



# CREAR UN DASHBOARD CON PENTAHO BI-SERVER

Dashboard Pentaho con CDE

Jortilles.com  
info@jortilles.com

## Índice de contenido

1.Introducción.....	3
2.Requerimientos.....	3
3.Caso de uso: Creación de un Dashboard.....	4

## 1. Introducción

Un Dashboard o cuadro de mando, es una aplicación que se utiliza para presentar el contenido de una serie de indicadores que muestran el comportamiento de los datos.

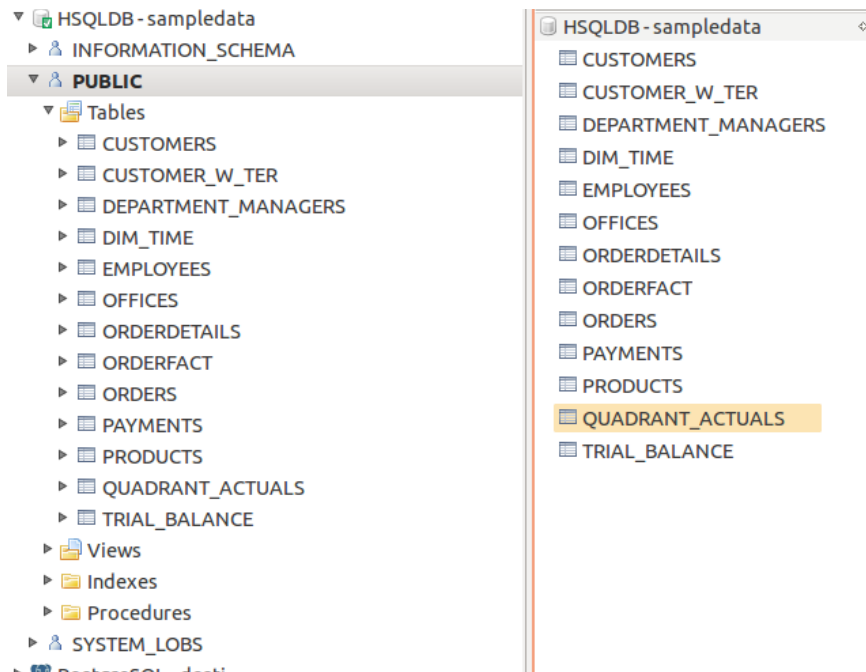
En este tutorial realizaremos un Dashboard con Pentaho, utilizando la herramienta que nos da Pentaho para hacer cuadros de mando, es decir CDE. <http://www.webdetails.pt/ctools/cde/>

## 2. Requerimientos

Para realizar este Dashboard necesitamos instalar Pentaho Bi Server , disponible para descargar desde la web de SourceForge

<https://sourceforge.net/projects/pentaho/files/Business%20Intelligence%20Server/6.0/>

La base de datos que utilizaremos serán la que viene de prueba, "SampleData", al configurar Pentaho BI Server.



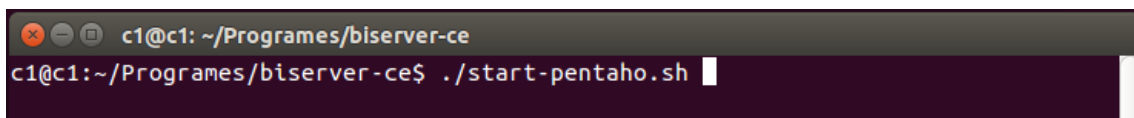
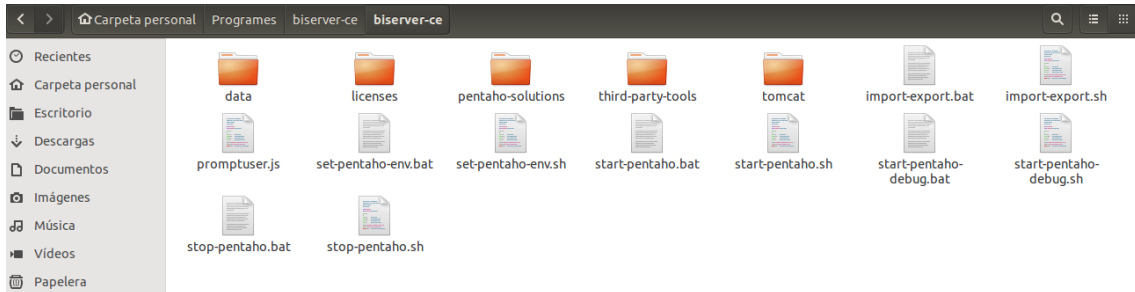
Estas son las tablas que tenemos disponibles para realizar nuestro Dashboard.

