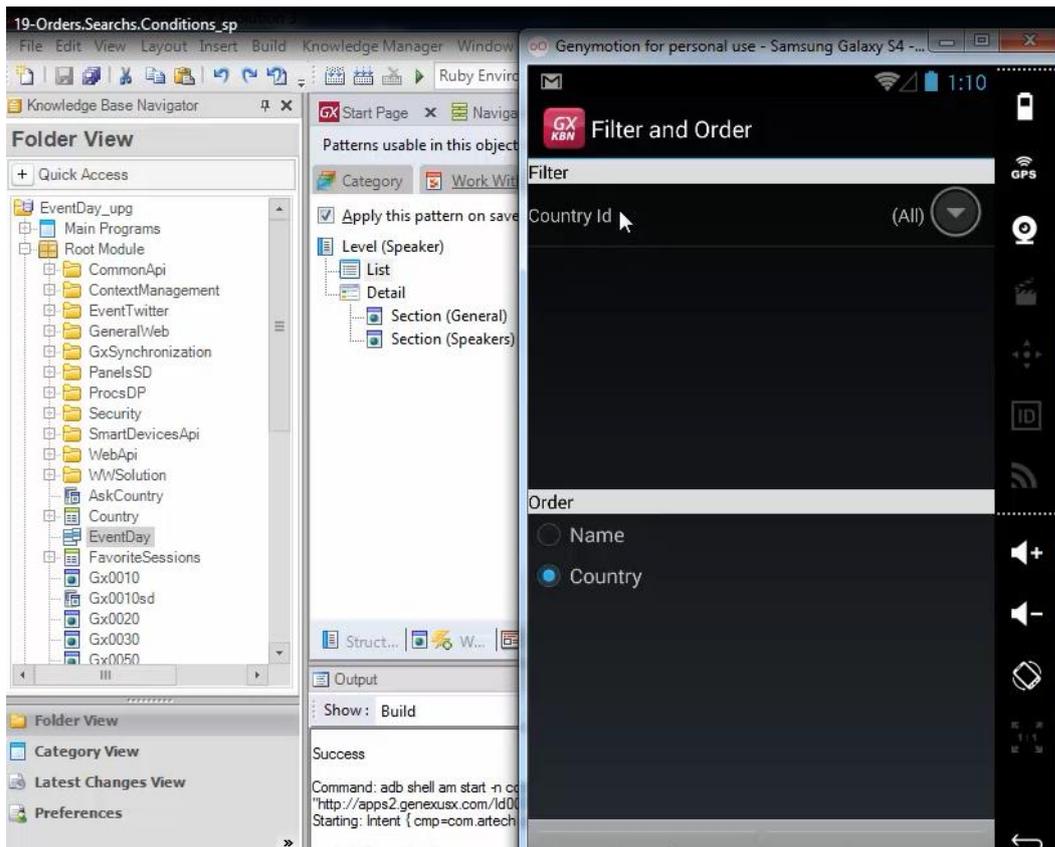
y ahora observemos que está saliendo como queríamos ordenada y **cortada** la información **por país**



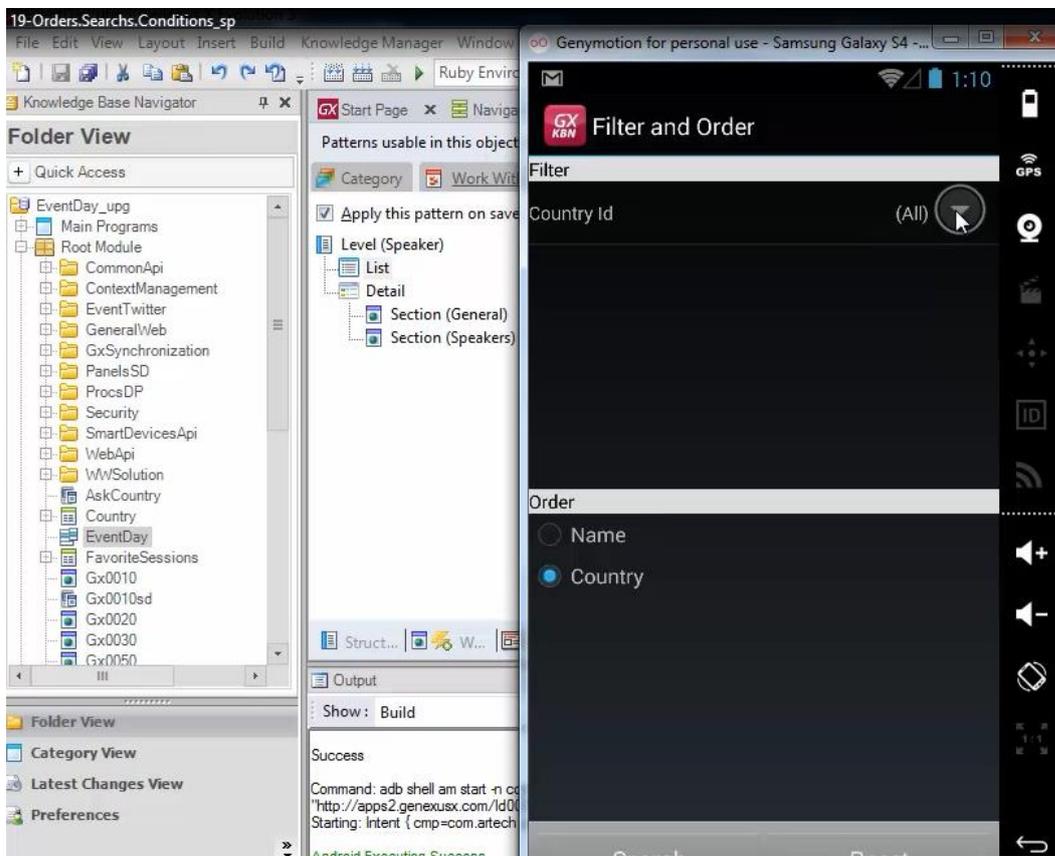
Reparemos ahora en los filtros:

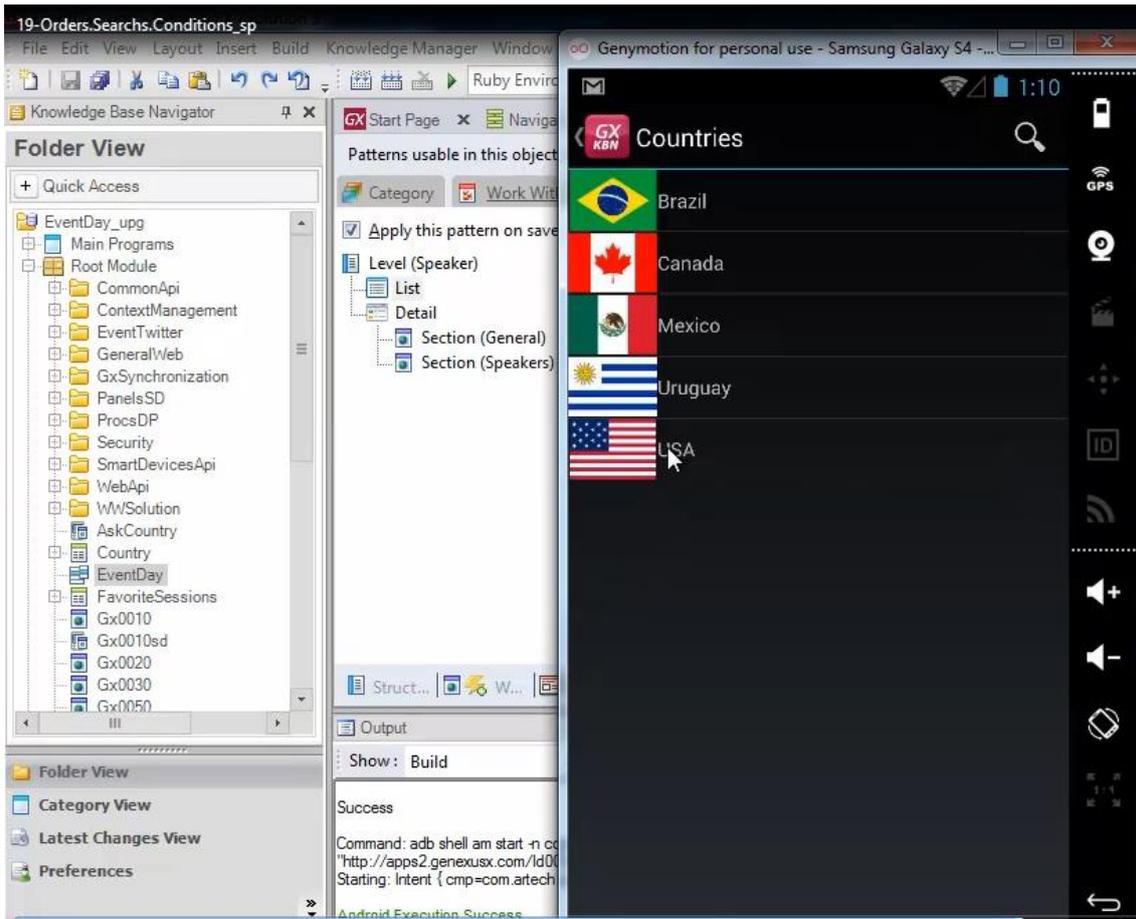


Por defecto al aplicar el pattern work with a la transacción Speaker, GeneXus nos agregó un filtro por la clave foránea CountryId

