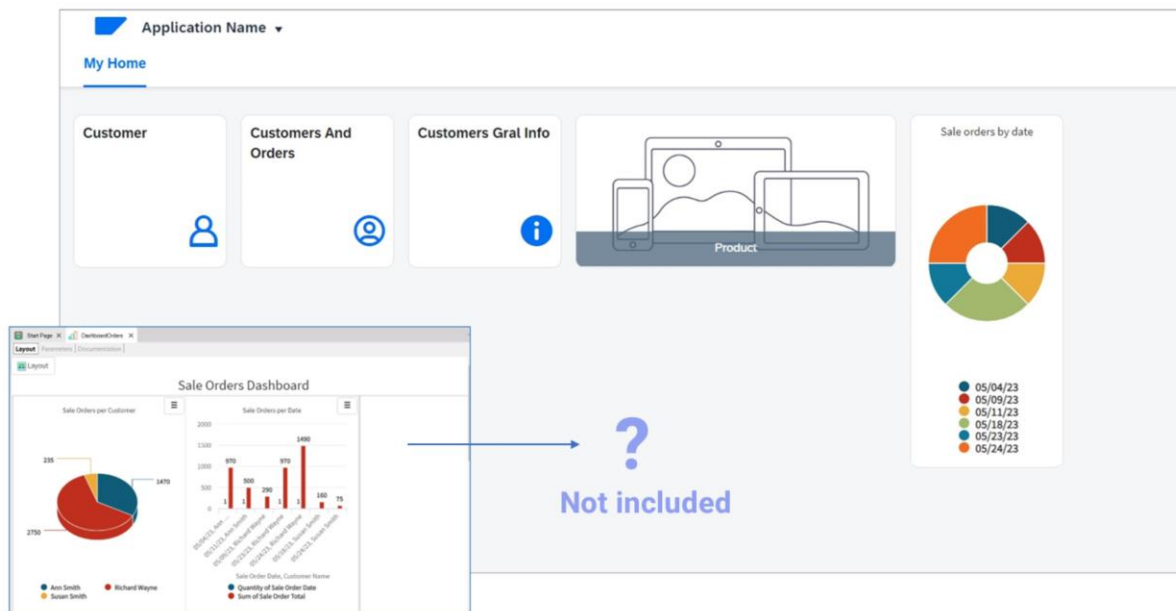


## Uso de patrones: Fiori Launchpad

Hasta el momento hemos visto que el Fiori launchpad incluye solamente tarjetas asociadas a los objetos a los cuales les hemos aplicado alguna instancia del pattern.

# Fiori Launchpad



Pero ¿qué pasa cuando ya tenemos un objeto en nuestra base de conocimiento que no tiene aplicado ningún florrplan? Lo que va a pasar es que no se incluya ninguna tarjeta en el Fiori launchpad.

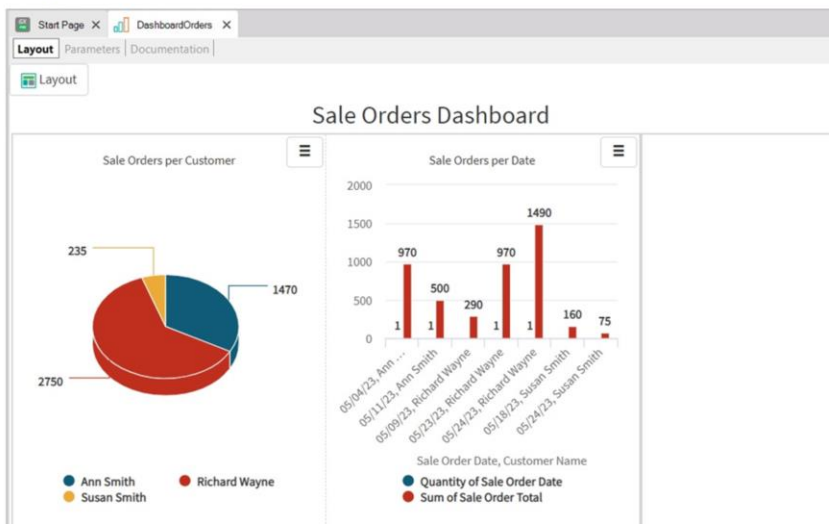
Pero como este launchpad será el objeto principal de la aplicación web, seguramente requerirá tarjetas para invocar a estos otros objetos, creados por el desarrollador.

Vayamos a GeneXus. Tenemos el Web panel Statistics que muestra un dashboard con diversos indicadores con información de interés. En términos generales, un dashboard es un objeto GeneXus que permite integrar varios objetos Query en forma independiente o que comparten filtros en común

Si ejecutamos nuestra aplicación, vemos que este objeto no se visualiza como tarjeta en el Fiori launchpad

La pregunta entonces es la siguiente: ¿Tenemos alguna opción para incluirlo también? Y la respuesta es SI.