En nuestro Dashboard, la información que queremos presentar contendrá será:

- Un selector con los nombres de los managers
- Un gráfico de barras que mostrará la varianza acumulada por cada departamento.

### 3. Caso de uso: Creación de un Dashboard

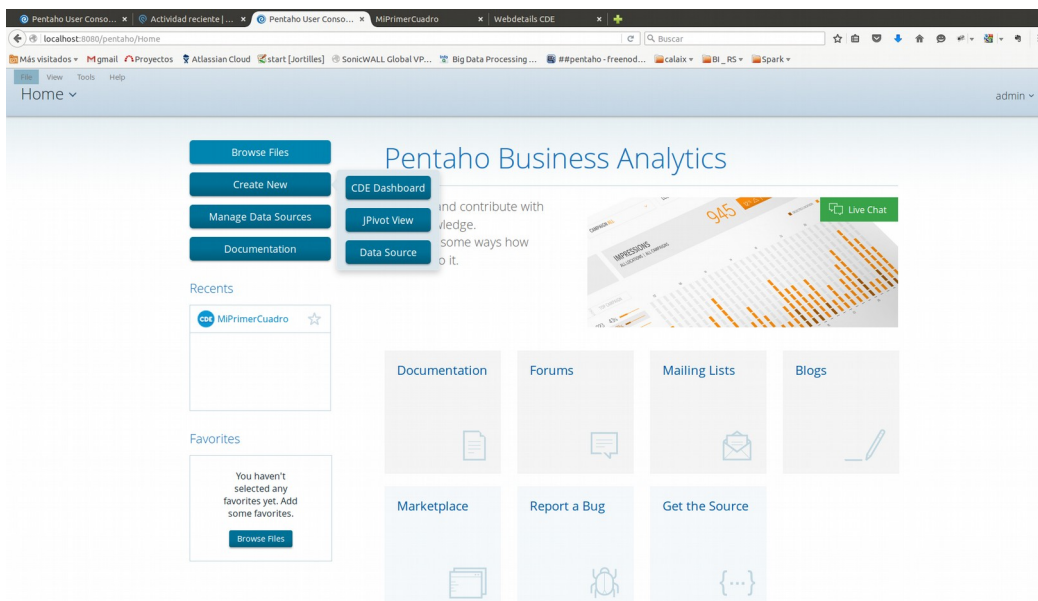
Una vez descargado el programa, descomprimos, y ejecutamos en el terminal `./start-pentaho.sh`



Abrimos el navegador con la ruta `localhost:8080`, y accedemos con el usuario admin



Seleccionamos Create New → CDE Dashboard



En el Dashboard tenemos tres vistas



- **Layout Panel**, es decir la presentación gráfica de los datos
- **Components Panel**, los objetos que mostraran los datos, como gráficos, tablas ...
- **DataSource Panel**, son las consultas a los orígenes de datos

Primero guardaremos el documento con el nombre deseado, en este caso elegiremos como base el framework de Bootstrap:

**Settings** Save Cancel

Title  
@DASHBOARD\_TITLE@

Author  
Insert Text...