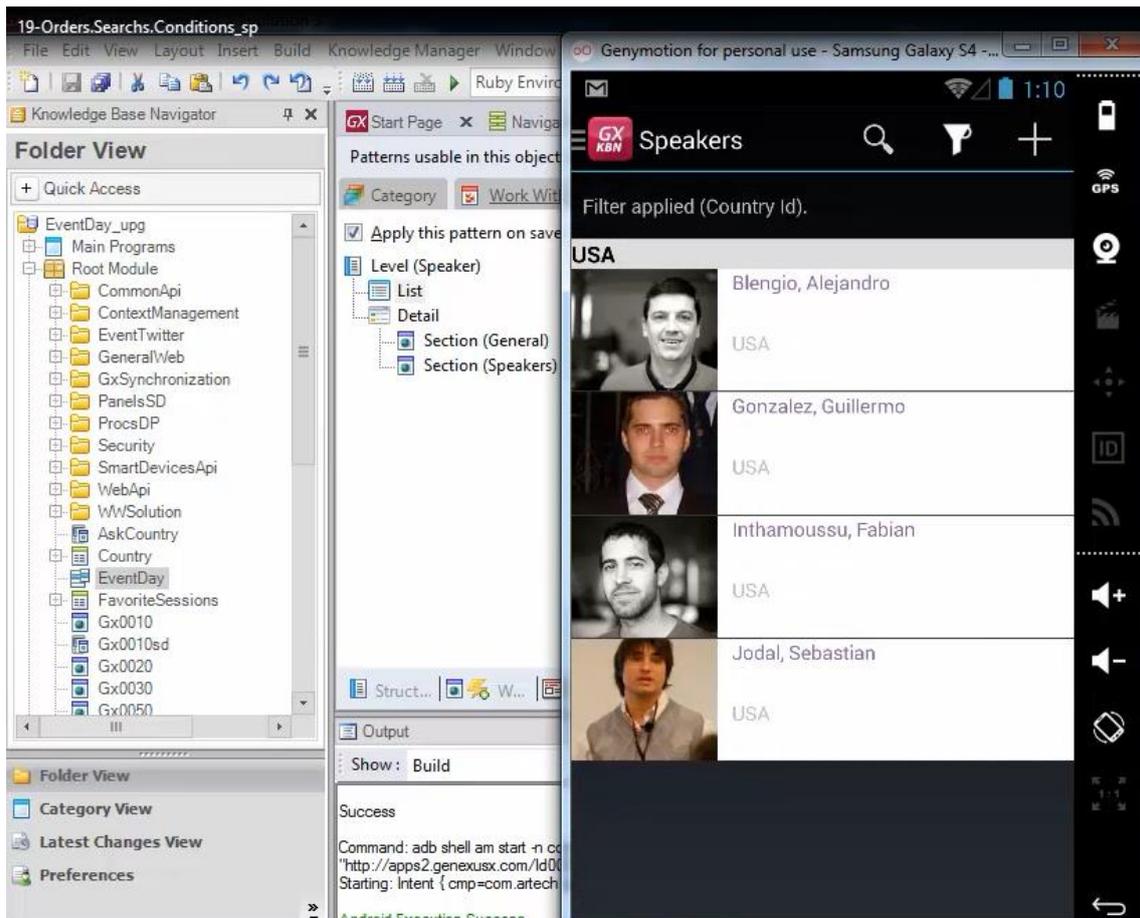


Por tanto, podemos elegir un país

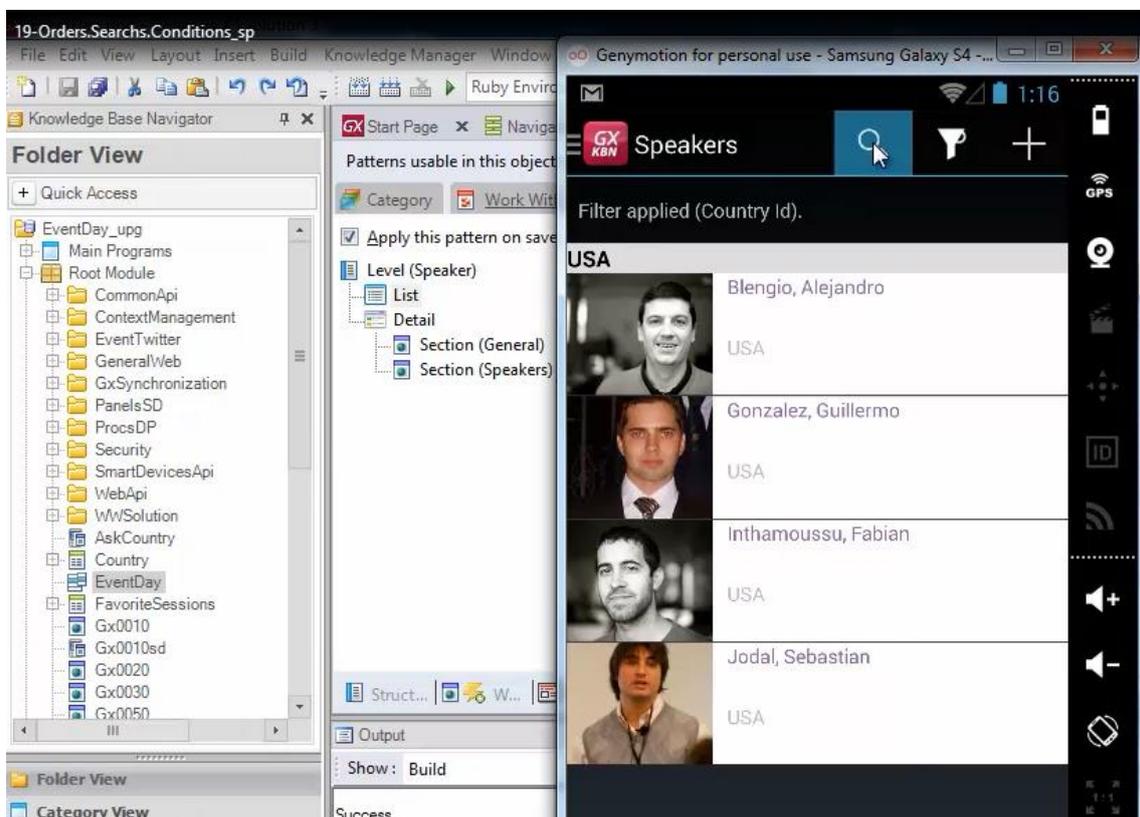


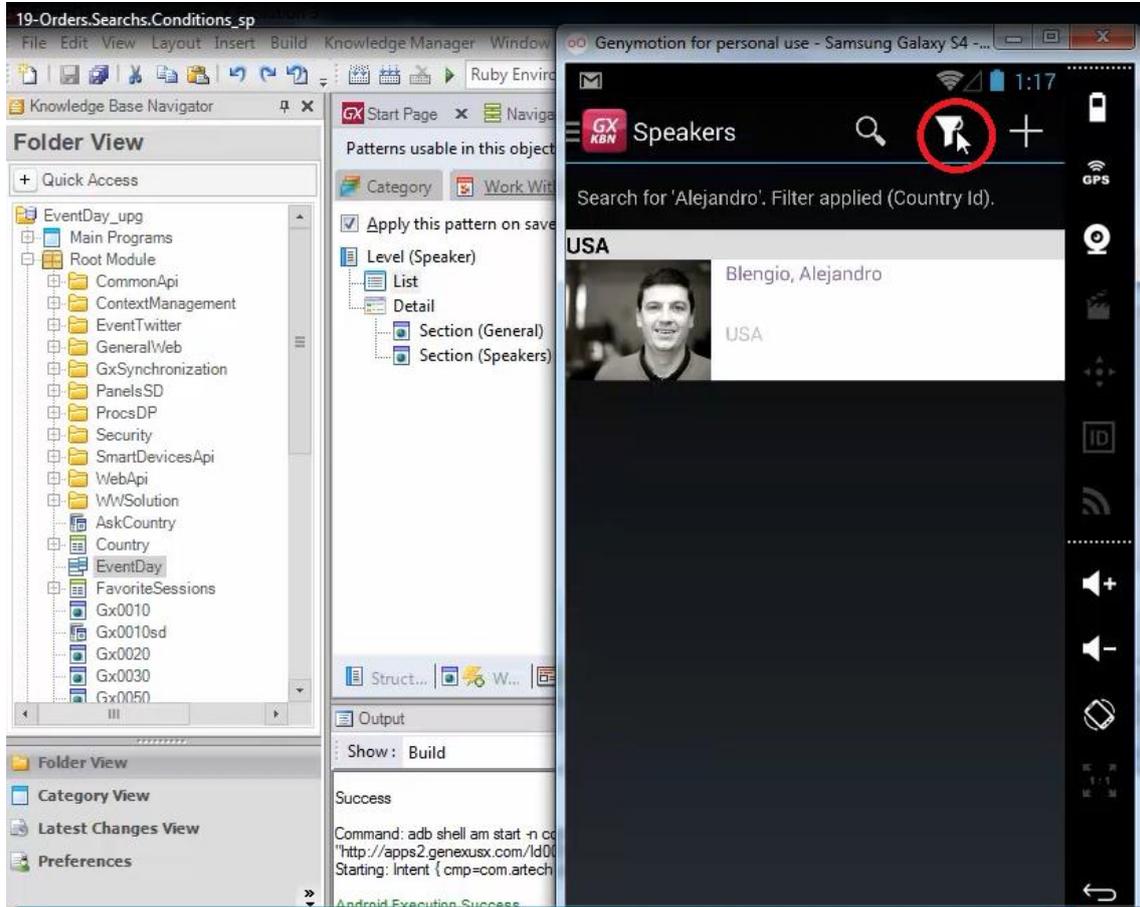
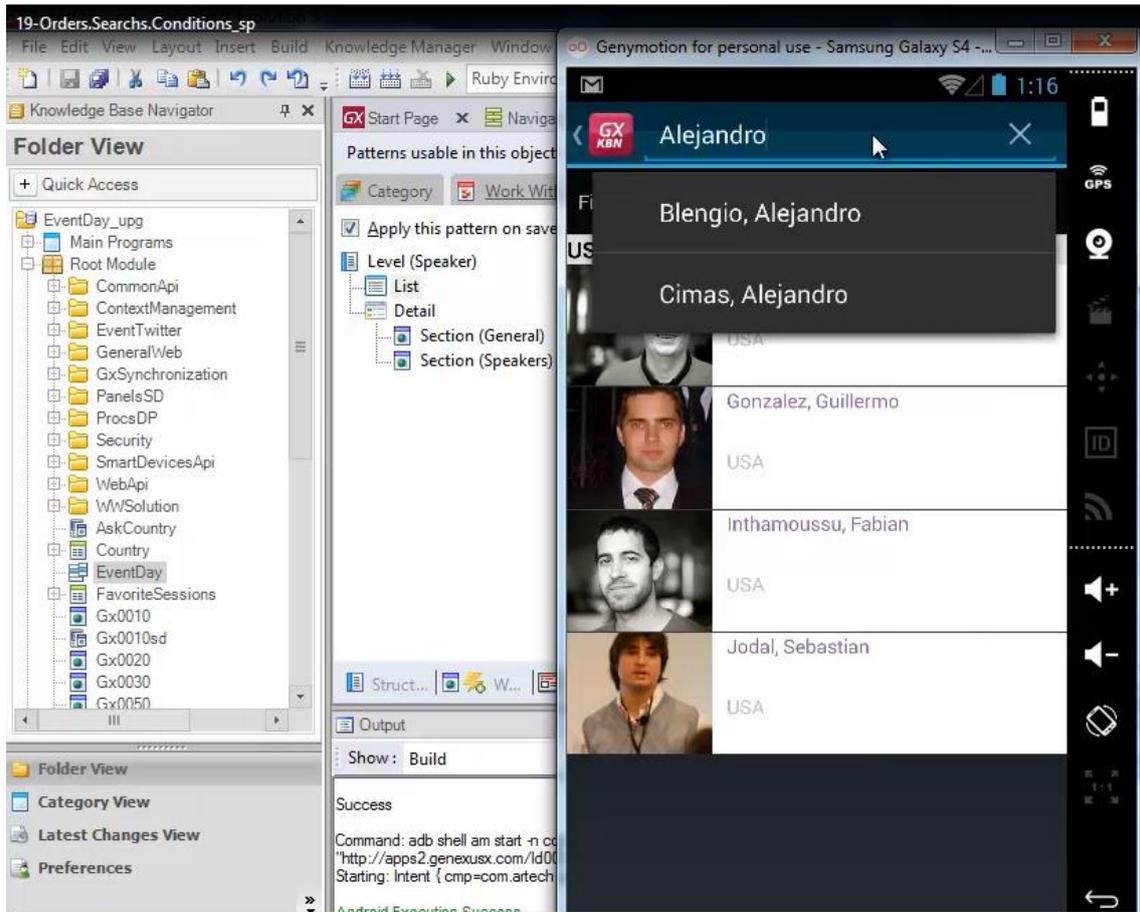