## Dashboard object



### More about Dashboard object:

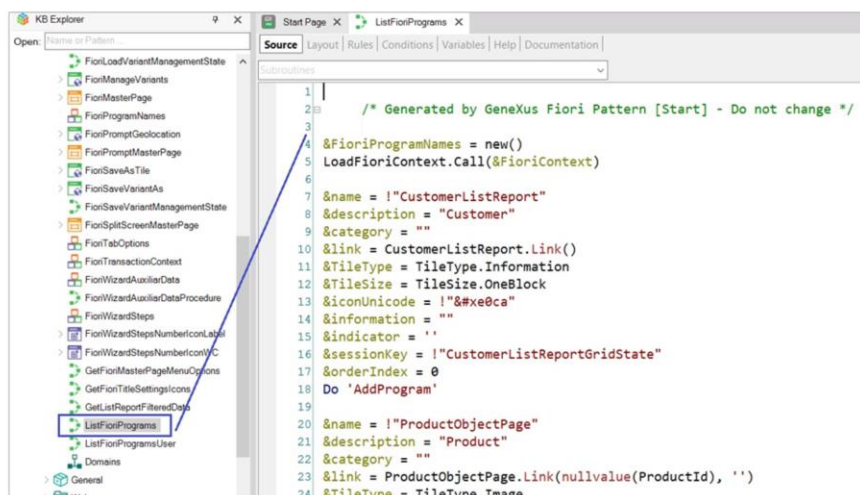
<https://wiki.genexus.com/commwiki/servlet/wiki?36769,Category%3ADashboard+object>

Vayamos a GeneXus. Tenemos el Web panel Statistics que muestra un dashboard con diversos indicadores con información de interés. En términos generales, un dashboard es un objeto GeneXus que permite integrar varios objetos Query en forma independiente o que comparten filtros en común

Si ejecutamos nuestra aplicación, vemos que este objeto no se visualiza como tarjeta en el Fiori launchpad

La pregunta entonces es la siguiente: ¿Tenemos alguna opción para incluirlo también? Y la respuesta es SI.

## Procedure ListFioriPrograms



Veamos esto.

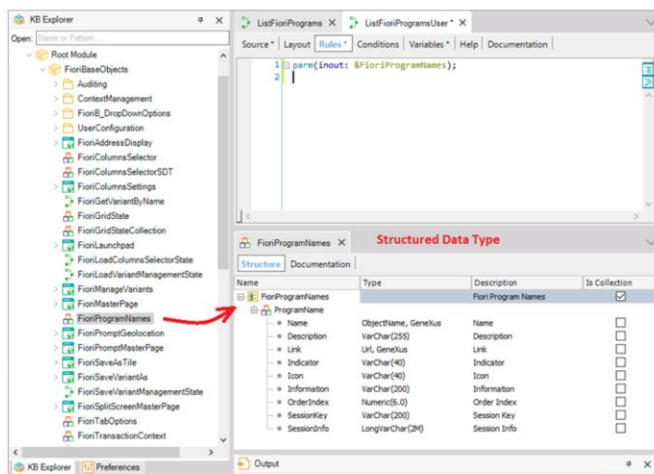
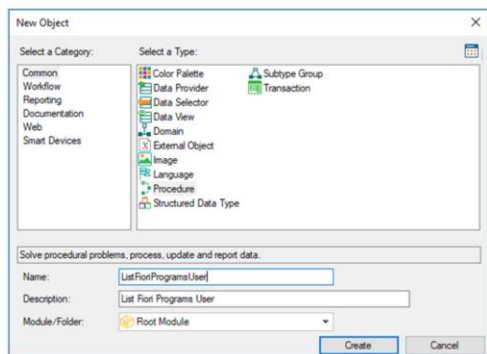
En el módulo FioriBaseObjects existe un procedimiento de nombre ListFioriPrograms que es el responsable de obtener la colección de programas a ser incluidos en el launchpad. Cuando se aplica el patrón Fiori for Web a un objeto, este procedimiento se modifica para incluir una tarjeta con el nuevo patrón.

Cada uno de estos bloques corresponde a la información de una tarjeta, y como vemos, es código generado automáticamente por GeneXus a partir de la aplicación del pattern Fiori. Y no debemos modificarlo.

Una opción, entonces, es incluir la información necesaria para genera la nueva tarjeta, fuera de las marchas del pattern. Aquí

Otra opción, más prolija, es crear un objeto de tipo procedimiento que reciba por parámetro la lista de programas creada por el ListFioriPrograms, y la manipule de acuerdo a las tarjetas que deban agregarse al launchpad.