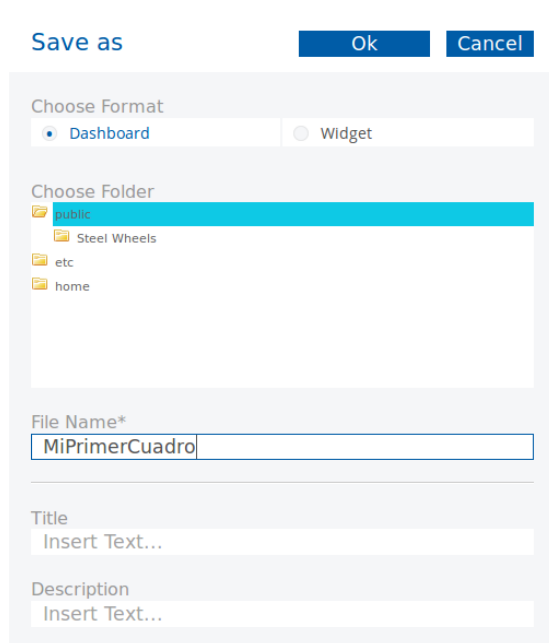
Description  
@DASHBOARD\_DESCRIPTION@

Style  
Select...

Dashboard Type  
bootstrap

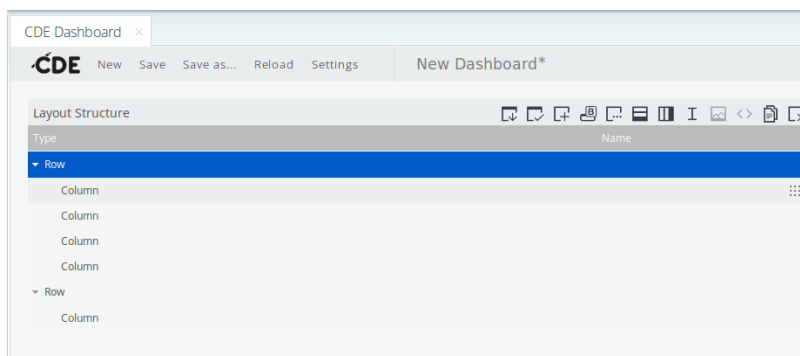
RequireJS Support

Lo guardamos en la carpeta publica, con el nombre elegido y continuamos:



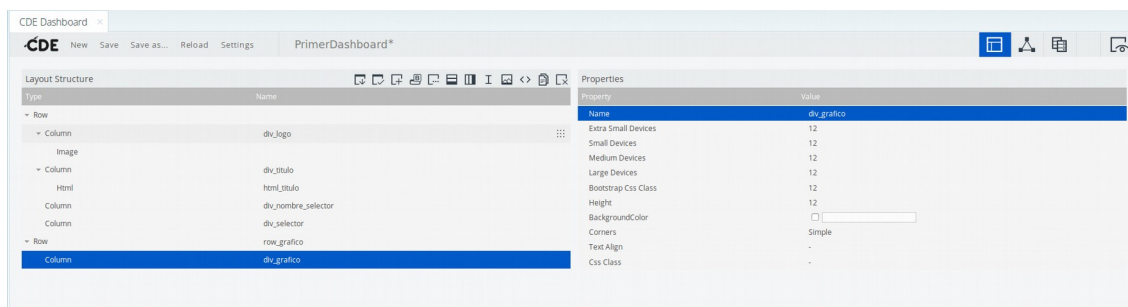
### ➤ Layout Panel

Empezamos a crear la estructura física donde irán los objetos. En esta capa podemos añadir los estilos css, código javascript y html

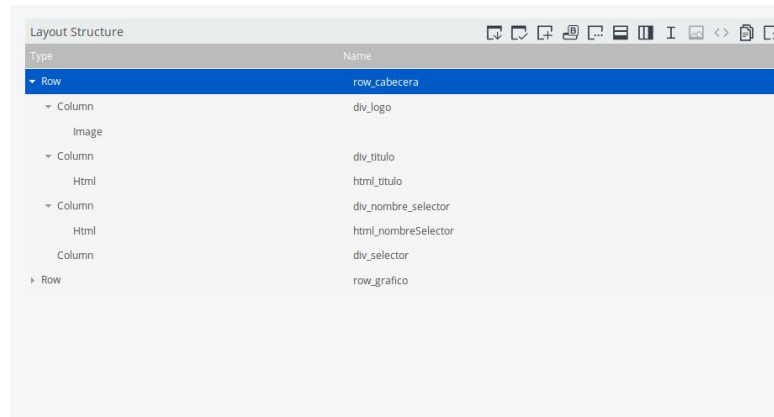


Como estamos trabajando con Bootstrap, se basa en modelo de cajas (filas y columnas) la suma total de las columnas tiene que ser 12.