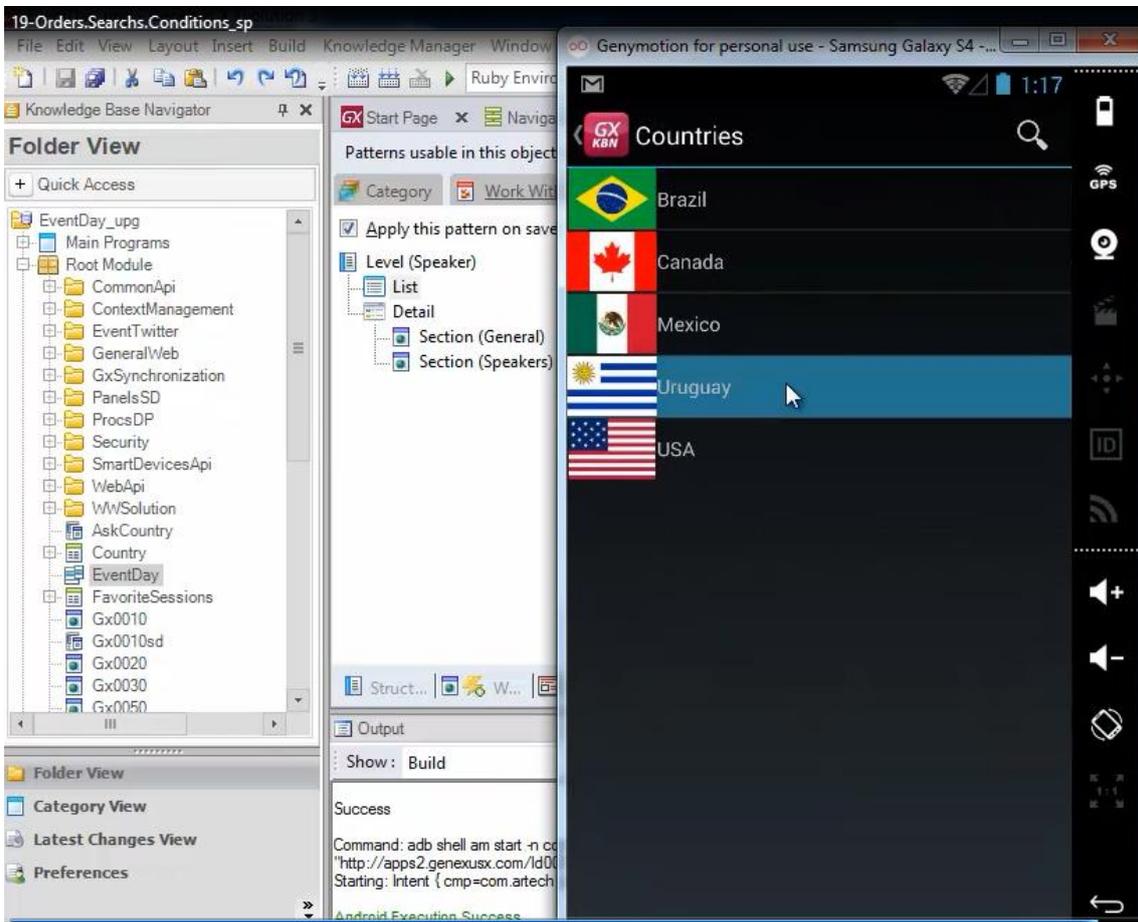
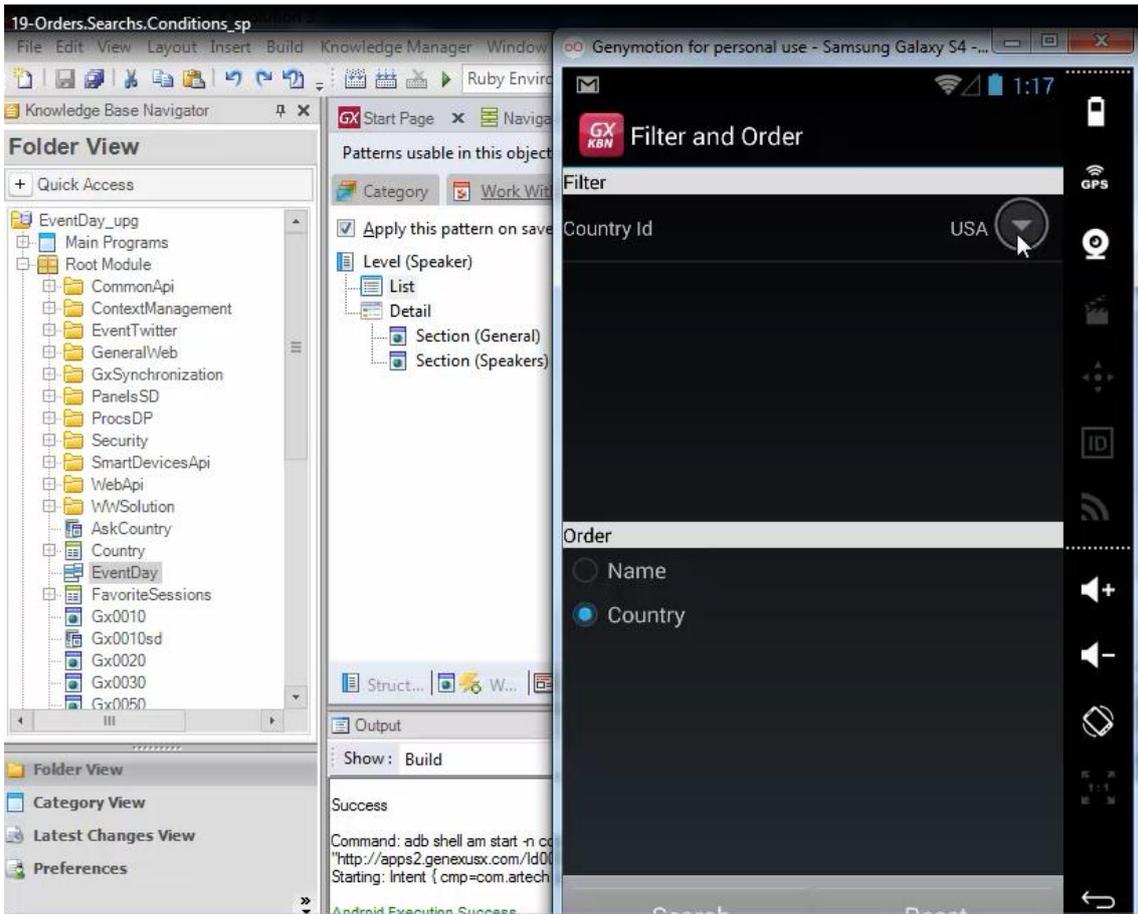
y filtrar la información por ese país