## New procedure: ListFioriProgramsUser



Vamos a agregar una tarjeta para invocar al Web panel Statistics. Para eso creamos un procedimiento de nombre ListFioriProgramsUser:

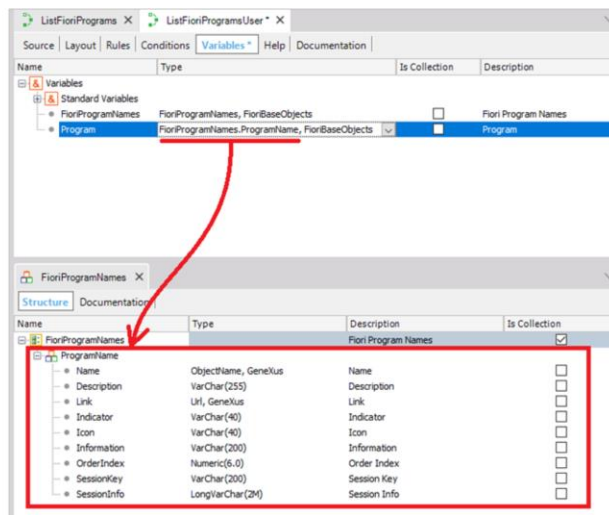
Este procedimiento debe recibir y devolver la colección de programas a ser invocados por el launchpad.

Observemos entonces, que en el módulo FioriBaseObjects existe un tipo de dato estructurado predefinido de nombre FioriProgramNames, que es precisamente el tipo de dato que permite registrar la colección de programas a ser invocados.

Si lo abrimos, se trata de una colección de un tipo de datos estructurado simple, que corresponde a la información de una tarjeta.

Bien. En nuestro procedimiento definimos entonces la regla Parm que permite especificar este parámetro que necesitamos de entrada/salida. Lo haremos a través de una variable basada en el tipo de datos estructurado FioriProgramNames. Recordemos que al definir una variable con el mismo nombre que el tipo de datos, automáticamente asumirá dicho tipo de datos.

## New procedure: ListFioriProgramsUser



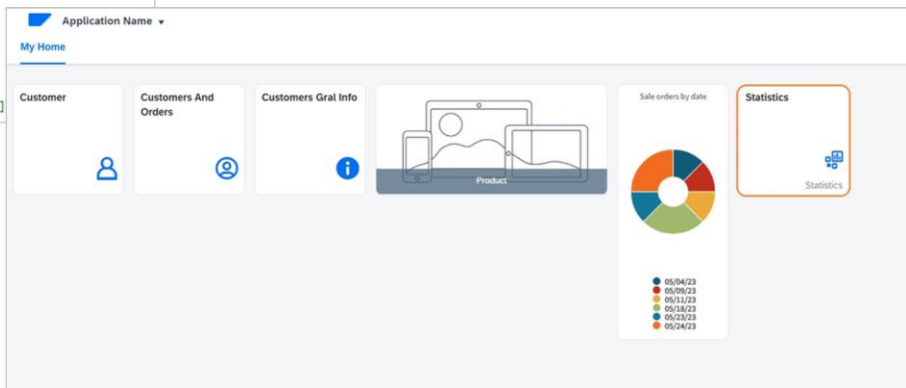
Definimos también una variable interna al procedimiento basada en este tipo de datos.. que corresponde a un elemento de la colección, o sea, a una tarjeta.

Vamos ahora a la solapa Source para cargar la información necesaria para incluir la tarjeta asociada a nuestro Web Panel Statistics.

Luego nos falta agregar esa "tarjeta" a la colección de FioriProgramNames:

# Fiori Launchpad

```
Start Page X | ListFioriPrograms X
Source | Layout | Rules | Conditions | Variables | Help | Documentation
Subroutines
64 | &sessionKey = ''
65 | &borderIndex = 0
66 | Do 'AddProgram'
67 |
68 | /* Generated by GeneXus Fiori Pattern [End] - Do not change */
69 |
70 | &Index = 1
71 | For &ProgramName in FioriGetUserTiles()
72 |   &FioriProgramNames.Add(&ProgramName, &Index)
73 |   &Index += 1
74 | EndFor
75 |
76 | ListFioriProgramsUser(&FioriProgramNames)
77 |
78 | Sub 'AddProgram'
79 |
80 | /* Generated by GeneXus Fiori Pattern [Start]
```



Por último, lo que debemos hacer es una invocación a este procedimiento desde el procedimiento predefinido ListFioriPrograms que, como vimos recientemente es quien carga las tarjetas en el launchpad.

Hacemos aquí la invocación, pasándole esta variable &FioriProgramNames como parámetro a nuestro nuevo procedimiento, para que agregue la tarjeta que queremos:

Presionemos F5: Ahora sí, vemos que la tarjeta Statistics fue incluida en el Fiori launchpad, y que está invocando, efectivamente, a nuestro Web Panel con el dashboard.

**GeneXus**<sup>™</sup>  
by **Globant**

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