Definimos las filas que tendrá nuestro diseño, y dentro de cada fila las columnas que necesitaremos

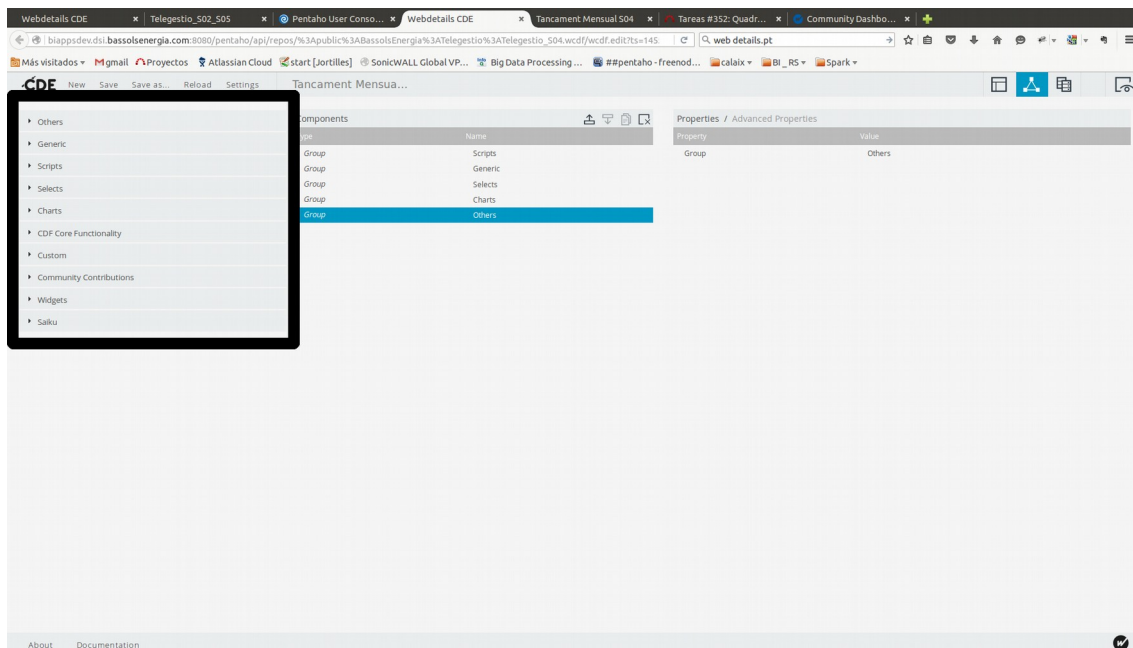


La Primera fila corresponde a la cabecera (con todos sus elementos) , y la segunda corresponde al gráfico.



### ➤ Component Panel

Aquí tenemos diferentes objetos para utilizar, organizados por categoría (gráficos, selectores..):



Una vez definida la estructura de la presentación, vamos a crear los objetos necesarios. Necesitamos:

- Un selector, es decir un desplegable que muestre el nombre de los **managers** para hacer la consulta.
- Gráfico de barras para mostrar el resultado

Para crear el selector, primero debemos crear una variable (*simple parametre*) donde guarde el valor asignado:

The screenshot shows the 'PrimerDashboard\*' interface. On the left, the 'Components' panel lists a 'Simple parameter' component named 'param\_manager'. On the right, the 'Properties / Advanced Properties' panel shows the following configuration:

Property	Value
Name	param_manager
Property value	-
Bookmarkable	False
Public	False

Después creamos el selector, (*select component*) es decir el desplegable, que devolverá el valor del parámetro que hemos creado anteriormente:

The screenshot shows the 'PrimerDashboard\*' interface. On the left, the 'Components' panel lists a 'Select Component' named 'select\_manager'. On the right, the 'Properties / Advanced Properties' panel shows the following configuration:

Property	Value
Name	select_manager
Parameter	param_manager
Listeners	Select options   OK
Parameters	[]
jQuery Plugin	-
Value as id	True
Datasource	SQL_SELECTOR
Values array	[]
HtmlObject	div_selector

**DataSource:** Ponemos este valor, que sera el nombre de la consulta que crearemos a continuación: SQL\_SELECTOR

**HtmlObject:** Es el nombre del objeto que hemos declarado en el layout, donde ira, en este caso el selector DIV\_SELECTOR

Y el último objeto que necesitamos es el gráfico de barras:

The screenshot shows the 'PrimerDashboard\*' interface. On the left, the 'Components' panel lists a 'CCC Bar Chart' named 'chart\_grafico'. On the right, the 'Properties / Advanced Properties' panel shows the following configuration:

Property	Value
Name	chart_grafico
Title	-
Listeners	[]
Parameters	[]
Datasource	SQL_GRAFICO
Height	-
Width	-
HtmlObject	div_grafico
clickable	False
clickAction	...
compatVersion	2
crosstabMode	True
legend	True
seriesInRows	False
timeSeries	False
timeSeriesFormat	%Y-%m-%d

**DataSource:** Ponemos este valor, que sera el nombre de la consulta que crearemos a continuación: SQL\_GRAFICO

**HtmlObject:** Es el nombre del objeto que hemos declarado en el layout, donde ira, en este caso el selector DIV\_GRAFICO



### ➤ Datasource Panel

- Para realizar la consulta SQL\_SELECTOR. Seleccionaremos “ sql over sqlJndi”

Type	Name
Group	SQL Queries
sql over sqlJndi	SQL_SELECTOR

Property	Value
Name	SQL_SELECTOR
Access Level	Public
Jndi	SampleData
Query	Select manager_name (...)
Parameters	[]
Calculated Columns	[]
Columns	[]
Output Options	[]
Output Mode	Include
Cache Keys	[]
Cache Duration	3600
Cache	True

Esta consulta devolverá el nombre de los managers

Sql Editor

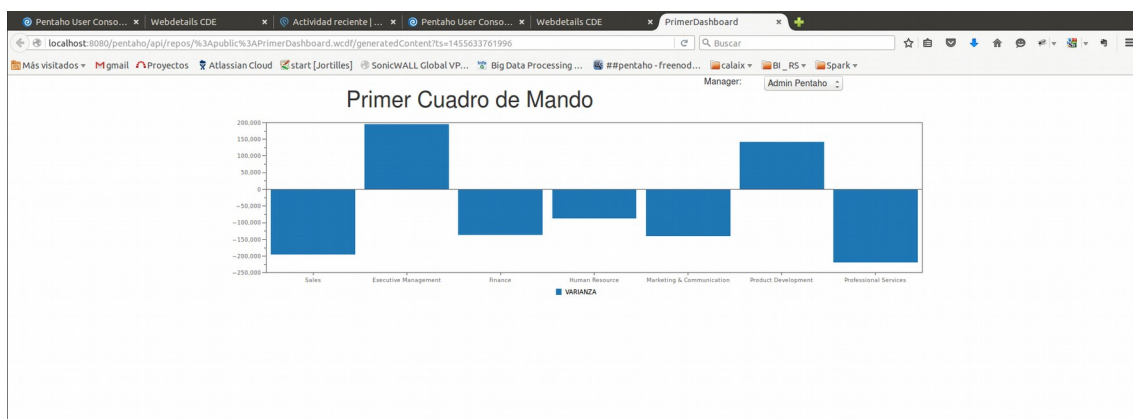
1 Select manager\_name from DEPARTMENT MANAGERS;

- Para realizar la consulta SQL\_GRAFICO

Sql Editor

1 select  
2 department,  
3 sum(variance) as varianza  
4 from quadrant\_actuals  
5 inner join DEPARTMENT MANAGERS  
6 on quadrant\_actuals.REGION = DEPARTMENT MANAGERS.REGION  
7 group by department;

Si vamos al navegador y actualizamos , ya podemos ver el resultado, con el selector y el gráfico. Aunque aún no se actualizan los datos al cambiar el valor del selector



Para actualizar los datos al cambiar la selección, debemos activar los “listeners”, y los parámetros.