A la vez, podemos hacer búsquedas

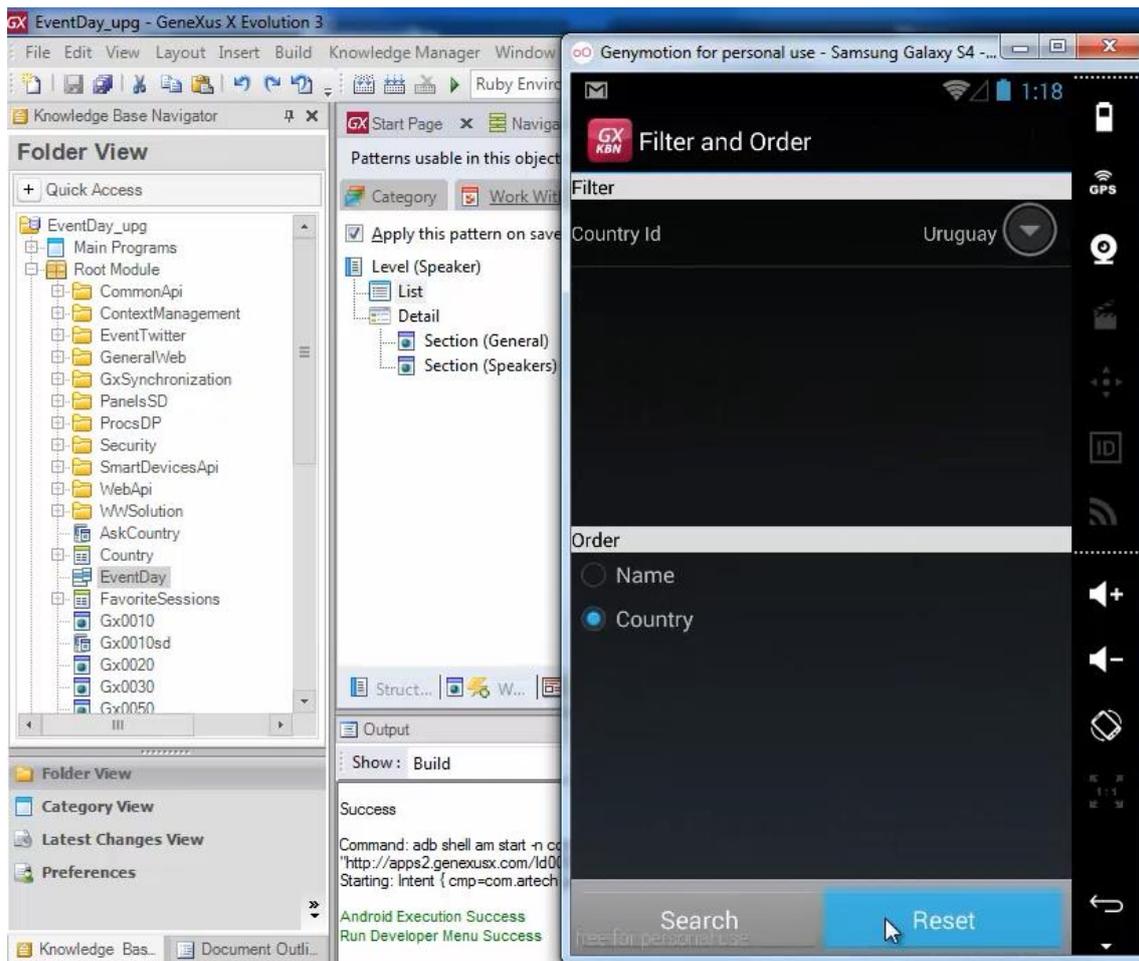




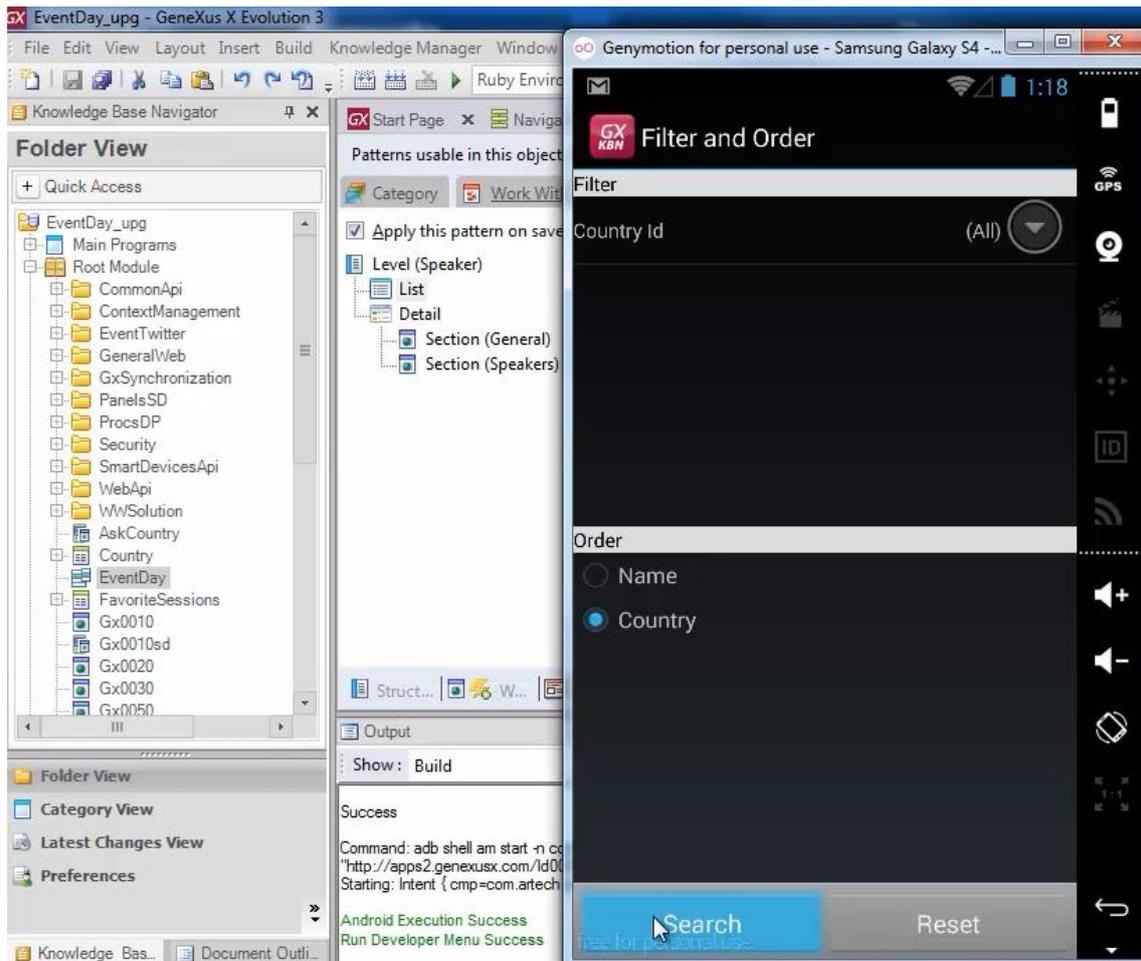


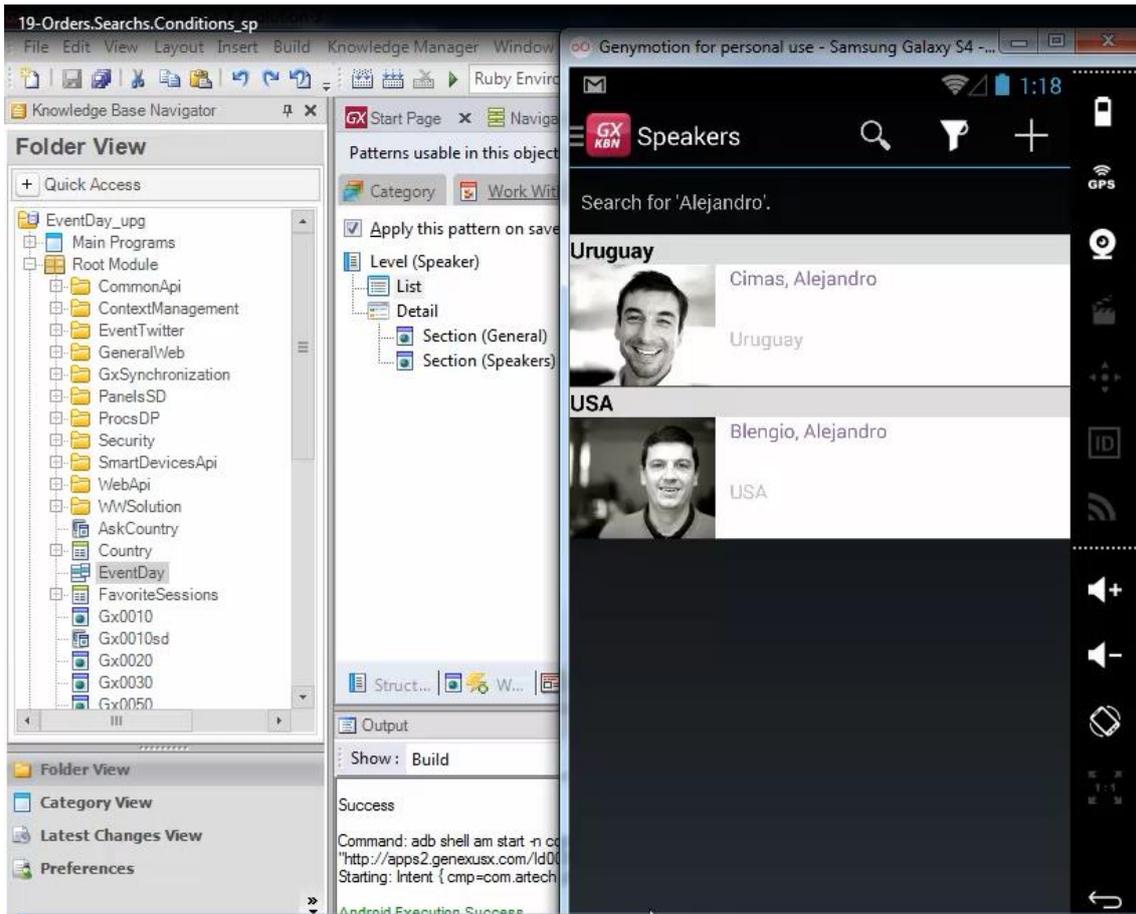
Video filmado con GeneXus X Evolution 3

que se suman a los filtros que tenemos especificados

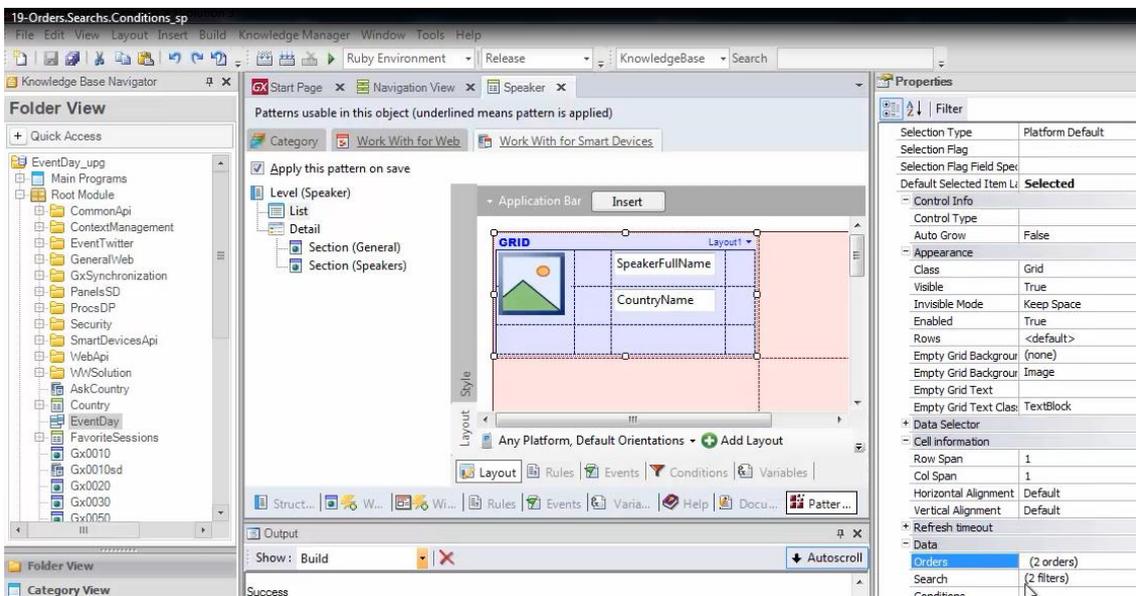


Reseteamos..



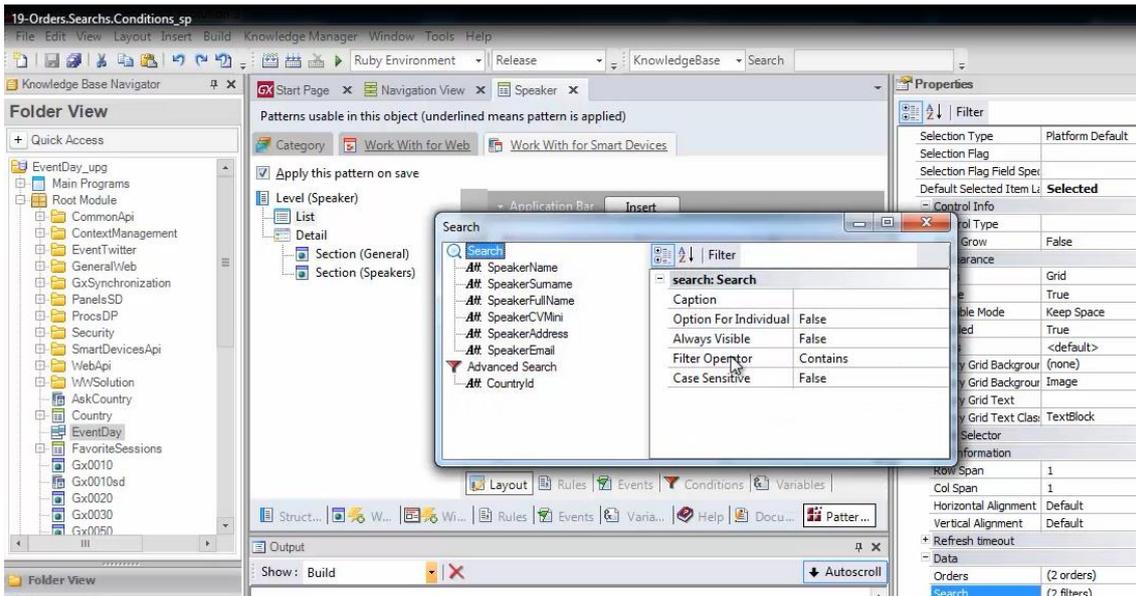


¿Dónde se implementa todo esto en GeneXus?

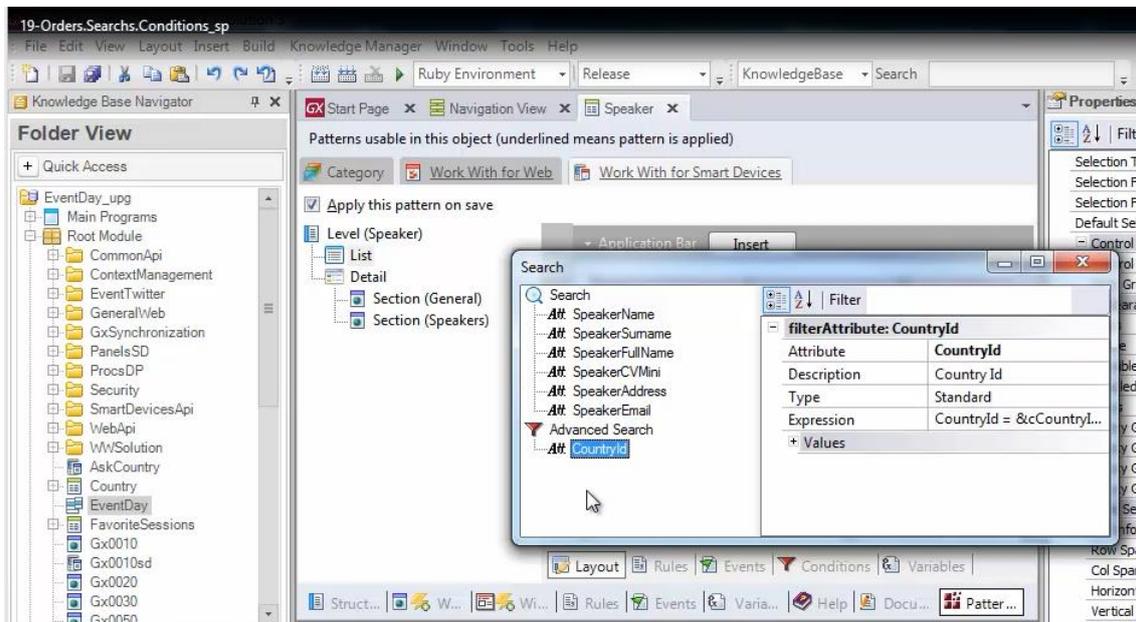


Evidentemente será a nivel de las propiedades del grid, bajo el grupo Data, las de nombre: **Search**