Un *listener* esta relacionado con un parámetro, cuando se produce un cambio de este se desencadena una actualización del componente, en este caso, cuando se produce un cambio de manager, se actualiza el gráfico.

*Parameters*, es el valor o los valores que se pasan como argumentos para ejecutar la acción. En este caso solo pasamos un parámetro, el manager.

Volvemos a la pestaña de Componentes, y en el gráfico, ponemos en ambos campos “param\_manager”

The screenshot shows the 'Components' panel on the left with 'CCC Bar Chart' selected. The 'Properties / Advanced Properties' panel on the right shows the 'Listeners' property set to 'param\_manager'. The 'Parameters' section below it has 'param\_manager' selected.

Ahora debemos actualizar el “datasource” SQL\_GRAFICO, con el valor del parámetro:

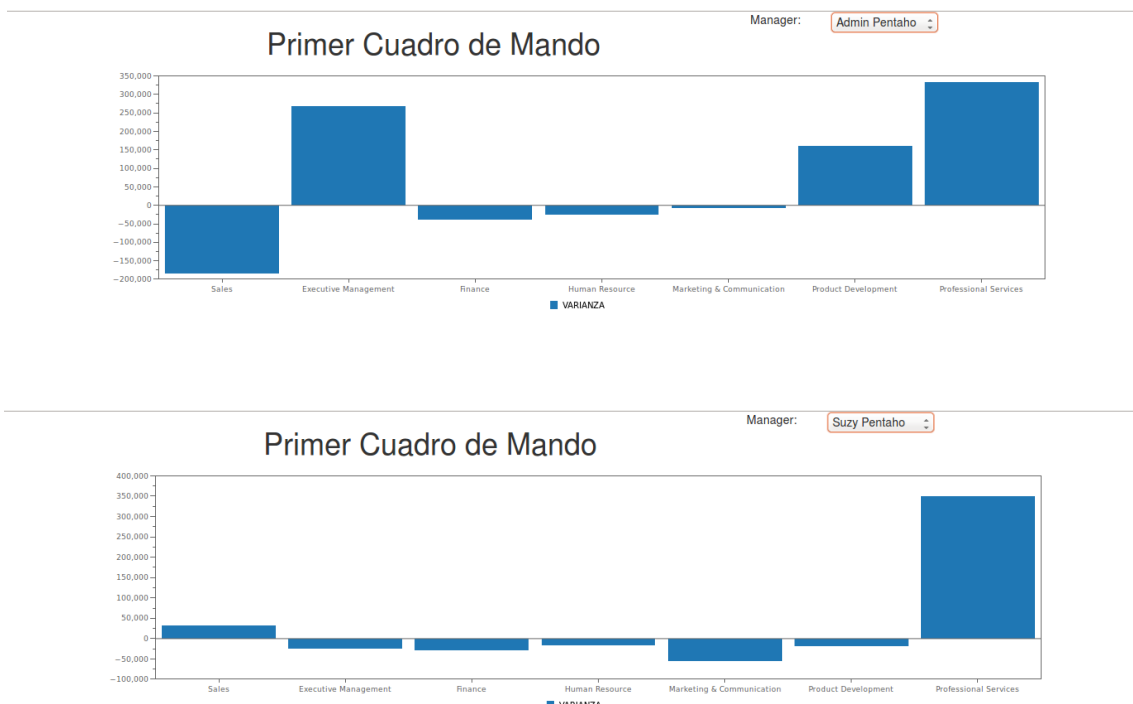
The SQL Editor shows a query with the following code:

```

1 select
2 department,
3 sum(variance) as varianza
4 from quadrant_actuals inner join DEPARTMENT MANAGERS
5 on quadrant_actuals.REGION = DEPARTMENT MANAGERS.REGION
6 where manager_name = ${param_manager}
7 group by department

```

Ahora si que cambia la tabla en función del nombre del manager:



Ahora vamos a añadir un **panel de bootstrap**, para mejorar la presentación del gráfico de barras. Para añadirlo simplemente clicamos en el icono