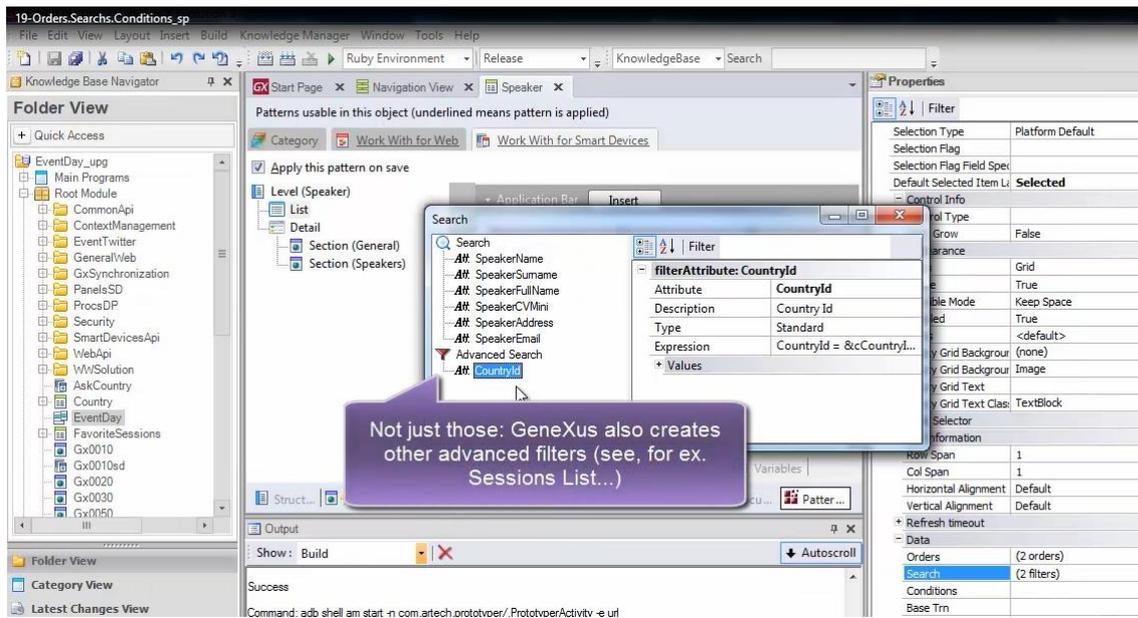
Si abrimos la ventana



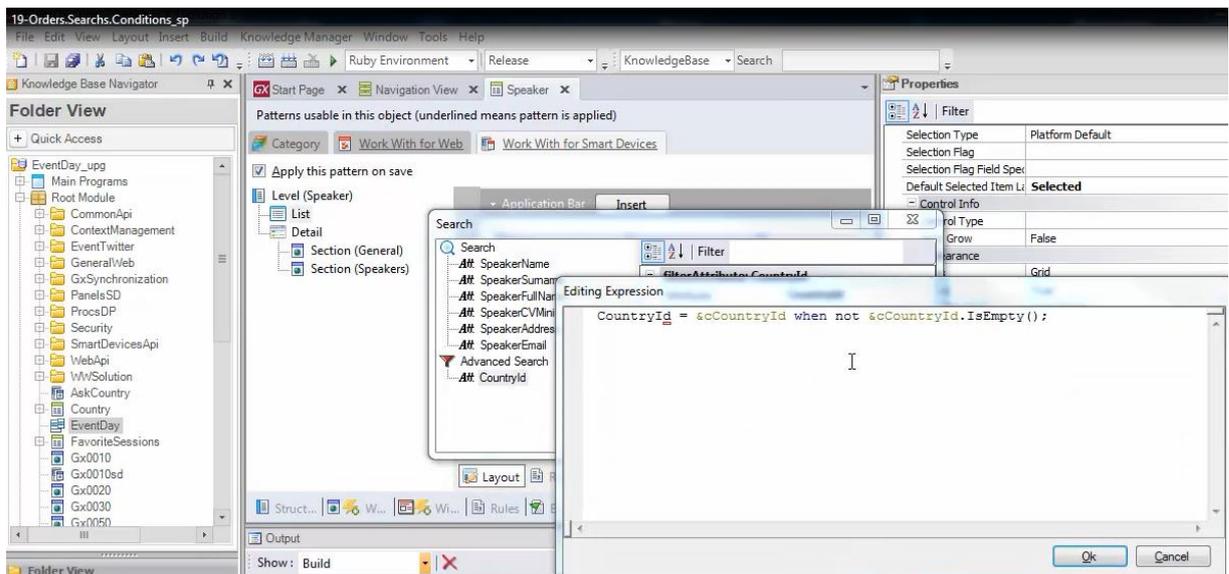
vemos que bajo **Advanced Search**



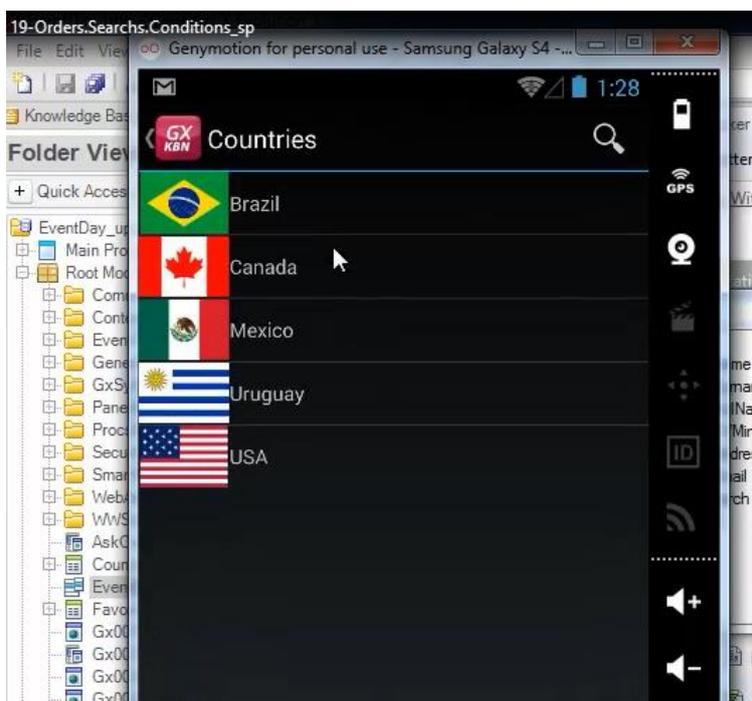
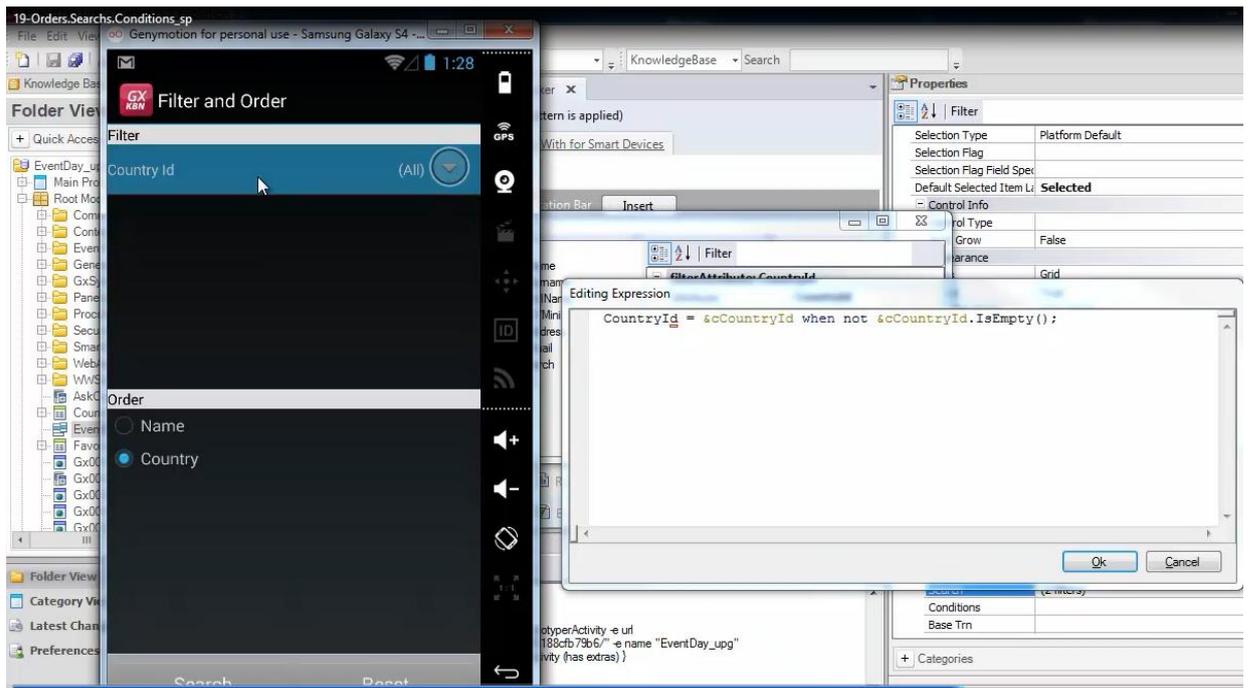
es donde se encuentran las condiciones que automáticamente crea el pattern, por clave foránea, en este caso: CountryId



Si observamos la expresión



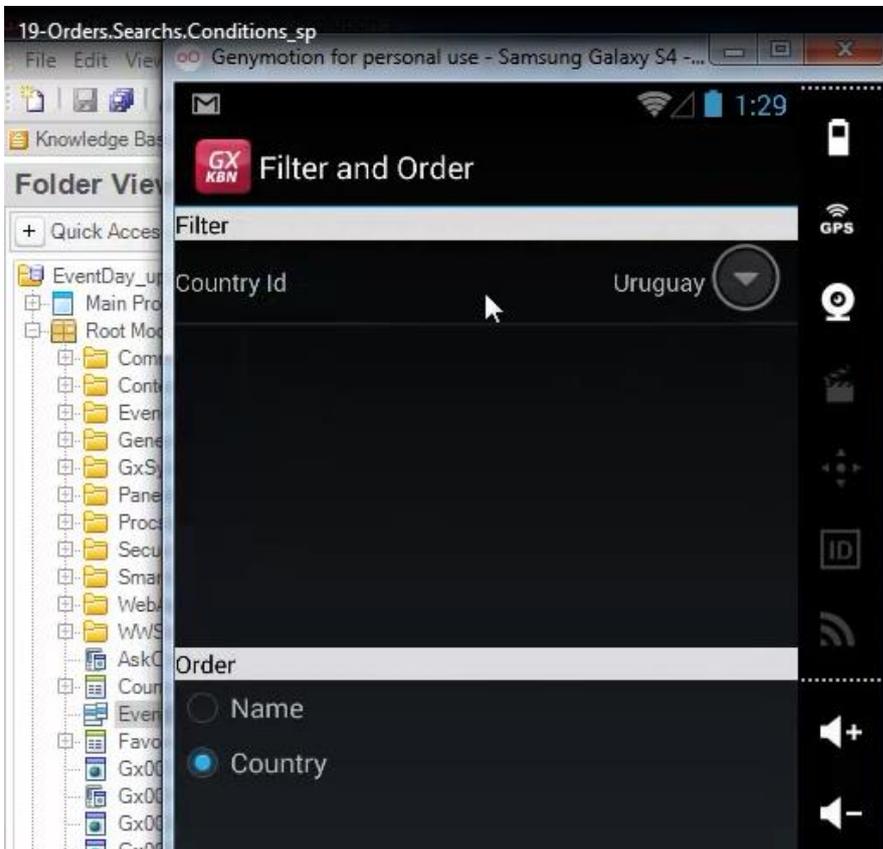
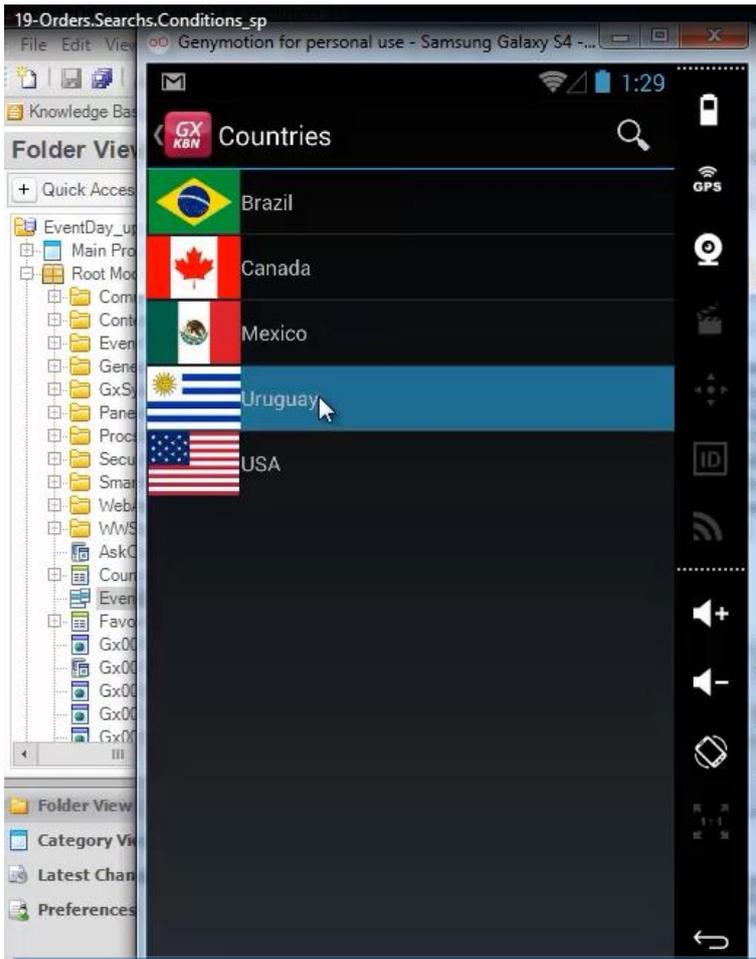
se va a filtrar la información a filtrar en el grid, por el atributo CountryId de acuerdo al valor de una variable. Esta variable es una variable interna que se carga cuando el usuario elije..



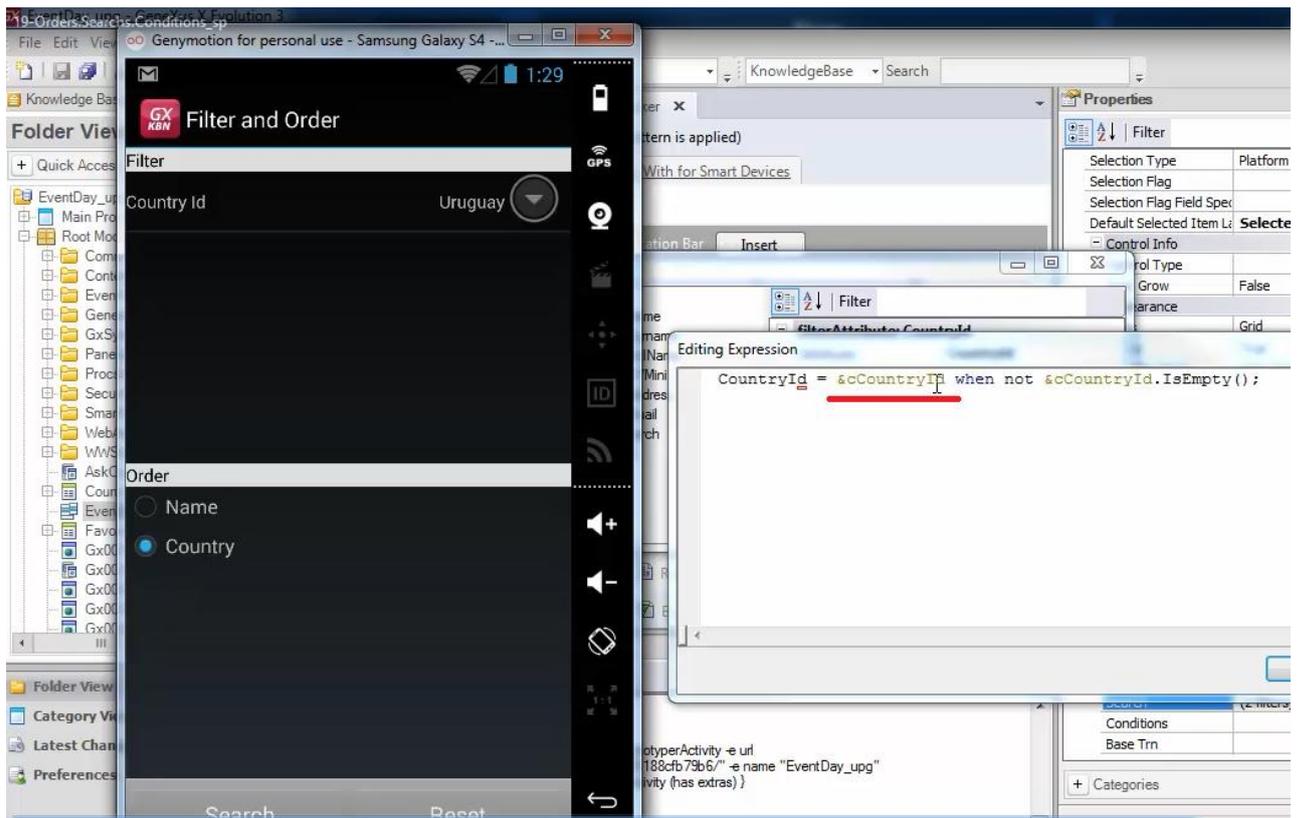
un país de aquí

Este es ni más ni menos que el List de Countries, que está funcionando como un prompt.

Elijo el país Uruguay

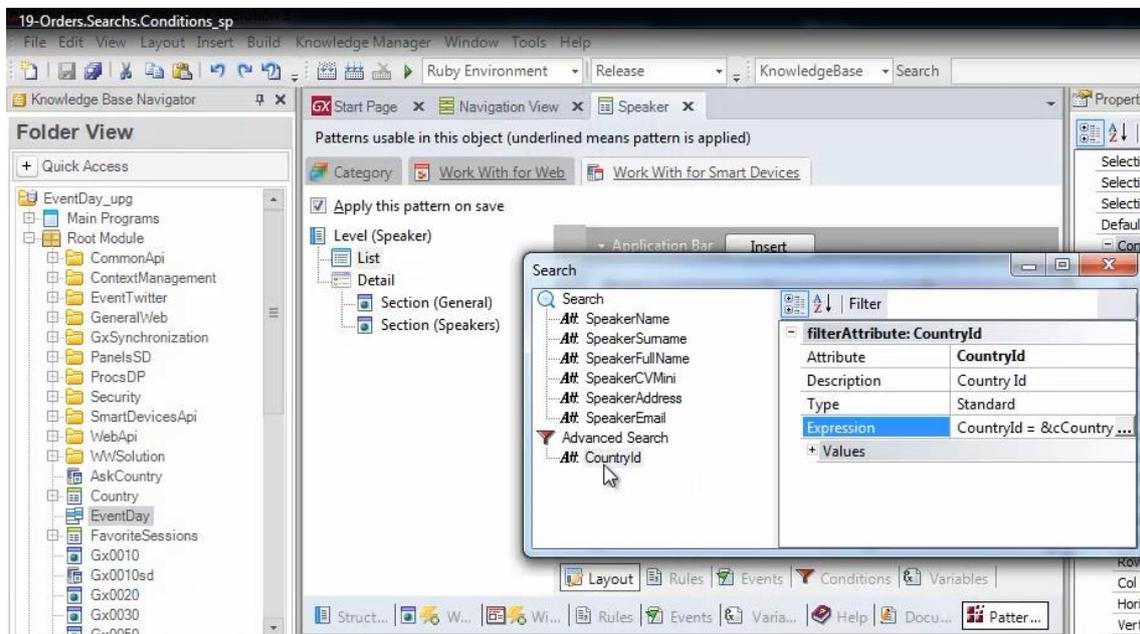


e internamente se va a cargar en esa variable

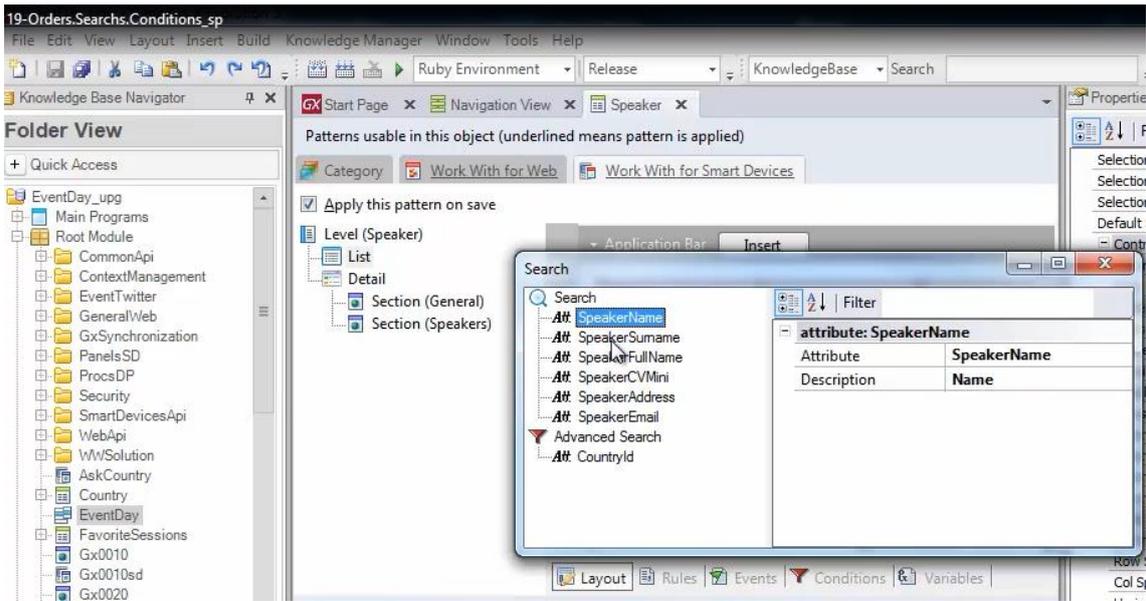


el valor correspondiente al CountryId elegido

Por supuesto, puedo agregar nuevos filtros avanzados o eliminar este que ya existe:

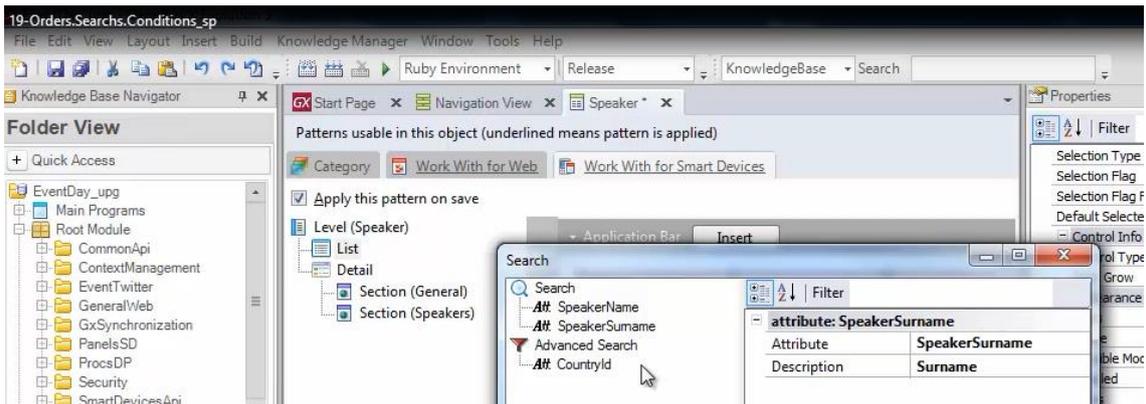


Por otro lado vemos que automáticamente GeneXus creó bajo el nodo Search, todos estos criterios de búsqueda

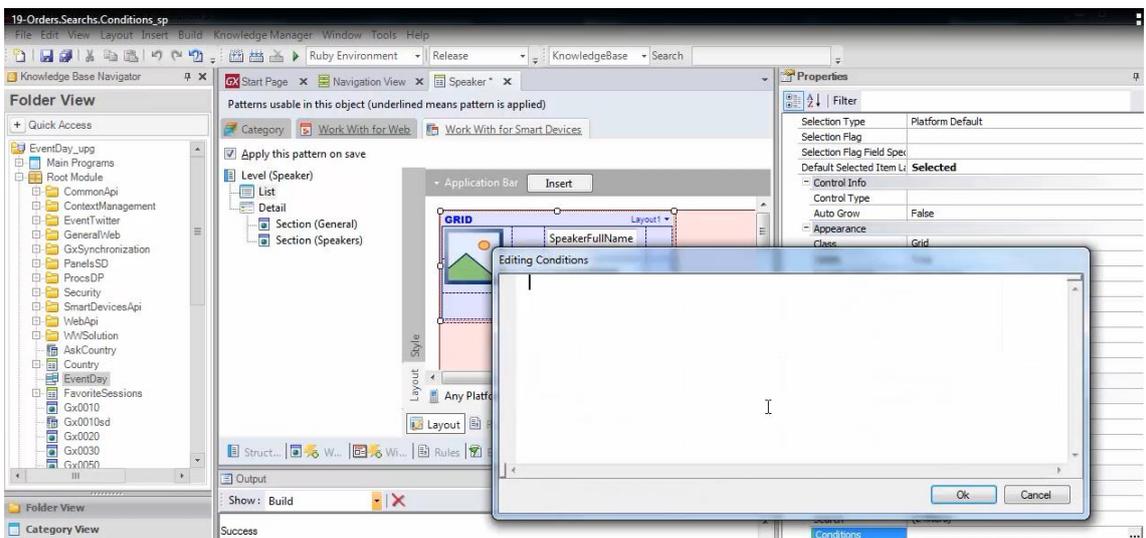


Por name, surname, fullname, curriculum vitae mini, etc.

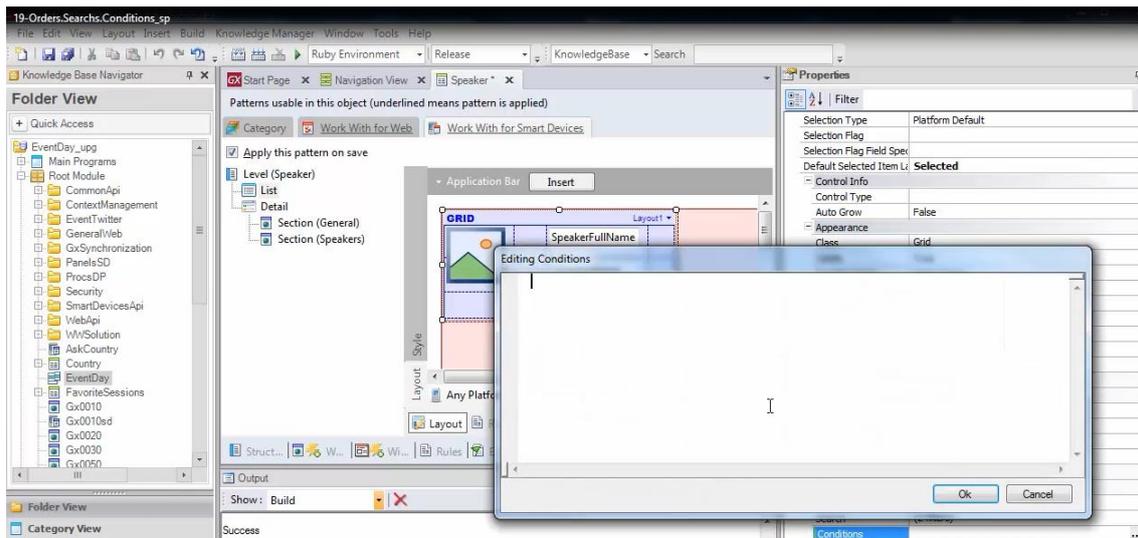
Podemos por supuesto eliminar los que no nos interesen y dejar sólo aquellos que sí nos interesen



Por otro lado y al igual que para los grids de web panels

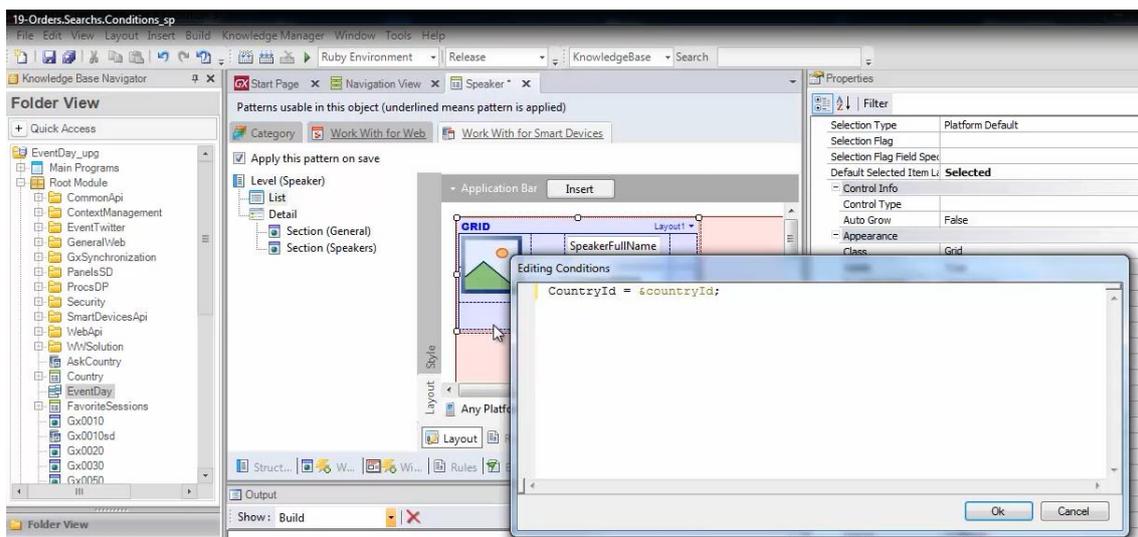


podemos especificar condiciones generales de un grid



es decir, condiciones más flexibles que no tengan por qué depender de la interfaz.

Por ejemplo, podemos querer filtrar los oradores de acuerdo al identificador de país CountryId cargado en una variable en el evento start



Podemos imaginar que esa variable la obtuvimos de una api, que nos informa sobre el país del dispositivo que está ejecutando la aplicación.

Asimismo, como ocurre con los web panels, tenemos una sección para establecer condiciones generales que van a aplicar tanto a la parte fija como al grid o los grids que existan.

Orders & Searchs & Conditions

Invocations

Events

Caching

Con esto, hemos visto cómo ordenar la información de un grid y cómo especificar criterios de búsqueda, filtros y condiciones.

Pasemos ahora a estudiar uno de los temas más importantes: **los eventos**