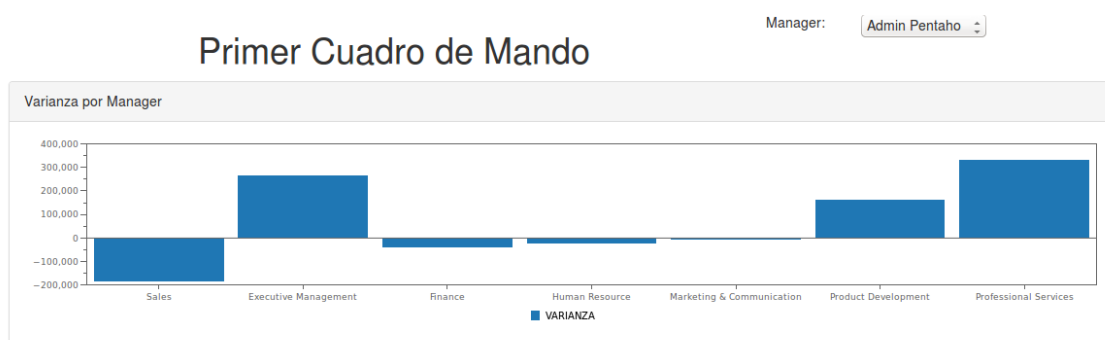
Type	Name
Row	row_cabecera
Column	div_logo
Column	div_titulo
Column	div_nombre_selector
Column	div_selector
Row	row_grafico
Column	div_grafico
Bootstrap Panel	
Panel Header	
Html	html_titulo_grafico
Panel Body	div_contenedor

En el Panel Header, declararemos el título del gráfico.

Y en el panel body irá el gráfico de barras. Por ello debemos actualizar el nombre del objeto html que estaba asignando al gráfico.

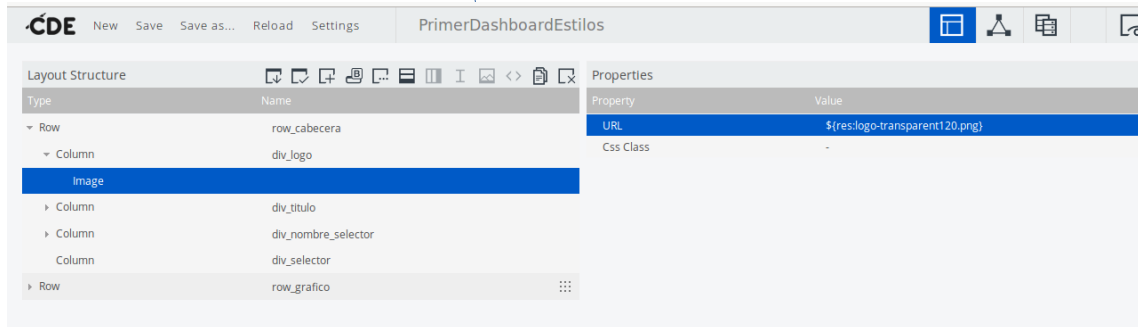
Property	Value
Name	chart_grafico
Title	-
Listeners	["param_manager"]
Parameters	[["param_manager","p (...)
Datasource	SQL_GRAFICO
Height	200
Width	-
HtmlObject	div_contenedor
clickable	False
clickAction	...
compatVersion	2
crosstabMode	True
legend	True
seriesInRows	False
timeSeries	False
timeSeriesFormat	%Y-%m-%d

Si volvemos a actualizar el navegador ya podemos ver el panel con el título en el gráfico.



Por último, vamos a añadir nuestro logo en el Dashboard:

Ya teníamos definido el espacio en la capa del **layout**, en la fila correspondiente a la cabecera, insertamos una imagen



Y en las propiedades, debemos introducir la url donde hemos puesto la imagen. En nuestro caso la hemos subido en la misma carpeta del dashboard, la carpeta publica.

Como url será: `$(res:nombre_logo.png)`

Volvemos a actualizar el navegador, y ahora si, ya tenemos nuestro primer Dashboard finalizado:

