



LECTRA FASHION PLM PRODUCT DEVELOPER CONFIGURACION DE LAS PANTALLAS

Guía de configuración

Fecha de la última actualización: Agosto 2016 abril 2017

Contenido

Convenciones	4#
Excepción	4#
Introducción	4#
1.# Nueva estructura de configuración	5#
1.1# Estructura previa	5#
1.2# Nueva estructura	5#
1.3# Directorio Config	6#
1.4# Directorio I18n	6#
1.5# Directorio lpf	6#
1.6# Directorio lpfext	7#
1.7# Directorio Webtool	7#
1.8# Operación	7#
2.# Definiciones y conceptos	8#
2.1# Archivos XML	8#
2.2# Herencia	11#
2.3# Lo que hay que saber sobre la estructura de descripción de las pantallas	14#
3.# Configuración añadiendo campos a "General Objectives"	16#
3.1# Agregar campos simples	16#
3.2# Añadir una lista	21#
3.3# Añadir la clasificación	23#
3.4# Añadir un panel de "Properties (Propiedades)"	26#
4.# Configuración añadiendo campos en el explorador de productos	29#
4.1# Requisitos del módulo de Administración y configuración	29#
4.2# Presentación de la pantalla Product (Producto)	29#
4.3# Archivo XML personalizable	30#
4.4# Estructura del panel de búsqueda en el archivo XML	30#
4.5# Identificar la parte que se va a duplicar y modificar	31#
4.6# Añadir un criterio a un grupo existente del panel de búsqueda	32#
4.7# Añadir un grupo de criterios en el panel de búsqueda	32#
4.8# Añadir una columna a la cuadrícula de resultados	34#
5.# Configuración añadiendo campos al Plan de colección	38#
5.1# Requisitos del módulo de Administración y configuración	38#
5.2# Presentación de las pantallas del plan de colección	38#
6.# Configuración avanzada para añadir campos y paneles	62#
6.1# Uso de la herencia	62#



6.2#	Ejemplo concreto.....	62#
7.#	Modificación del diseño de página.....	63#
7.1#	Definición del diseño de página	63#
7.2#	La estructura ExtJSErreur ! Source du renvoi introuvable.es la única que tiene un " layout (diseño de página)".....	63#
ANEXOS.....		65#

#



Las modificaciones introducidas en el documento desde la última vez que se publicó aparecen subrayadas en [azul](#).

CONVENCIONES

Módulo de desarrollo del producto = Product Developer

Módulo de administración y configuración = PLM Manager

EXCEPCION

Ciertas pantallas de Lectra Fashion PLM V5R1 no se pueden personalizar:

- Explorador del set de composición
- Forma del set de composición
- Pestaña Composición en Objetivos

INTRODUCCION

Las técnicas utilizadas para la descripción de las pantallas ha evolucionado de la versión V4R1 a la versión V5R1 de Lectra Fashion PLM.

Los archivos de descripción de pantallas siguen estando en formato XML. Sin embargo, han cambiado la estructura de los archivos y el contenido de las pantallas.

Este documento está pensado para ayudarle a identificar los archivos XML que hay que modificar para configurar las pantallas, así como qué partes hay que duplicar y cambiar.

En el primer capítulo se presentan los cambios que ha experimentado la estructura de configuración.

En el segundo capítulo se presentan las definiciones y los conceptos necesarios para leer y entender los archivos XML. Se incluyen algunas definiciones relacionadas con las técnicas de ExtJS.

En la tercera parte, la configuración, se trata la incorporación de campos y bloques a las pantallas **Objetivos generales** para los **Productos**, en el explorador de **Productos** con el nuevo panel de **Búsqueda**, así como las pantallas del **Plan de colección**.

Cuando se haya familiarizado con esta configuración, puede utilizar el concepto de herencia para mejorar la eficacia de la configuración y disminuir la duplicación de los bloques.

En la cuarta parte se explica cómo reorganizar el contenido de la pantalla cambiando el **diseño de página** predeterminado. El **diseño de página** consiste en definir la organización de los datos en una pantalla: por columnas, tablas, etc.

Los anexos contienen recordatorios sobre la creación de datos en el módulo de Administración y configuración, así como la documentación técnica relativa a los elementos gráficos utilizados, con una parte de las estructuras ExtJS y LPFExt.

1. NUEVA ESTRUCTURA DE CONFIGURACION

El directorio LectraPLMParam, utilizado en modo de producción para personalizar diversos archivos (xml, modelo html, menú, árboles), se ha reestructurado para introducir un directorio "personalizado" que contiene todos los archivos personalizados.

1.1 Estructura previa

ext3	01/03/2016 06:45
namespaces	01/03/2016 06:45
screens	01/03/2016 06:45
template	01/03/2016 06:45
xls	01/03/2016 06:45
ApplicationConfiguration.xml	29/02/2016 21:36
javascript.properties	29/02/2016 21:36
lastUsed.xml	29/02/2016 21:36
LPFConfiguration.xml	29/02/2016 21:36
menu.dtd	29/02/2016 21:36
menu.xml	29/02/2016 21:38
PDMPreferences.xml	29/02/2016 21:36
rootScreens.xsd	29/02/2016 21:36
saveAsDependencyRules.xml	29/02/2016 21:36
screens.xml	29/02/2016 21:36
screensCusto.xml	29/02/2016 21:36
SpecPackageTreeView.xml	29/02/2016 21:36
TemplateSwitch.xml	29/02/2016 21:36
webclient.config	29/02/2016 21:36

1.2 Nueva estructura

config
i18n
lpf
lpfext
webtool

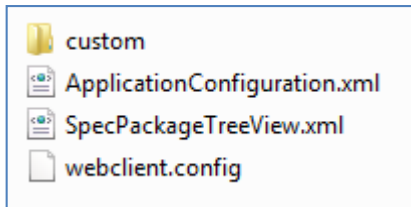
Se han creado {5} productos, y cada uno de ellos contiene un tipo de archivo (según las tecnologías o el uso).

También se han eliminado ciertos archivos que ya no se pueden personalizar o que no se han personalizado durante un cierto período.

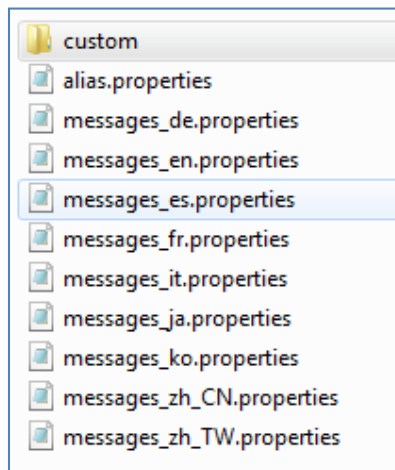
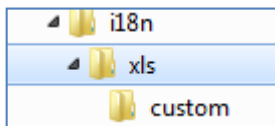


Cada uno de estos directorios contiene un directorio "personalizado", y esta es la única ubicación en la que se deben añadir los archivos personalizados.

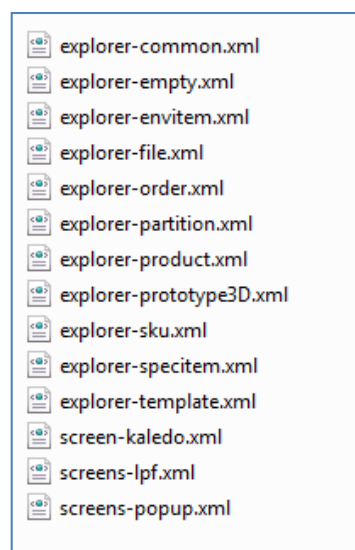
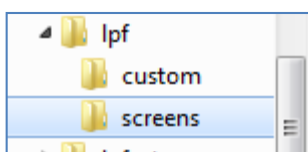
1.3 Directorio Config



1.4 Directorio I18n



1.5 Directorio Ipf



En esta tecnología no existe la herencia, el archivo que se va a personalizar debe copiarse entero en el directorio personalizado y luego modificarse con arreglo a la configuración que haga falta.

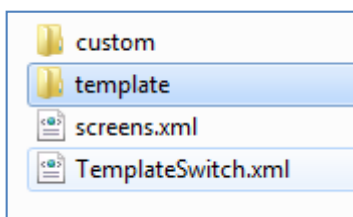
1.6 Directorio Ipfext

Contiene la estructura estándar para los archivos de namespace (xml).


En esta tecnología no hay grandes cambios.

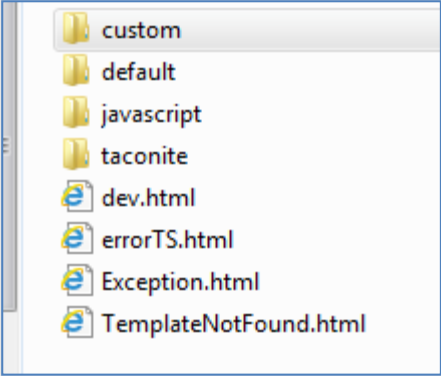
1.7 Directorio Webtool

Contiene los dos archivos Screens.xml y Template Switch.xml y el directorio de modelos.



El directorio de modelos contiene todos los modelos HTML de Product Developer y su propio directorio de modelos.

 Es el único directorio especificado para la personalización o para sobrescribir modelos ya existentes:



Todos los modelos HTML personalizados para los clientes deben estar en el mismo plano en el directorio "personalizado", independientemente de la jerarquía inicial de la norma. Product Developer recupera los modelos HTML prioritariamente en el directorio "personalizado" (si existen).

1.8 Operación

1.8.1 Operación anterior

Al instalar PLM e iniciar JBoss, si no existe se crea el directorio LectraPLMParam y se copia su contenido desde WebClient WAR.

Cada vez que se inicia JBoss, se comprueba la presencia de LectraPLMParam. Si no existe, se vuelven a copiar los recursos.

1.8.2 Nueva operación

Cada vez que se inicia JBoss (después de una nueva instalación o simplemente una parada / puesta en marcha), el contenido de LectraPLMParam es reemplazado por nuevos recursos de WebClient WAR, salvo por los directorios personalizados, que no se modifican.



¡Todas las modificaciones introducidas para personalizar los archivos originales se pierden!

2. DEFINICIONES Y CONCEPTOS

2.1 Archivos XML

Para definir pantallas se pueden utilizar varios archivos XML de descripción. A diferencia de las versiones anteriores, desde la versión V3R2 se crea un archivo XML para una pantalla o incluso una parte de la pantalla.

Cuando se lee el código fuente de un módulo de desarrollo de producto, se observa que en el encabezado del código fuente se puede introducir la variable "**screenName**" (nombre de pantalla), cuyo valor indica el nombre de la pantalla.

2.1.1 Archivo screens.xml

El archivo de entrada a tener en cuenta para saber qué archivo XML hay que modificar para personalizar una pantalla es el siguiente: [PLM-Fashion\PDM\LectraPLMParam\webtool\screens.xml](#). Este archivo hace referencia a todas las pantallas definidas con las técnicas anteriores, así como con las nuevas.

Ejemplo:

Para la pantalla **ProductGO**, tenemos la descripción de la nueva pestaña **General Objectives**, así como la parte de su **Description** y la descripción de las otras pestañas que todavía no figuren en LPFExt.

```
<!-- Generic screen for Products General Objectives. -->
<screen name="ProductGO" displayCustoms="false">
  <icons>
    <icon name="defaultImageField" />
  </icons>

  <!-- Tabs -->
  <tabs name="pgoTabs" displayed="false" placement="top"
  prefKey="ProductGO.objectives.selectedTab"
  tabManager="com.lectra.pdm.webapp.tabs.ProductTabManager"
  tabSelectionStrategy="com.lectra.pdm.webtool.config.tab.ProductGOTabStrategy">

  <!-- Objectives tab -->
  <tab name="descriptionTab" i18nKey="ProductGO.description.Title"
  autoScroll="true" autoHeight="false"
  lpfExtPath="http://lectra.com/pdm/productgo#description" />

  <!-- Cost and Margin tab -->
```



```
<tab name="costTab" i18nKey="ProductGO.description.cost" autoLoad="true"
autoScroll="true" autoHeight="false">
  <block name="detailcost" displayed="true" columns="4" template="Content"
extendedProfileTag="PrivateArea" titled="true">
  <block name="detailcostCol2" titled="true"
template="screens\Block_3_hr.html">
    <field name="targetCost.defaultPrice" editable="true"/>
    <field name="targetCost.defaultCollectionPrice" editable="true"/>
    <field name="targetCost.approved" editable="true"/>
  </block>
.../...
```

2.1.2 IpfExtPath


Esta información tiene que estar en el archivo `PLM-Fashion\PDM\LectraPLMParam\webtool\screens.xml` para cada LPFExt, e indica la ruta del archivo XML correspondiente. Todos los archivos XML se encuentran en la carpeta **Namespaces**.

Ejemplo:

```
<!-- Objectives tab -->
<tab name="descriptionTab" i18nKey="ProductGO.description.Title"
autoScroll="true" autoHeight="false"
IpfExtPath="http://lectra.com/pdm/productgo#description" />
```

El contenido de la pestaña **descriptionTab** se describe en un archivo cuyo **prefijo** es **productgo**, y de nombre **Lectra.PDM.ProductGO**.

Dentro del archivo XML, el contenido de la pestaña se describe en el bloque de **description**.

 Este bloque de descripción corresponde a un contenedor de tipo panel. Consulte el apartado [2.1.4 - Descripción de la pantalla: enfoque técnico](#).

2.1.3 Namespaces

Un archivo XML de descripción tiene la siguiente estructura:

```
<namespace uri="http://lectra.com/pdm/productgo" prefix="productgo"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xs="http://www.w3.org/2001/XMLSchema">

<!-- comment -->
<element1 name="elt1Name"> ... </element1>
<!-- comment -->
<element2 name="elt2Name"> ... </element2>

<!-- Private field(s) : do not edit -->
<private> ... </private>
<!-- End of private field(s) -->

</namespace>
```

Ejemplo: `Lectra.PDM.ProductGO.xml`

El archivo de descripción debe declararse mediante una etiqueta **namespace**.

Esta etiqueta **debe** contener al menos los siguientes atributos:

- **uri**: identificador del archivo de descripción. Incluye una sección que varía en función del **prefijo** del archivo.
- **prefijo**: identificador que describe la pantalla. Se cita en el atributo **uri** y en el archivo [screens.xml](#). Identifica únicamente el archivo.
- **xmlns:xsi** con el valor correspondiente: "**http://www.w3.org/2001/XMLSchema-instance**"
- **xmlns:xs** con el valor correspondiente: "**http://www.w3.org/2001/XMLSchema**"

El atributo **uri** que contiene el identificador del archivo XML también figura en el archivo [namespace.properties](#).

Este archivo contiene la definición del identificador en función de todos los archivos de descripción.

Los archivos de descripción contienen comentarios que comienzan por "**<!--**" y terminan por "**-->**".

También contienen secciones privadas que los usuarios no deben modificar. Estas secciones comienzan con el siguiente comentario:

```
<!-- Private field(s) : do not edit -->
```

y terminan con:

```
<!-- End of private field(s) -->
```

2.1.4 Descripción de la pantalla: enfoque técnico

La descripción de la pantalla comienza con la definición del encabezado del archivo XML con la etiqueta **namespace** que figura en la sección [2.1.3 Namespaces](#).

A continuación se define la pantalla mediante los elementos gráficos que la componen, y que se describen en el archivo de XML de base (en la carpeta namespaces), así como en el archivo xml del mismo nombre situado en la subcarpeta custom.

Estos elementos son elementos gráficos de la biblioteca ExtJS y la estructura de LPFExt. El más utilizado es `<panel>`, que siempre se define entre `< >`.

Definiciones de "panel":

`<panel>` es una etiqueta de apertura de un panel

`</panel>` es una etiqueta de cierre de un panel.

Todas las características descriptivas del elemento gráfico "Panel" se definen en la etiqueta de apertura: disposición, colores, ancho, desplazamiento...

Todos los demás elementos gráficos de este panel se definen entre la etiqueta de apertura y la de cierre. Normalmente un panel se compone de otros elementos `<panel>`, compuestos ellos mismos a su vez de `<combobox>`, `<nodecombo>`, `<textarea>`, `<textfield>`, `<richText>`, `<image>`, etc...

2.2 Herencia

La nueva estructura LPFExt permite acceder más rápidamente a las pantallas. Con el uso de varios archivos XML de descripción, se establece un sistema de herencia que facilita la lectura de pantallas, así como una estandarización de los elementos genéricos.

A diferencia de las anteriores técnicas de descripción, para definir la herencia se emplea una palabra clave única: **"inherits"**. En lo sucesivo se pierde la noción de modificación de los elementos heredados.

Hay dos tipos de herencia: herencia interna y herencia externa.

2.2.1 Herencia interna

Es el modo de herencia más frecuente, especialmente para definir las descripciones de las pantallas para las diferentes categorías de productos.

En la herencia interna se utiliza un elemento declarado dentro del archivo de descripción de la pantalla.

```
<parent name="parentNameExample">
#   <!-- child element -->
#   <child inherits="childNameExample">
</parent>

<child name="childNameExample">
#   <!-- Title of child -->
#   <title>Titre</title>
</child>
```

La etiqueta <parent> toma como elemento la etiqueta <child> (definida más abajo). En este ejemplo, la etiqueta <child name= « childNameExample »> define el contenido del elemento <child>.

Esto equivale a escribir:

```
<parent name="parentNameExample">
#   <!-- child element -->
#   <child name="childNameExample">
#     #   <!-- Title of child -->
#     #   <title>Titre</title>
#     </child>
</parent>
```

Para un determinado archivo xml de descripción, es muy recomendable que el usuario añada sus propias herencias en el archivo XML específico del mismo nombre que se encuentra en la subcarpeta namespaces/custom.

Ejemplo:

- El archivo de descripción `namespaces/Lectra.PDM.ProductGO.xml` contiene la definición de un panel de descripción que hereda de varios paneles posibles:
 - `<panel name="description_wrap" title18n="description" region="north" border="false" height="300" split="true" collapsible="true" statelId="panel_{$#type}" layout="fit" maximizableChild="true">`
 - `<panel name="description" inherits="description_{$#type},description_{$#topCategoryName},description" />`
 - `<plugin xsi:type="xmlmap" ptype="panelCollapsedTitle" />`
 - `</panel>`
- En el archivo `namespaces/custom/Lectra.PDM.ProductGO.xml`, puede definir su propia descripción **para una subcategoría personalizada, bajo el modelo siguiente:**

Habiendo definido "MyCustomCategory" en el módulo de Administración y configuración:

```
<panel name="description_MyCustomCategory">
  <!-- Add your custom panels here -->
</panel>
```

Como esta subcategoría es un tipo, el panel `description_MyCustomCategory` corresponde a la herencia definida por `description_{$#type}`, que se posiciona primero.

2.2.2 Herencia externa

En la herencia externa se declara un elemento en un archivo de descripción (véase el ejemplo a continuación: **Common.xml**) y después se utiliza en otro archivo de descripción de pantalla (véase el ejemplo a continuación: **Ecran.xml**).

```
<namespace uri="http://lectra.com/pdm/example/common"
  prefix="common" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <child name="childNameExample">
  #   <!-- Title of child -->
  #   <title>Titre</title>
  </child>

</namespace>
```

Lectra.PDM.Example.Common.xml

```
<namespace uri="http://lectra.com/pdm/example/ecran" prefix="ecran"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <parent name="parentNameExample">
  <!-- child element -->
  <child inherits="http://lectra.com/pdm/example/common#childNameExample">
  </parent>
```

```
</namespace>
```

Lectra.PDM.Example.Ecran.xml

La etiqueta <parent> considera que la etiqueta <child> indica el título del elemento, pero que se define en otro archivo de descripción.

El valor del atributo "inherits" se construye como sigue:

`#{URI del archivo de descripción que contiene el elemento}###{NAME del elemento}`

Ejemplo:

Por ejemplo, la herencia se utiliza para las barras de herramientas que aparecen en varias pantallas.

Namespaces/Lectra.PDM.ChangeTracking.xml utiliza una barra de herramientas que hereda la barra de herramientas de paginación definida en el archivo **namespaces/Lectra.PDM.Common.xml**.

```
<paging name="pagingToolbar" displayInfo="true" pageSize="25">  
  <tbseparator />  
  <plugin xsi:type="pagingtoolbarresizer">  
    <option>25</option>  
    <option>50</option>  
    <option>100</option>  
  </plugin>  
</paging>
```

Lectra.PDM.Common.xml

```
<bbar xsi:type="paging"  
inherits="http://lectra.com/pdm/common#pagingToolbar" pageSize="7" />
```

Lectra.PDM.ChangeTracking.xml

2.2.3 Creación de nuevos archivos

El usuario puede crear sus propias pantallas definiéndolas en el archivo XML correspondiente. Se recomienda identificar estos nuevos archivos creándolos en una carpeta específica, por ejemplo, en una subcarpeta de **namespaces** llamada **MyCompany**.

- Cree el archivo xml siguiendo la estructura descrita en la sección [2.1.3 Namespaces](#).
- Haga referencia al nuevo archivo en el archivo **namespaces.properties**.

```
#Lectra namespaces  
${DIRECTORY}namespaces/« MyCompany »/Lectra.PDM.Common =  
http://lectra.com/pdm/common
```

- Se recomienda incluir una referencia al nuevo archivo en el archivo **screensCusto.xml** que contendrá las declaraciones .xml de las pantallas, duplicadas y modificadas, como se hace para la pantalla **ProductGO**, por ejemplo:

```
<screen name="ProductGO" displayCustoms="false">
<icons>
  <icon name="defaultImageField" />
</icons>

<!-- Tabs -->
<tabs name="pgoTabs" displayed="false" placement="top"
  prefKey="ProductGO.objectives.selectedTab"
  tabManager="com.lectra.pdm.webapp.tabs.ProductTabManager"
  tabSelectionStrategy="com.lectra.pdm.webtool.config.tab.ProductGOTabStrategy">
  <tab name="descriptionTab" idKey="ProductGO.description.Title"
    lpfExtPath="http://lectra.com/pdm/productgo#description_Style" />
</tabs>
</screen>
```

2.2.4 Propiedades generales

- La herencia se define mediante el atributo **"inherits"** y toma el valor del atributo **name** (nombre) del elemento que se va a integrar.
- El tipo de etiqueta (<panel> por ejemplo) al que hace referencia la herencia (declarando el atributo **"inherits"**) debe ser **idéntico** al de la etiqueta declarada.

2.3 Lo que hay que saber sobre la estructura de descripción de las pantallas

2.3.1 La búsqueda de pantalla comienza leyendo el archivo screens.xml

La pantalla de búsqueda se identifica mediante:

la etiqueta <screen> seguida de un campo **name** cuyo valor se corresponde con el **prefix** (prefijo) de la pantalla.

Para esta etiqueta, identificamos la pestaña que nos interesa, la etiqueta <tab>, así como el nombre del archivo xml, y el nombre del panel correspondiente en el **lpfExtPath** asociado.

Ejemplo:

```
<!-- Generic screen for Products General Objectives. -->
<screen name="ProductGO" displayCustoms="false">
  <icons>
    <icon name="defaultImageField" />
  </icons>

  <!-- Tabs -->
  <tabs name="pgoTabs" displayed="false" placement="top"
    prefKey="ProductGO.objectives.selectedTab"
    tabManager="com.lectra.pdm.webapp.tabs.ProductTabManager"
    tabSelectionStrategy="com.lectra.pdm.webtool.config.tab.ProductGOTabStrategy">
    <!-- Objectives tab -->
    <!-- NUEVO PROCEDIMIENTO -->
```

```

<tab name="descriptionTab" il8nKey="ProductGO.description.Title"
  autoScroll="true" autoHeight="false"
  lpfExtPath="http://lectra.com/pdm/productgo#description" />

<!-- Cost and Margin tab -->
<!-- ANTIGUO PROCEDIMIENTO -->
<tab name="costTab" il8nKey="ProductGO.description.cost"
  autoLoad="true" autoScroll="true" autoHeight="false">
  <block name="detailcost" displayed="true" columns="4"
  template="Content" extendedProfileTag="PrivateArea"
  titled="true">
  <block name="detailcostCol2" titled="true"
  template="screens\Block_3_hr.html">

  ..../

```

Con las nuevas técnicas, la ruta que lleva al archivo xml correspondiente se expresa como sigue:

```
lpfExtPath="http://lectra.com/pdm/productgo#description"
```

```
lpfExtPath="http://lectra.com/pdm/ « prefix du fichier xml »# « nom du
panel » "
```

En este ejemplo, la primera pestaña (**Objectives**) corresponde al archivo **Lectra.PDM.ProductGO** que tiene el mismo prefijo que **ProductGO** y que el panel **description**.



Los bloques que se describen con las técnicas anteriores en el archivo **screens.xml** vienen definidos ahora por los **paneles** del archivo xml específico.

2.3.2 Lectura del archivo xml de base

En el ejemplo dado, la declaración del panel description se busca en el archivo Lectra.PDM.ProductGO.xml.

```

<panel name="description"
inherits="description_${#type},description_${#topCategoryName},description"
/>

```

Esto significa que, si se define *description* `<panel>`, hereda:

- En la primera posición de `<panel name = "description_${#type}" ...>`; la etiqueta `_${#type}` define el tipo actual en el contexto, a elegir de entre los tipos posibles: Style, MyCustomStyle, Fabrics, Denim, Fur, Bags, Order, SKU, etc...
- En segunda posición (si no se ha definido la primera) de `<panel name = "description_${#topCategoryName}" ...>` la etiqueta `_${#topCategoryName}` define la categoría principal del contexto: Styles, Fabrics, Trims o PackagingLabel.
- En tercera posición, si no se definen las otras dos, `<panel name = "description" ...>` se define por defecto en el archivo xml de base.

Para determinar cuáles se han definido deben buscarse alternativas en el documento o el archivo xml del mismo nombre de la subcarpeta custom.

2.3.3 Modificación de la descripción de la pantalla

En los párrafos y ejemplos siguientes, se explica cómo editar, añadir <panel>, añadir campos de tipo <nodecombo>, <textarea>, <textfield>, etc...

- Uno de los métodos consiste en reescribir los párrafos (<panel>) correspondientes con el contenido modificado. Es la forma más fácil de empezar.
- Otro método es utilizar el concepto de herencia que se explica en la sección 2.2 - [Herencia](#). La herencia se integra en 6 [.Configuración avanzada para añadir campos y paneles](#).

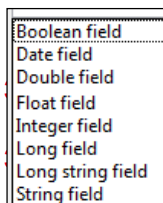
3. CONFIGURACIÓN AÑADIENDO CAMPOS A "GENERAL OBJECTIVES"

La pantalla **General Objectives** de los productos (**Styles, Materials, Trims y Packaging Labels**) puede configurarse mediante la descripción de la interfaz en el archivo `namespaces/Lectra.PDM.ProductGO.xml`.

3.1 Agregar campos simples

3.1.1 Definición

Los campos simples son tipos de campos correspondientes a los siguientes datos del módulo de Administración y configuración:



Deben aplicarse todos estos pasos:

3.1.4

3.1.5 Paso 1: Requisitos del módulo de Administración y configuración

Todos los campos que se añadan a la pantalla deben definirse en el módulo de Administración y configuración

Puede utilizar los numerosos campos disponibles o crear otros nuevos. En este caso, consulte la ayuda en línea correspondiente.



También puede consultar el recordatorio del [Anexo H - Cambio de ubicación del directorio de LPFExt](#)

En PLM V5R1, los archivos LPFExt se encuentran en `lpf/ext3/nombre del directorio`.

Recordatorio de la aplicación de Administración y configuración

3.1.6 Paso 2: Configuración en el archivo XML específico

Para añadir campos a la pantalla **General Objectives** de los **Productos**, es necesario introducirlos en el archivo que se encuentra en la subcarpeta "**custom**" [namespaces/custom/Lectra.PDM.ProductGO.xml](#) (y no el archivo [namespaces/Lectra.PDM.ProductGO.xml](#)).

Si no existe, es necesario crearlo en la carpeta custom y estructurarlo del siguiente modo:

```

<namespace uri="http://lectra.com/pdm/productgo/custom"
           prefix="productgo"
           xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
           xmlns:xs="http://www.w3.org/2001/XMLSchema">

<!--      AÑADIR AQUÍ LA CONFIGURACIÓN  -->
<!--      _____      -->

</namespace>
```

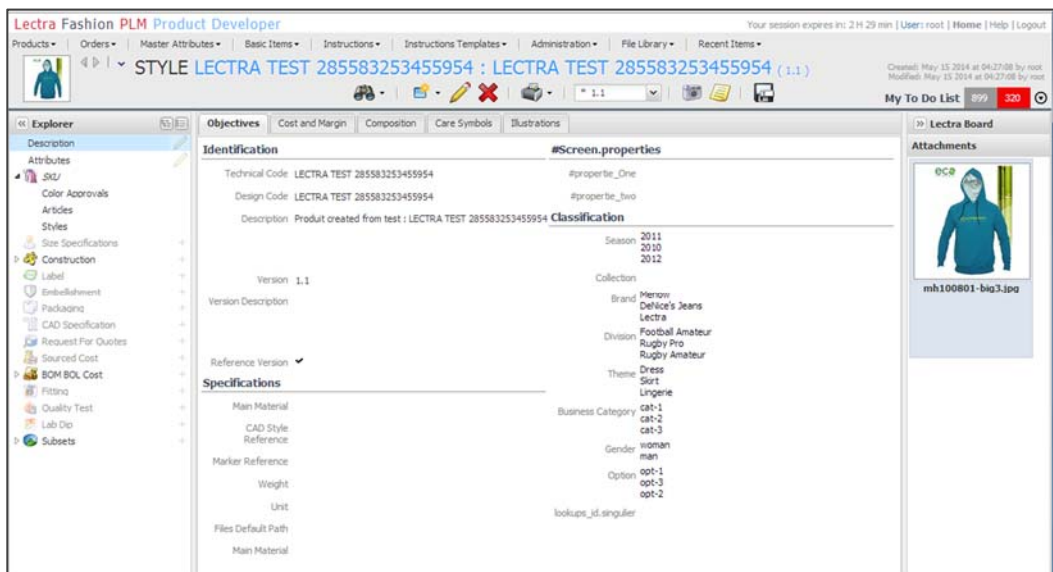
3.1.7 Paso 3: Identificar la parte que se va a duplicar y modificar

Para cada modificación:

- Identifique la parte correspondiente en el archivo [/namespaces/LectraPDM.ProductGO.xml](#) (xml de base).
- Copie el <panel> que contiene esta parte en el archivo [/namespaces/custom/LectraPDM.ProductGO.xml](#) (custom xml).
- Introduzca la modificación en esta parte duplicada.
- Actualice la presentación: utilice la tecla **F5** para actualizar la pantalla.

3.1.7.1 Identificación de la parte a modificar en el archivo XML de base

- Para la categoría de STYLE (modelo):



Busque el panel de descripción Style (modelo) = contenido de la pestaña **General Objectives**.

En el archivo [/namespaces/LectraPDM.ProductGO.xml](#), busque la parte que describe **General Objectives** para la categoría Style (modelo).

```
<panel name="description_Style" layout="column" autoScroll="true" >
```

Este panel está formado por varios paneles...

Busque el panel que se va a modificar.

- Para modificar el `<panel>` **"Identification (Identificación)"** `<`, busque:

```
<panel name="identification" titleI18n="Screen.identification"
inherits="looklikefieldset" >
    .../...
</panel>
```

- Para modificar el `<panel>` **"Classification (Clasificación)"**, busque:

```
<panel name="collection" titleI18n="Screen.collection"
columnWidth="0.5" inherits="looklikefieldset" >
    .../...
</panel>
```

- Para modificar el `<panel>` **"Specifications (Especificaciones)"**, busque:

```
<panel name="detail" titleI18n="Screen.specification"
inherits="looklikefieldset">
    .../...
</panel>
```

- Para modificar el panel **"Validation Table (Tabla de validación)"** `<`, busque:

```
<panel name="validationTable" ... >
    .../...
</panel>
```

Si desea ocultar los campos generados en el panel de la tabla de validación, cambie la propiedad `headerOnly="false"` por `headerOnly="true"`.

Si desea mostrar un campo de validación en la pantalla ProductGo, debe añadir un campo específico en función del tipo de campo que se va a mostrar. El nombre del campo corresponde a la concatenación del nombre de la tabla y el del campo (por ejemplo: table name: 'VT', field name: 'Weight' => field name: 'VT_Weight').

Para un campo numérico, añada la siguiente línea:

```
<numberfield isValidaionField="true" titleI18n="Weight" name="Table_Weight" >
```

Para un campo de texto, añada la siguiente línea:

```
<textfield isValidatIonField="true" title18n="Name" name="Table_Name" >
```

Para un campo de fecha, añada la siguiente línea:

Si desea hacer un cambio en función del huso horario del cliente:

```
<datefield isValidatIonField="true" title18n="Birth" name="Table_Birth" >
```

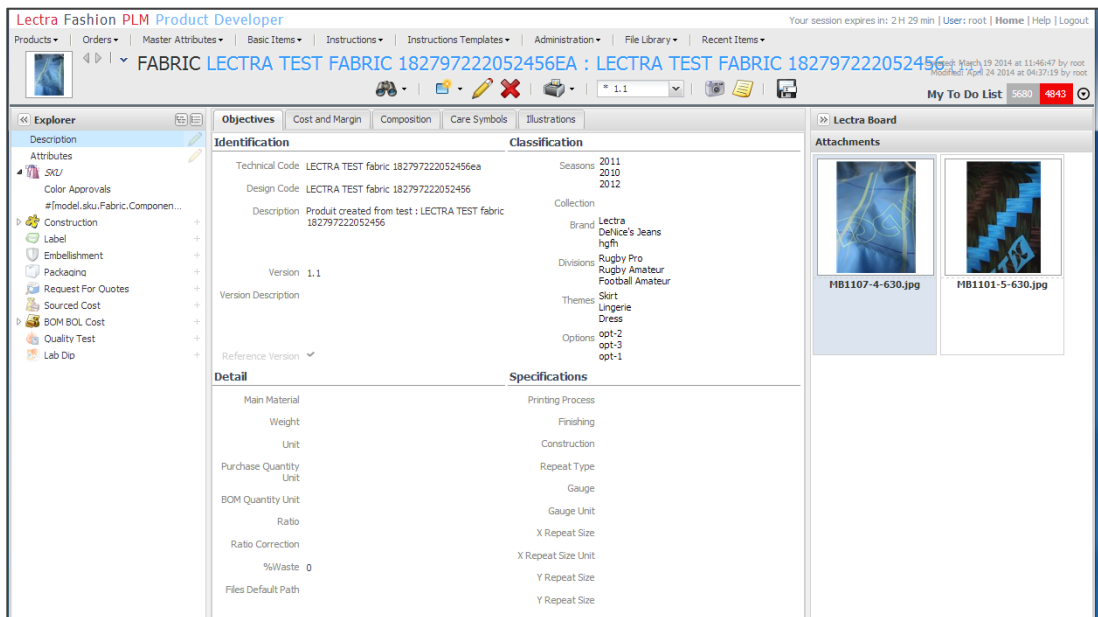
Si desea que la fecha cambie según el huso horario del cliente:

```
<datefield          isValidatIonField="true"          title18n="Birth"          name="Table_Birth"
useServerTimezone="true">
```

Para un campo de casilla de verificación, añada la siguiente línea:

```
<xcheckbox  isValidatIonField="true"  title18n="International"  name="Table_International"
readOnly="true" >
```

- Para las categorías FABRIC, TRIMS y PACKAGINGLABEL:



Busque el panel de descripción Fabric.

En el archivo [/namespaces/LectraPDM.ProductGO.xml](#), busque la parte que describe **GeneralObjectives** para la categoría Fabric.

```
<panel name="description_default" layout="column" autoScroll="true" >
```

Siga los paneles relacionados por herencia hasta llegar al panel que corresponde a la parte que se va a modificar.

Ejemplo: el `<panel>` "description" a continuación contiene el panel "north" que hereda del panel "description_north_default" que contiene el `<panel>` "identification" y el `<panel>` "classification".



Consulte la sección [2.2 - Herencia](#).

Busque el panel que se va a modificar.

- Para modificar el <panel> "**Identification (Identificación)**" de Fabric, busque:

```
<panel name="identification_default"
titleI18n="Screen.identification" inherits="looklikefieldset">
.../...
</panel>
```

- Para modificar el <panel> "**Classification (Clasificación)**", busque:

```
<panel inherits="looklikefieldset" name="classification_default"
titleI18n="Screen.collection" >
.../...
</panel>
```

- Para modificar el panel > "**Details (Detalles)**" <, busque:

```
- <panel name="detail_FabricTrim" inherits="detail_default">
- .../...
- </panel>
```

- Para modificar el <panel> "**Specifications (Especificaciones)**", busque:

```
- <panel name="specification_Fabric" inherits="specification_default">
- .../...
- </panel>
```

3.1.7.2 Duplicación de la parte a modificar del archivo XML personalizado

Copie la descripción del **panel** que contiene el **panel** que se va a modificar en el archivo [namespaces/custom/Lectra.PDM.ProductGO.xml](#).

Ejemplo: Para modificar el panel Style identification:

```
<panel name="identification" titleI18n="Screen.identification" inherits="looklikefieldset" >
```

Copie todo el panel Description Style:

```
<panel name="description_Style" layout="column" autoScroll="true" >
.../...
</panel >
```

3.1.7.3 Añadir la descripción xml de un campo de texto

Una vez identificada y duplicada la parte pertinente (si aún no se ha hecho) en el archivo XML, añada la siguiente línea al **panel** en la ubicación deseada:

```
<textfield name="MyCustomStringField" fieldLabelI18n = "MyCustomStringField" />
```



Para añadir otros tipos de campos simples, véase el [Anexo J - Añadir otros tipos de campos simples en xml](#).

3.1.8 Paso 4: Referenciar las traducciones del campo añadido

Si ha añadido un campo personalizado llamado **MyCustomStringField** en el módulo de Administración y configuración, aparece la pantalla del módulo de desarrollo del producto:



Si el nombre del archivo va precedido del carácter #, quiere decir que no existe la traducción del nombre.

Traduzca este texto modificando el archivo [PLM-Fashion\PDMLectraPLMParam\messages_en.properties](#) y añadiéndolo al final del archivo:

MyCustomStringField = My String Field.

El resultado es el siguiente:



En Internet Explorer, ejecute el siguiente comando:

`http:// "Nombre del servidor"/pdm/admin.SessionMonitor.reloadConfig.wbx`

En el menú Internacionalización del módulo de Administración y configuración, haga clic en Vaciar la caché.

3.2 Añadir una lista

3.2.1 Definición

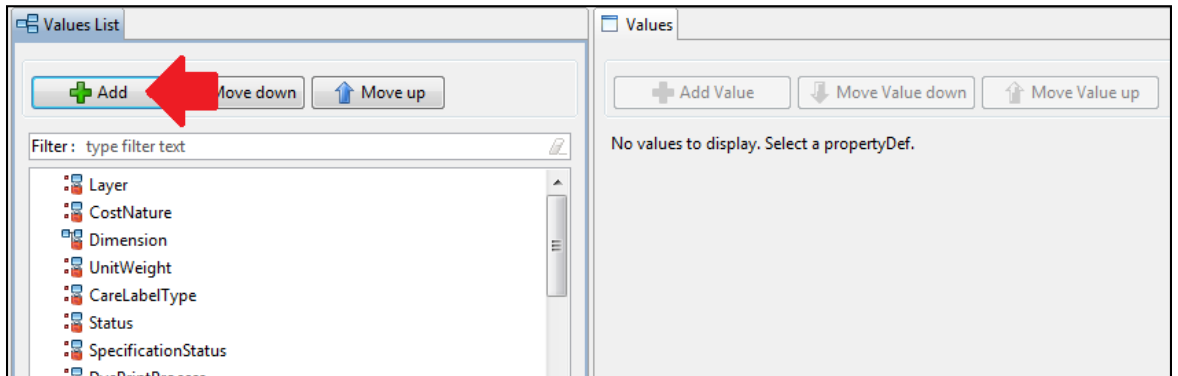
Estas listas, llamadas **Value Lists** (Listas de valores) en la aplicación, se asocian a las categorías añadiendo un "**CustomRole**" como "**PickList**" "**Target**" para cada lista de valores.

En el módulo de desarrollo del producto, se añaden a la pantalla a través del componente `<nodecombo />`.

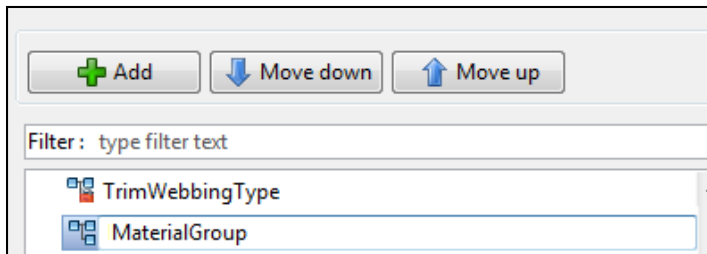
3.2.2 Paso 1: Definir la lista en el módulo de Administración y configuración y asociarla a la categoría Style

3.2.2.1 Creación

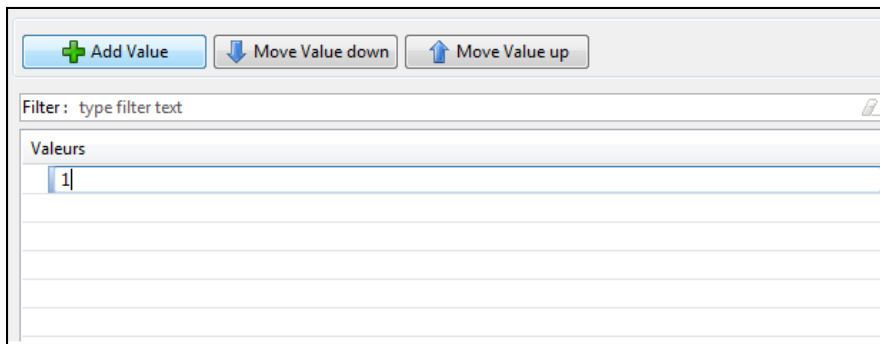
Haga clic en **Listas de valores** en el menú **Datos** de **Administración**.



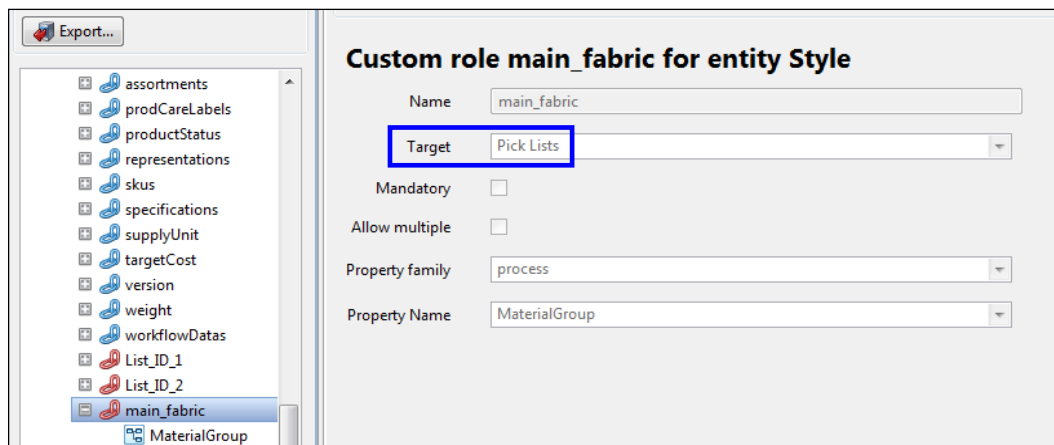
Haga clic en **Añadir** y escriba el nombre de la lista de valores.



Seleccione la nueva lista y añada los nuevos valores en la pantalla de la derecha:



3.2.2.2 Asociación con CustomRole



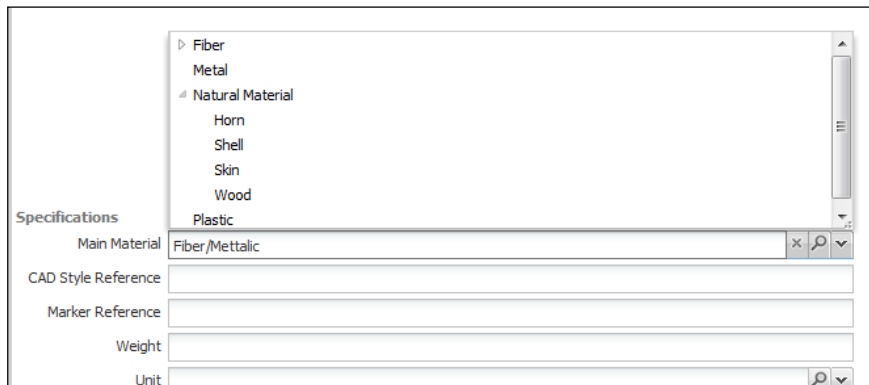
3.2.3 Paso 2: Descripción del archivo custom/Lectra.PDM.ProductGO.xml

```
<panel name="description_Style"  
  <panel name="detail":  
    <nodecombo fieldLabelI18n="Style.mainFabric"  
              hiddenName="mainFabric" />
```

3.2.4 Paso 3: Referenciar las traducciones del campo añadido

En el archivo messages_en.properties.

3.2.5 Resultado: Presentación en el módulo de desarrollo del producto



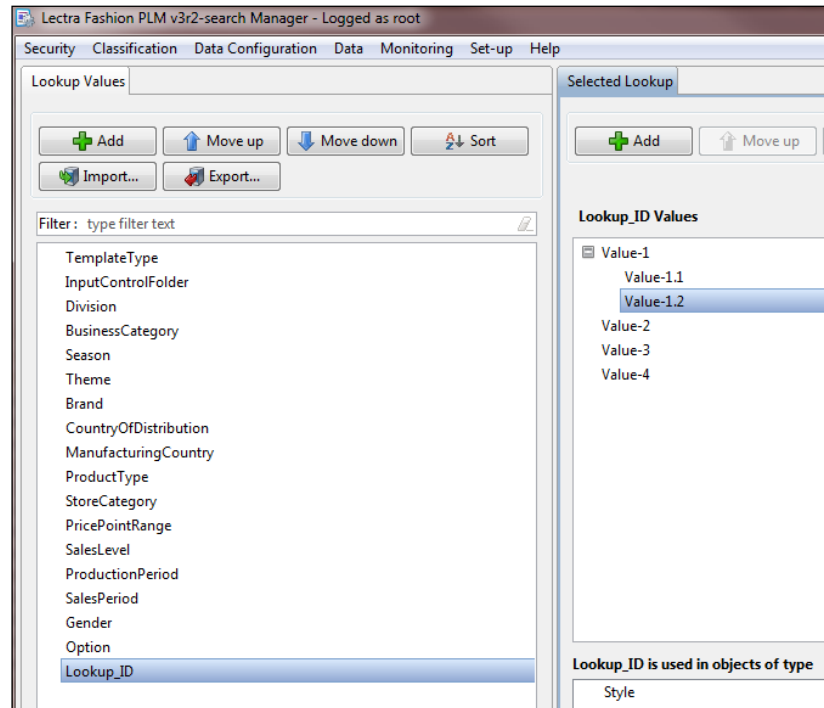
3.3 Añadir la clasificación

3.3.1 Definición

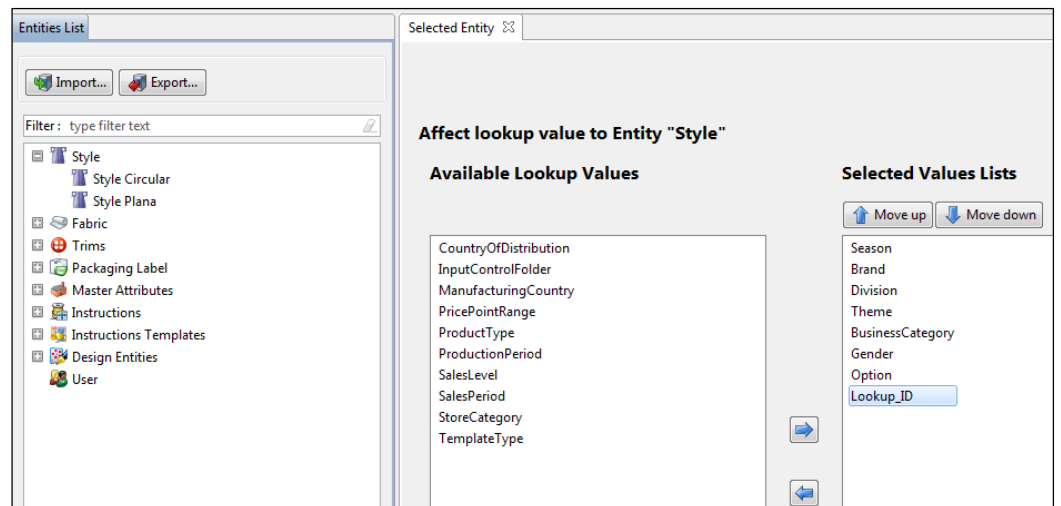
Llamados **Lookup Values** (Valores de clasificación) en el menú **Clasificación** del módulo de Administración y configuración, se pueden crear definiendo los valores de clasificación de manera jerárquica y se asocian a una o varias categorías mediante la opción **Asignar los valores de clasificación**.

3.3.2 Paso 1: Definición en el módulo de Administración y configuración

3.3.2.1 Creación



3.3.2.2 Asociación en la categoría Modelo



3.3.3 Paso 2: Descripción en el archivo Lectra.PDM.ProductGO.xml

En el módulo de desarrollo del producto, esta nueva lista debe aparecer automáticamente en la parte **Classification** de los productos correspondientes, en la página **GeneralObjectives**.

3.3.3.1 Descripción automática de la clasificación

- Compruebe que el **panel Colección** figura en el archivo XML de base, que se encuentra en la carpeta **namespaces**.

```
<panel name="collection" titleI18n="Screen.collection" columnWidth="0.5"
  inherits="looklikefieldset">
  <nodecombo fieldLabelI18n="{#axeRoleName}.singulier"
    hiddenName="{#classifRoleName}" multiSelect="true"
    propertyName="{#propertyName}" genProductClassif="prodgo" />

  <!-- Private field(s) : do not edit -->
  <customSerializer type="com.lectra.pdm.lpf.ext.serializers.
    ProductClassifGeneratorCustomSerializer" />
  <!-- End of private field(s) -->
```

</panel >

- **Custom Serializer**

Custom Serializer se utiliza para generar un campo único o un grupo de campos. Se define en los archivos XML mediante la etiqueta `<customSerializer .../>`, que por lo general se encuentra en una sección privada:

```
<!-- Private field(s) : do not edit -->
  <customSerializer ...
  <!-- End of private field(s) -->
```

Esta definición no se debe modificar. Se corre el riesgo de dañar la generación de páginas.

3.3.3.2 Descripción manual de la clasificación

- En un `<panel>` similar al anterior, es posible definir una lista de `<nodecombo>` haciendo referencia a una lista HVL definida en el módulo de Administración y configuración, sin seguir el proceso de **Custom Serializer**.

Ejemplo:

- Con una **Pick List (Lista de selección)** definida en el módulo de Administración y configuración:
 - `<nodecombo fieldLabelI18n="ApprovalStatus" hiddenName="ApprovalStatus"
 propertyName="ApprovalStatus" multiSelect="false" chkVisible="false"/>`
- Con un **Classification Lookup Value (Valor de clasificación)** definido en el módulo de Administración y configuración: "**Brand**" (marca) por ejemplo
 - `<nodecombo fieldLabelI18n="brands_isa" hiddenName="brandsClassif"
 propertyFamily="process" propertyName="Brand" multiSelect="true"/>`



`hiddenName` se debe componer del siguiente modo: `"Lookup Value Name"sClassif`.

- Si la **Pick List (Lista de selección)** o el **Classification Lookup Value (Valor de clasificación)** se introduce como criterio de búsqueda en el panel **Search**

(**Búsqueda**) del explorador de productos, ".values" debe tener una referencia en "hiddenName".

- `<nodecombo fieldLabel18n="ApprovalStatus" hiddenName="ApprovalStatus.values" propertyName="ApprovalStatus" multiSelect="false" chkVisible="false"/>`

3.3.4 Paso 3: Referenciar las traducciones del campo añadido

En el archivo `messages_en.properties`.

3.3.5 Resultados: Presentación en el módulo de desarrollo del producto



3.4 Añadir un panel de "Properties (Propiedades)"

También es posible configurar la pantalla con un nuevo "panel" compuesto por campos nuevos o ya existentes.

3.4.1 Paso 1: Identificación del panel en el que hay que añadir el panel

El "panel primario" donde se inserta el nuevo panel debe copiarse íntegramente en el archivo `custom/Lectra.PDM.ProductGO.xml`.



Si ya se ha duplicado y configurado alguno de los paneles del "panel primario" correspondiente en el archivo XML de la carpeta personalizada, es necesario copiar la definición del panel ya modificado.

3.4.2 Paso 2: Descripción en el archivo "custom/Lectra.PDM.ProductGO.xml"

Para añadir el panel "Properties (Propiedades)" a la pantalla **GeneralObjectives** de la categoría **Styles**:

3.4.2.1 Duplicación de la definición del panel "description_Style"

```
<panel name="description_Style" layout="column" autoScroll="true" padding="4" >
```

```

<panel columnWidth="0.5" border="false">
  <panel name="identification" titleI18n="Screen.identification"
    inherits="looklikefieldset" >
    .../...
  </panel>
  <panel name="detail" titleI18n="Screen.specification"
    inherits="looklikefieldset" >
    .../...
  </panel>
  <panel name="collection" titleI18n="Screen.collection"
    columnWidth="0.5" inherits="looklikefieldset" >
    .../...
  </panel>
</panel>

```

Para añadir el panel **"Properties (Propiedades)"** a la pantalla **GeneralObjectives** de las categorías Fabrics, Trims y PackagingLabel (Tejidos, Fornituras y Etiquetas – Empaquetado):

3.4.2.2 Duplicación de la definición del panel "description_north_default" o "description_south_default"

```

<panel name="description_north_default" border="false" layout="column">
  <panel name="identification"
    inherits="identification_${#categoryName},identification_${#top
      CategoryName},identification_default" columnWidth="0.5"/>
  <panel name="classification"
    inherits="classification_${#categoryName},classification_${#top
      CategoryName},classification_default" columnWidth="0.5"/>
</panel>
<panel name="description_south_default" border="false" layout="column">
  <panel name="detail" inherits="detail_${#categoryName},
    detail_${#topCategoryName}, detail_default" columnWidth="0.5"/>
  <panel name="specification" inherits="specification_${#categoryName},
    specification_${#topCategoryName}, specification_default"
    columnWidth="0.5"/>
</panel>

```

3.4.2.3 Añadir la definición del panel "Properties (Propiedades)"

```

<panel name="description_Style" layout="column" autoScroll="true"
  padding="4">
  <panel name="identification" titleI18n="Screen.identification"
    inherits="looklikefieldset">
    .../...
  </panel>
  <panel name="detail" titleI18n="Screen.specification"
    inherits="looklikefieldset">
    .../...
  </panel>
  <panel name="properties" titleI18n="Screen.properties"
    columnWidth="0.5" inherits="looklikefieldset">
    <textfield fieldLabelI18n="propertie_One" />
    <textfield fieldLabelI18n="propertie_two" />
  </panel>
  <panel name="collection" titleI18n="Screen.collection"
    columnWidth="0.5" inherits="looklikefieldset">

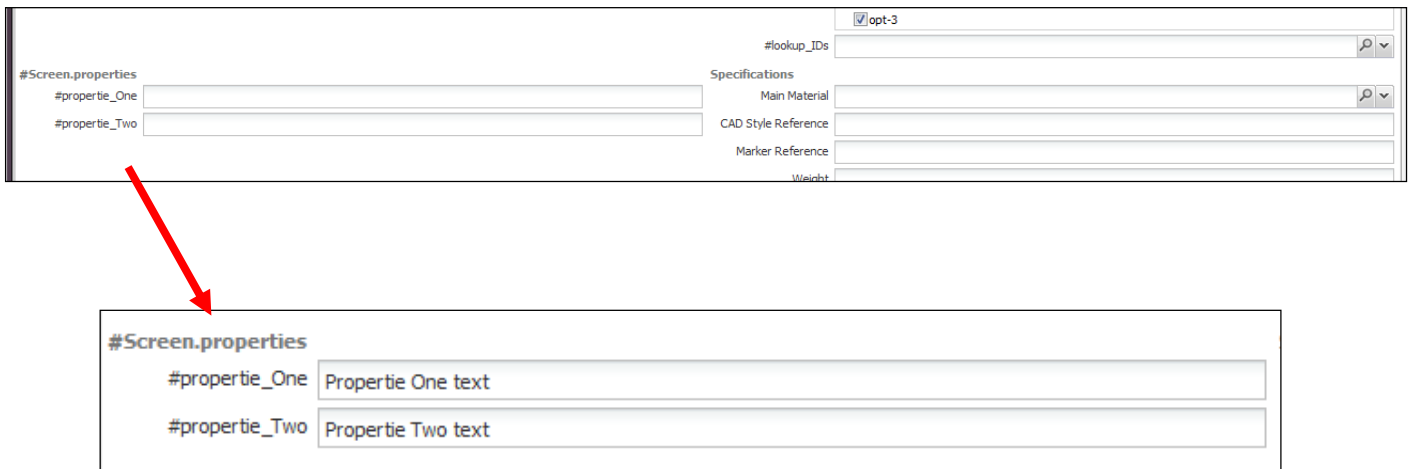
```

```
.../...  
</panel>  
</panel>
```

3.4.3 Paso 3: Referenciar las traducciones de los nombres de los campos añadidos

En el archivo `messages_en.properties`, traducción de `Screen.properties`, `propertie_One`, `propertie_Two`

3.4.4 Resultado: Presentación en el módulo de desarrollo del producto



4. CONFIGURACION AÑADIENDO CAMPOS EN EL EXPLORADOR DE PRODUCTOS

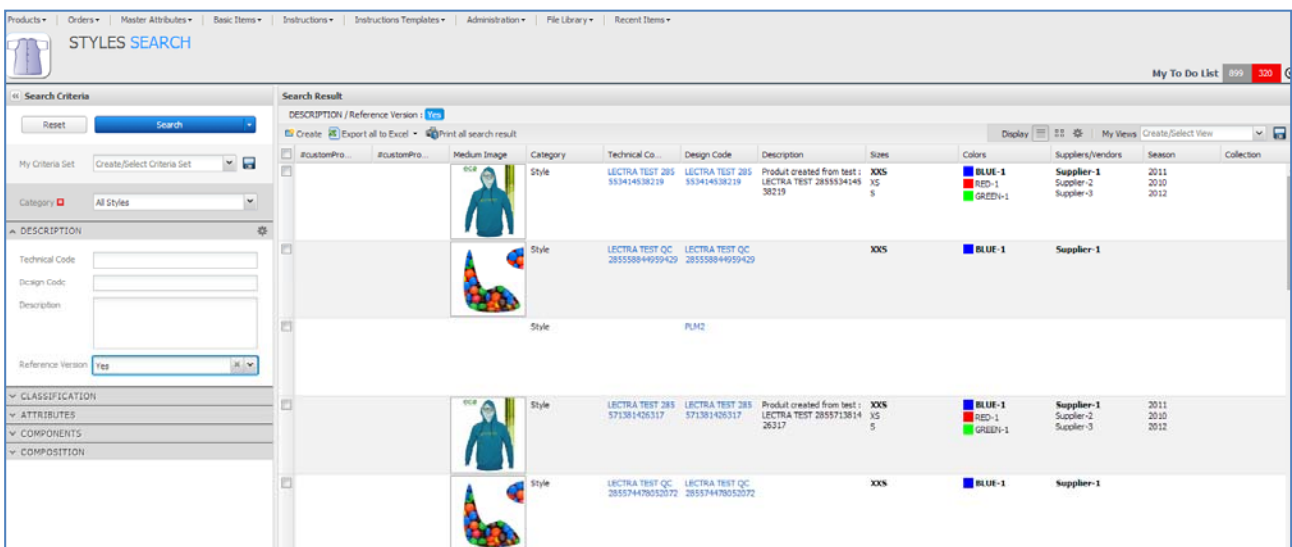
4.1 Requisitos del módulo de Administración y configuración

Los campos personalizados que se van a añadir a esta pantalla deben definirse en el módulo de Administración y configuración, como se indica en el [Anexo H - Cambio de ubicación del directorio de LPFExt](#)

En PLM V5R1, los archivos LPFExt se encuentran en lpf/ext3/nombre del directorio.

Recordatorio de la aplicación de Administración y configuración.

4.2 Presentación de la pantalla Product (Producto)



Esta pantalla es idéntica para los distintos productos: **Styles (Modelos)**, **Fabrics (Tejidos)**, **Trims (Fornituras)** y **PackagingLabel (Etiquetas – Empaquetado)**. Consta de dos partes personalizables:

- El panel **Search (Búsqueda)**, que aparece en la parte izquierda con el nombre de **Search Criteria (Criterios de búsqueda)**. Esta parte puede configurarse a través del módulo de Administración y configuración añadiendo las definiciones del archivo XML (véase el procedimiento para hacerlo a continuación).
- La cuadrícula de **Results (Resultados)**, situada a la derecha, se puede configurar directamente en el módulo de Administración y configuración agregando o quitando columnas, cambiando el orden de aparición, etc. Consulte la guía de usuario del módulo de desarrollo del producto.

Para configurar el panel de búsqueda, es importante entender cómo se define el archivo XML y cómo se puede modificar esta definición.

4.3 Archivo XML personalizable

El explorador de productos se puede configurar cambiando la descripción de la interfaz que se describe en el archivo [Lectra.PDM.Search.Product.xml](#). Es necesario completar el archivo que se encuentra en la subcarpeta **custom namespaces/custom/Lectra.PDM.Search.Product.xml** (y no **namespaces/Lectra.PDM.Search.Product.xml**).

Si no existe, es necesario crearlo en la carpeta custom y estructurarlo del siguiente modo:

```
<namespace uri="http://lectra.com/pdm/search/product/custom"
  prefix="searchproduct"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:common="http://lectra.com/pdm/common">

<!-- AÑADIR AQUÍ LA CONFIGURACIÓN -->

<!-- _____ -->
<explorergrid name="resultGrid_XXX" inherits="resultGrid">
  <!-- XXX : categoryName (Style, Fabric, etc ....) -->
  <!-- Columns description -->
  <!-- securityCells : define the tagNames tagName1, tagName1 etc
  -->
  <!-- tagName:column Name can be controlled before render
  explorer -->
  <colModel inherits="resultGrid/colModel"
    securityColumns="colorTag,supplierTag,descTag"
  securityBranch="Explorer" securityType="{#categoryName}"
  colorTag="colors,sizes"
  supplierTag="suppliers,seasonsClassif"
  descTag="description,codeAlpha1" />
  <store xsi:type="lpfdirectstore" inherits="resultGrid/store" />
</explorergrid>

</namespace>
```

4.4 Estructura del panel de búsqueda en el archivo XML

namespaces/Lectra.PDM.Search.Product.xml

El panel **Search (Búsqueda)** viene definido por:

- un explorador `<explorerform name="SearchForm"` a su vez definido por:
- el panel `<panel name="accordionPanel" id="myaccordionPanel"`

Cada grupo de criterios (es decir: grupo **"Description"**, que contiene el criterio **"Reference Version (Versión de referencia)"**) está definido por un panel. El contenido de cada **"panel criteria group"** se define en el archivo y después se referencia en la definición del panel **"accordionPanel"**.

Ejemplo: Para el grupo del criterio "Classification":

La descripción del panel "classification" se realiza del siguiente modo:

```

<!-- ***** -->
<!--      Collection block definition (Classification)      -->
<!-- ***** -->
<panel name="classification" titleI18n="ProductGO.collection" collapsible="true">
  <nodecombo fieldLabelI18n="{#axeRoleName}" hiddenName="{#classifRoleName}.values" forceSelection="true"
    multiSelect="true" propertyName="{#propertyName}" genProductClassif="product"
    chkVisibleExpr="{#propertyName eq 'Season'} or {#propertyName eq 'Division'} or {#propertyName eq 'Theme'} or

  <!-- Private field(s) : do not edit -->
  <customSerializer type="com.lectra.pdm.lpf.ext.serializers.ProductClassifFromCriteriaGeneratorCustomSerializer" />
  <!-- End of private field(s)      -->
</panel>

```

Su inclusión en el "AccordionPanel" se realiza del siguiente modo:

```

<!-- ***** -->
<!--      PANEL : Accordion Panel      -->
<!-- ***** -->
<panel name="accordionPanel" id="myaccordionPanel" layout="accordionform" activeItem="0"
  autoScroll="true" border="false" isCriteriaPanel="true" style='border-top: 1px solid #fff'>
  <defaults style="border-top: 1px solid #000"/>
  <plugin xsi:type="xmlmap" ptype="panelchildrenvisibility"/>

  <!-- ***** -->
  <!--      PANEL : description      -->
  <!-- ***** -->
  <panel name="description" inherits="description_{#explorerName},description_{#topCategoryName},description" layout="form"
    border="false" labelWidth="65" labelAlign="left" autoScroll="true" bodyStyle="padding: 14px 20px 10px 20px">
    <defaults anchor="100%" />
  </panel>
  <panel name="classification" inherits="classification_{#explorerName},classification_{#topCategoryName},classification"
    layout="form" border="false" labelWidth="65" labelAlign="left" autoScroll="true" bodyStyle="padding: 14px 20px 10px 20px">
    <defaults anchor="100%" />
  </panel>
  <panel name="axis" inherits="axis_{#explorerName},axis_{#topCategoryName},axis" layout="form"
    border="false" labelWidth="65" labelAlign="left" autoScroll="true" bodyStyle="padding: 14px 20px 10px 20px">
    <defaults anchor="100%" />
  </panel>
  <panel name="mainProduct" inherits="mainProduct_{#explorerName},mainProduct_{#topCategoryName},mainProduct" condition="{#is
    border="false" labelWidth="65" labelAlign="left" autoScroll="true" bodyStyle="padding: 14px 20px 10px 20px">
    <defaults anchor="100%" />
  </panel>
  <panel name="workflow" inherits="workflow_{#explorerName},workflow_{#topCategoryName},workflow" condition="{#isPickerExplor
    border="false" labelWidth="65" labelAlign="left" autoScroll="true" bodyStyle="padding: 14px 20px 10px 20px">
    <defaults anchor="100%" />
  </panel>
  <panel name="task" inherits="task_{#explorerName},task_{#topCategoryName},task" condition="{#isPickerExplorer}" layout="for
    border="false" labelWidth="65" labelAlign="left" autoScroll="true" bodyStyle="padding: 14px 20px 10px 20px">
    <defaults anchor="100%" />
  </panel>
</panel>
<!-- ***** -->
<!--      END PANEL Accordion      -->
<!-- ***** -->

```

4.5 Identificar la parte que se va a duplicar y modificar

Para cada modificación:

- Identifique la parte correspondiente en el archivo [//namespaces/LectraPDM.Search.Product.xml](#) (xml de base).
- Copie el <panel> que contiene esta parte en el archivo [/namespaces/custom/LectraPDM. Search.Product.xml](#) (custom xml)
- Introduzca la modificación en esta parte duplicada.
- Actualice la presentación: utilice la tecla **F5** para actualizar la pantalla.

4.6 Añadir un criterio a un grupo existente del panel de búsqueda

Por ejemplo, para añadir un criterio al grupo "Description" como en el ejemplo anterior:

4.6.1 Identificación del bloque de descripción del panel de "Description" en el archivo XML de base

```
<!-- ***** -->
<!--           Description block definition           -->
<!-- ***** -->
<panel name="description" titleI18n="ProductGO.tree.generalObjectives"
  collapsible="true">
  <textfield name="codeAlpha1.values" fieldLabelI18n="codeAlpha1"
    chkVisible="true" />
  <textfield name="codeAlpha2.values" fieldLabelI18n="codeAlpha2"
    chkVisible="true" />
  <textarea name="description.values" fieldLabelI18n="description"
    chkVisible="true" />
  <twincombo fieldLabelI18n="RefVersion" hiddenName="refVersion.values"
    ignoreFormClear="true" inherits="http://lectra.com/pdm/common#yesno"
    conditionCtrl="com.lectra.lpf.uimodel.FeatureFilter" />
</panel>
```

4.6.2 Duplicación de este bloque en el archivo XML personalizado

4.6.3 Añadir los campos que se quiera en este bloque

Ejemplo 1: Añadir una cadena de caracteres para un campo Custom_Notes (Notas personales) creado en el módulo de Administración y configuración:

Añada la siguiente línea al principio del bloque:

```
<textarea name="Custom_Notes.values" fieldLabelI18n="Custom_Notes"
  chkVisible="true" />
```

Ejemplo 2: Añadir una lista de países creada en el módulo de Administración y configuración:

Añada la siguiente línea al principio del bloque:

```
<nodecombo fieldLabelI18n="Countries" propertyFamily="process"
  propertyName="Country" multiselect="false" chkVisible="false" />
```

4.7 Añadir un grupo de criterios en el panel de búsqueda

A la descripción de **AccordionPanel** de la sección 4.4 - [Estructura del panel de búsqueda en el archivo XML del archivo namespaces/Lectra.PDM.Search.Product.xml](#), añadimos un nuevo grupo compuesto de varios campos creados en el módulo de Administración y configuración.

4.7.1 Identificación del bloque de descripción "accordionPanel" en el archivo XML de base

```

<!-- ***** -->
<!-- PANEL : Accordion Panel -->
<!-- ***** -->
<panel name="accordionPanel" id="myaccordionPanel" layout="accordionform" activeItem="0"
  autoScroll="true" border="false" isCriteriaPanel="true" style="border-top: 1px solid #fff">
  <defaults style="border-top: 1px solid #000"/>
  <plugin xsi:type="xmlmap" ptype="panelchildrenvisibility"/>

  <!-- ***** -->
  <!-- PANEL : description -->
  <!-- ***** -->
  <panel name="description" inherits="description_${explorerName},description_${topCategoryName},description" layout="form"
    border="false" labelWidth="65" labelAlign="left" autoScroll="true" bodyStyle="padding: 14px 20px 10px 20px">
    <defaults anchor="100%" />
  </panel>
  <panel name="classification" inherits="classification_${explorerName},classification_${topCategoryName},classification"
    layout="form" border="false" labelWidth="65" labelAlign="left" autoScroll="true" bodyStyle="padding: 14px 20px 10px 20px">
    <defaults anchor="100%" />
  </panel>
  <panel name="axis" inherits="axis_${explorerName},axis_${topCategoryName},axis" layout="form"
    border="false" labelWidth="65" labelAlign="left" autoScroll="true" bodyStyle="padding: 14px 20px 10px 20px">
    <defaults anchor="100%" />
  </panel>
  <panel name="mainProduct" inherits="mainProduct_${explorerName},mainProduct_${topCategoryName},mainProduct" condition="$(!#is
    border="false" labelWidth="65" labelAlign="left" autoScroll="true" bodyStyle="padding: 14px 20px 10px 20px">
    <defaults anchor="100%" />
  </panel>
  <panel name="workflow" inherits="workflow_${explorerName},workflow_${topCategoryName},workflow" condition="$(!#isPickerExplor
    border="false" labelWidth="65" labelAlign="left" autoScroll="true" bodyStyle="padding: 14px 20px 10px 20px">
    <defaults anchor="100%" />
  </panel>
  <panel name="task" inherits="task_${explorerName},task_${topCategoryName},task" condition="$(!#isPickerExplorer)" layout="form"
    border="false" labelWidth="65" labelAlign="left" autoScroll="true" bodyStyle="padding: 14px 20px 10px 20px">
    <defaults anchor="100%" />
  </panel>
</panel>
<!-- ***** -->
<!-- END PANEL Accordion -->
<!-- ***** -->

```

4.7.2 Duplicación de este bloque en el archivo XML personalizado

4.7.3 Añadir la descripción de este nuevo grupo en este bloque

```

<panel name="CustomGroup" inherits="description_${explorerName},
  description_${topCategoryName}, CustomGroup" layout="form"
  border="false" labelWidth="65" labelAlign="left" autoScroll="true"
  bodyStyle="padding: 14px 20px 10px 20px">
  <defaults anchor="100%" />
</panel>


```

Añadir este bloque con el nombre de grupo correcto en el lugar deseado, debajo del comentario "PANEL: description".

4.7.4 Definir estos campos como criterios de búsqueda

Los campos añadidos en este panel de búsqueda de la pantalla Productos (cualquiera que sea la categoría) son campos específicos. No sólo muestran sus valores (como en la pantalla **GeneralObjectives**), estos campos facilitan la búsqueda de productos. Estos campos deben identificarse como criterios de búsqueda en el módulo de Administración y configuración.

Esta definición se realiza en un archivo de Excel con una estructura establecida y que debe introducirse correctamente. Este archivo debe importarse en el módulo de Administración y configuración.

 Consulte la ayuda en línea del módulo de Administración y configuración, apartado 13: **Criterios de búsqueda**
 También puede consultar el recordatorio del [Anexo H - Cambio de ubicación del directorio de LPFExt](#)

En PLM V5R1, los archivos LPFExt se encuentran en lpf/ext3/nombre del directorio.

Recordatorio de la aplicación de Administración y configuración

4.8 Añadir una columna a la cuadrícula de resultados

4.8.1 Identificar la descripción de la cuadrícula de resultados

Identifique la parte correspondiente en el archivo namespaces/Lectra.PDM.Search.Product.xml.

```
<!-- ***** -->
<!-- Grid definition (result): explorerGrid (colModel and store) -->
<!-- ***** -->
<explorereditablegrid name="resultGrid" explorerSubType="Product"
  explorerType="{#categoryName}" lockColumn="true"
  forceValidation="true" multiCell="true"
  stateId="ExplorerProduct_grid" stateful="true"
  tempCreationId="codeNum2" singleSelectExpr="{#singleSelect}"
  quickCreateMode="false" >

  .../ ...

</explorereditablegrid>
```

Esta descripción <explorereditablegrid> contiene 2 partes que hay que modificar:

- La lista de las columnas de la cuadrícula:

```
<!-- Columns description -->
<colModel> .../... </colModel>
```

- El "Store (Almacén)" de los datos a recopilar en la base de datos para rellenar la cuadrícula:

```
<!-- Data description -->
<!-- Advice : Do not edit to keep all data definition -->
<store xsi:type="lpfdirectstore" remoteSort="true"
  forceUpdateRecordsOnFail="true">
  .../...
</store>
```

4.8.2 Duplicación de este bloque en el archivo XML personalizado

Si no conoce el significado de la sección 2.2 - Herencia, duplique todo el bloque <explorereditablegrid> en el archivo personalizado namespaces/custom/Lectra.PDM.Search.Product.xml y añada las 2 líneas necesarias al añadir la columna.

4.8.3 Añadir la descripción de una columna de tipo texto

Para añadir una columna que inserta un tipo de datos simples:

Añada la columna.

En `<colModel>`, añada por ejemplo:

Para una de texto:

```
<column      xsi:type="textcolumn"      headerI18n="TEXT_NAME"  
dataIndex="TEXT_NAME_INDEX" />
```

Para la fecha, si desea que la fecha cambie según el huso horario del cliente:

```
<column      xsi:type="datecolumn"      headerI18n="TEXT_NAME"  
dataIndex="DATE_INDEX" />
```

Para la fecha, si no desea que la fecha cambie según el huso horario del cliente:

```
<column      xsi:type="datecolumn"      headerI18n="TEXT_NAME"  
dataIndex="DATE_INDEX" useServerTimezone="true" />
```

1. Añada la referencia al "Store (Almacén)"

En el bloque `<store>`, añada por ejemplo:

```
<dataField name="TEXT_NAME_INDEX" />  
<dataField name=" DATE_INDEX" />
```



También puede consultar el [Anexo Erreur ! Source du renvoi introuvable. - Erreur ! Source du renvoi introuvable.](#)

4.8.4 Store (Almacén)

Para que la cuadrícula se complete correctamente con los valores requeridos debe actualizarse el store (almacén). Para la actualización, en la descripción del Store (Almacén) debe hacerse referencia mediante una etiqueta `<dataField>` a todos los tipos de datos que hay en la cuadrícula. Por consiguiente, en el marco del proceso de recuperación del "producto" actual y sus atributos, sólo se extraerán los valores necesarios de los atributos del "producto".

Hay 3 modos de definir un nivel de filtro adicional para cada atributo necesario para la presentación. Con el modo = "full", "shallow" o "strict", los datos se ordenan en orden descendente.

mode="full": total. Se carga desde la base de datos todo el objeto (incluidos sus atributos).

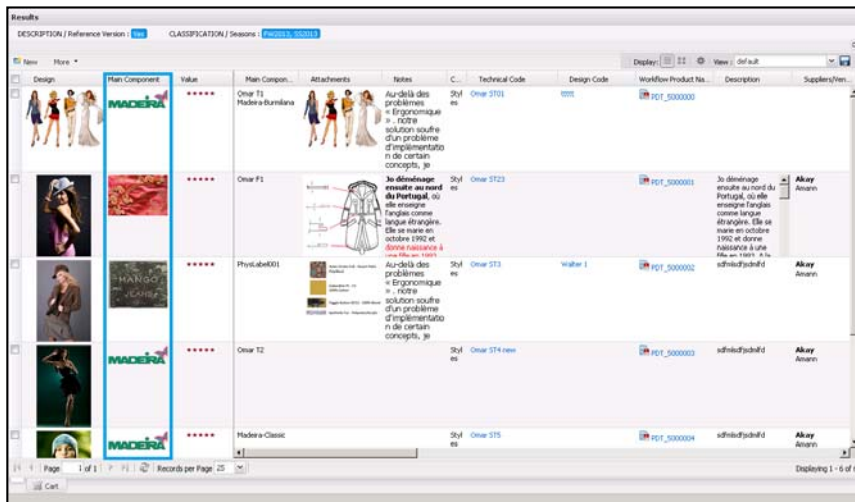
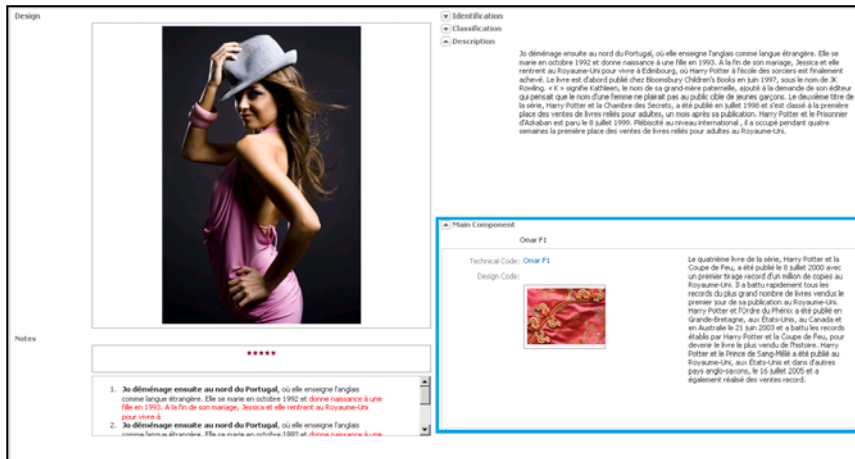
mode="shallow": parcial. Se recuperan de la base de datos los atributos simples del objeto.

mode="strict": mínimo. Sólo se recupera el nombre.

4.8.5 Añadir columnas a los campos personalizados

A continuación figura un ejemplo de configuración:

Objetivo: Vincular el producto al Style GO e insertarlo en la cuadrícula de búsqueda.



Paso 1 : Requisitos del módulo de Administración y configuración

Definición de un "Custom Role" con "Product" "Target" en la entidad "Style". En el ejemplo se denomina "mainComponent".

Paso 2: Añadir el panel "mainComponent" a la pantalla GeneralObjectives.

Archivo `LectraPLMParam\namespaces\custom\Lectra.PDM.ProductGO.xml`

```
<panel name="mainComponent" titleI18n="mainComponent" columnWidth="1.0"
border="false" collapsible="true" >
  <productcombo hiddenName="mainComponent" openerType="{#type}" />
  <panel columnWidth="1." layout="column" >
    <panel columnWidth="0.4" border="false">
      <renderfield name="mainComponent" fieldLabelI18n="technicalCode"
listenToProductCombo="mainComponent">
      <renderFn name=
" Lectra.PDM.Format.productLinkRenderer('codeAlpha1') " />
    </renderfield>
      <renderfield name="mainComponent" fieldLabelI18n="studyCode"
listenToProductCombo="mainComponent">
      <renderFn name=" Lectra.PDM.Format.productLinkRender
('codeAlpha2') " />
    </renderfield>
  </panel>
</panel>
```

```

        <imagefield name="mainComponent.defaultImageField"
            imageSize="large" readOnly="true" submitValue="false"
            listenToProductCombo="mainComponent" >
            <managerInstance name="Lectra.PDM.Mgrs.imageFieldMgr" />
        </imagefield>
    </panel>
    <panel columnWidth="0.6" border="false" >
        <textarea name="mainComponent.description" readOnly="true"
            submitValue="false" listenToProductCombo="mainComponent" />
    </panel>
</panel>

```

Paso 3: Añadir la columna a la cuadrícula de resultados del explorador de productos

Archivo LectraPLMParam\namespaces\custom\Lectra.PDM.Search.Product.xml

```

<!-- Grid definition (result) : explorerGrid (colModel and store)-->
<explorergrid name="resultGrid" explorerSubType="Product"
    explorerType="{#categoryName}" lockColumn="true"
    forceValidation="true" multiCell="true"
    stateId="ExplorerProduct_grid" stateful="true"> <!--
    autoLoadOnRender="true">-->

    <!-- Columns description -->
    <colModel>
        .../...
        <column xsi:type="productcolumn" headerI18n="mainComponent"
            dataIndex="mainComponent" subDataIndex="defaultImageField"
            editable="true" />
        <column xsi:type="imagecolumn" headerI18n="mainComponent"
            dataIndex="mainComponent" subDataIndex="defaultImageField" />
        .../....
    </colModel>
    <!-- Data description -->
    <!-- Advice : Do not edit to keep all data definition -->
    <store xsi:type="lpfdirectstore" remoteSort="true"
        forceUpdateRecordsOnFail="true">
        .../...
        <dataField name="mainComponent" mode="shallow">
            <filter name="defaultImageField" mode="full" />
        </dataField>
        .../...
    </store>
    .../...
</explorergrid>

```

Código que añadir

5. CONFIGURACION AÑADIENDO CAMPOS AL PLAN DE COLECCION

5.1 Requisitos del módulo de Administración y configuración

Los campos personalizados que se van a añadir a esta pantalla deben definirse en el módulo de Administración y configuración, como se indica en el [Anexo H - Cambio de ubicación del directorio de LPFExt](#)

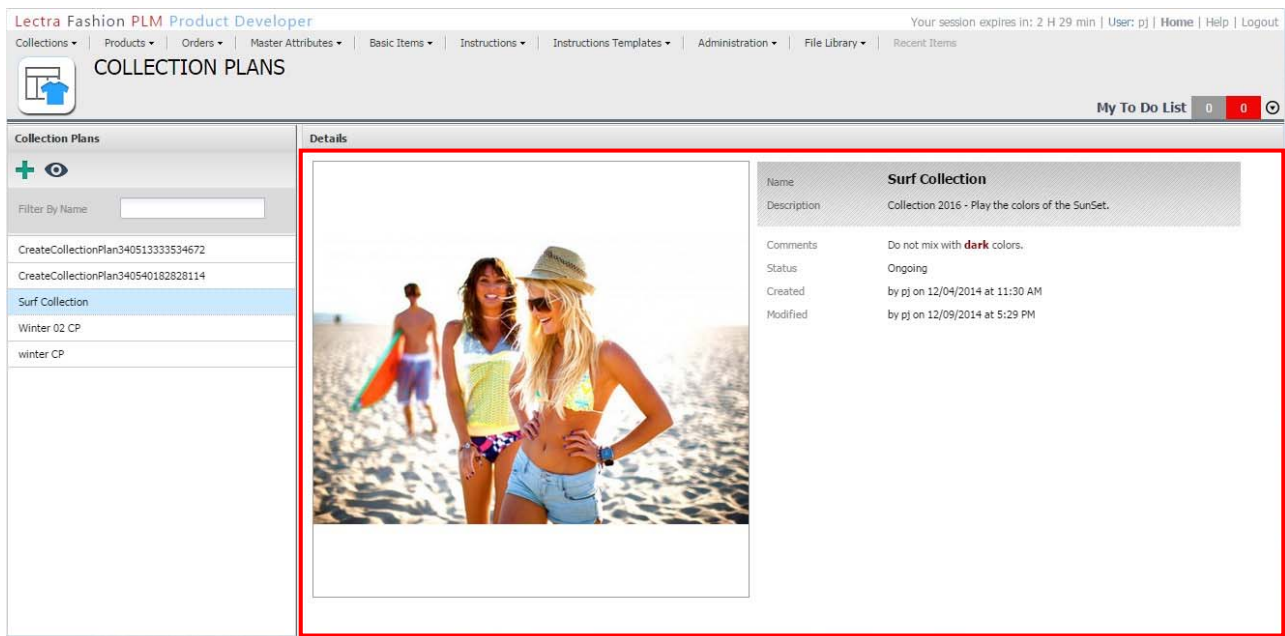
En PLM V5R1, los archivos LPFExt se encuentran en lpf/ext3/nombre del directorio.

Recordatorio de la aplicación de Administración y configuración, [Apéndice K - Añadir campos y funciones personalizados a los datos generales de un plan de colección](#), [Anexo L – Añadir campos y funciones personalizados a la cuadrícula Variantes de un plan de colección](#).

5.2 Presentación de las pantallas del plan de colección

5.2.1 Pantalla del plan de colección

5.2.1.1 Presentación de la pantalla



Surf Collection	
Name	Surf Collection
Description	Collection 2016 - Play the colors of the SunSet.
Comments	Do not mix with dark colors.
Status	Ongoing
Created	by pj on 12/04/2014 at 11:30 AM
Modified	by pj on 12/09/2014 at 5:29 PM

Esta pantalla tiene una única parte personalizable: el centro (**marco rojo**). Nota: la información es de "sólo lectura". Esta pantalla no admite el modo de edición.

5.2.1.2 Archivo XML personalizable

La pantalla de lista del plan de colección se puede personalizar modificando el archivo [Lectra.PDM.CollectionPlan.ListCollectionPlan.xml](#). Es necesario completar el archivo que se

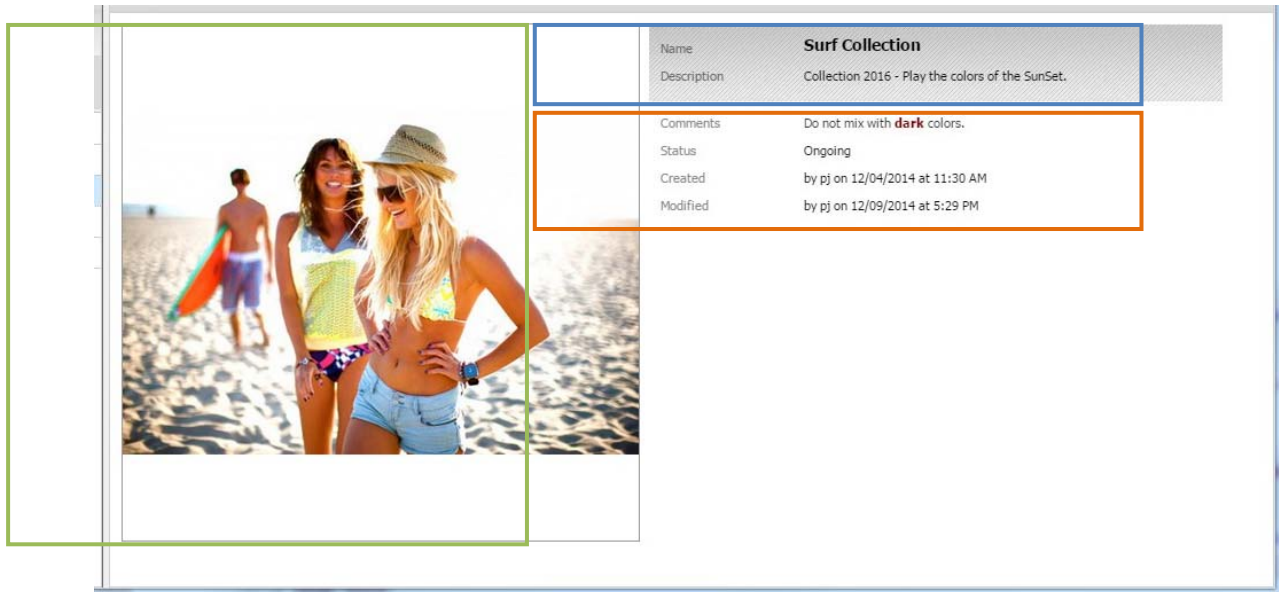
encuentra en la subcarpeta **custom namespaces/custom/**
[Lectra.PDM.CollectionPlan.ListCollectionPlan.xml](#) (y no en la original [<2756namespaces/](#)
[Lectra.PDM.CollectionPlan.ListCollectionPlan.xml](#)).

Si no existe, es necesario crearlo en la carpeta custom y estructurarlo del siguiente modo:

```
<namespace xmlns:xsi=http://www.w3.org/2001/XMLSchema-instance
uri=http://lectra.com/pdm/ListCollectionPlan/custom prefix="ListCollectionPlan">

  <fieldset name="ListCollectionDetailsDefault" border="false" hidden="true"
layout="column" padding="0">
  <fieldset inherits="ListCollectionDetails/ListCollectionDetailsLeft" >
    <!-- See Dictionary.xml for the available kinds of fields -->
  </fieldset>
  <fieldset inherits="ListCollectionDetails/ListCollectionDetailsRight" >
    <!-- See Dictionary.xml for the available kinds of fields -->
  </fieldset>
</fieldset>
</namespace>
```


5.2.1.3 Identificación de la parte que hay que modificar



La parte central tiene la siguiente estructura:

listCollectionDetails (fieldset)

 |_ listCollectionDetailsLeft (fieldset)

 |_ imagefield

 |_ listCollectionDetailsRight (fieldset)

 |_ no name (fieldset)

 |_ name (displayfield)

 |_ description (displayfield)

 |_ no name (fieldset)

 |_ comments (displayfield)

 |_ status (displayfield)

 |_ created (displayfield)

 |_ modified (displayfield)

5.2.1.4 Añadir campos debajo de imagefield de listCollectionDetailsLeft

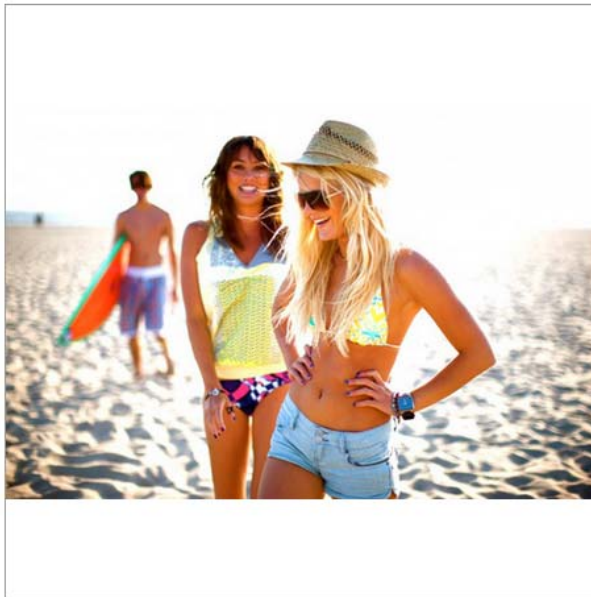
Añada los campos y funciones que haya creado en el módulo de Administración y configuración, en la etiqueta fieldset:

```
<fieldset inherits="listCollectionDetails/listCollectionDetailsLeft" >
  <!-- See Dictionary.xml for the available kinds of fields -->
</fieldset>
```

Y actualice su explorador para ver la modificación.

Ejemplo: casilla de verificación situada debajo de la imagen.

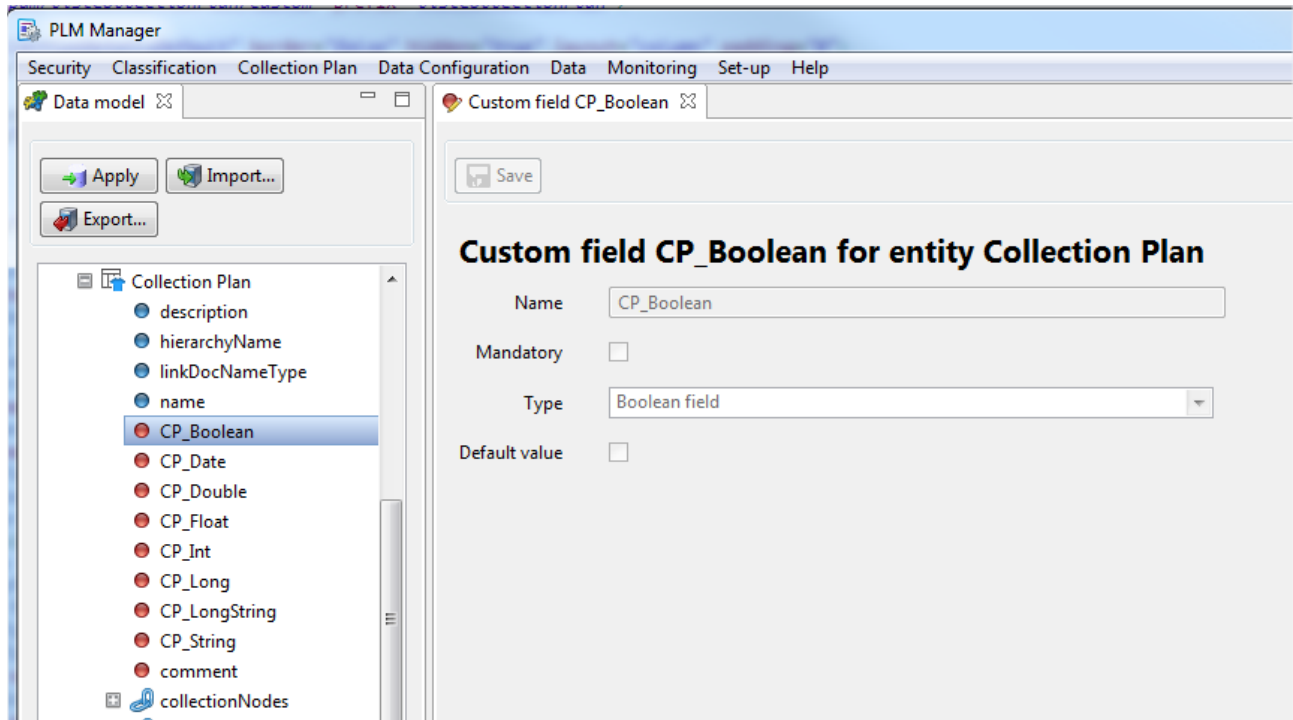
```
<fieldset inherits="listCollectionDetails/ListCollectionDetailsLeft" >  
  <!-- See Dictionary.xml for the available kinds of fields -->  
  <checkbox name="CP_Boolean" fieldLabelI18n="custom.cp.label.boolean" readOnly="true" />  
</fieldset>
```



Name	Surf Collection
Description	Collection 2016 - Play the colors of the SunSet.
Comments	Do not mix with dark colors.
Status	Ongoing
Created	by pj on 12/04/2014 at 11:30 AM
Modified	by pj on 12/09/2014 at 5:29 PM

To improve

Tenga en cuenta que el nombre de la casilla de verificación es idéntico al que se ha creado en el módulo de Administración y configuración:



5.2.1.5 Añadir campos hasta imagefield de listCollectionDetailsLeft

Si tiene que añadir campos y funciones hasta imagefield, ya no puede utilizar el mecanismo de herencia. La etiqueta fieldset debe redefinirse copiando la del archivo [namespaces/Lectra.PDM.CollectionPlan.ListCollectionPlan.xml](#) original:

```
<fieldset name="ListCollectionDetailsLeft" padding="10" border="false" margin="-4 0 0 0">
  <imagefield fieldLabelI18n="action.toggle.storyBoard" hideLabel="true" name="attachment" readOnly="true" useREST="true"
    inherits="http://Lectra.com/pdm/common#defaultImageFieldEXT5" managerRef="attachmentImageField"
    imageWidth="450" imageHeight="450" width="460" height="460" imageSize="full">
    <managerInstance name="Lectra.PDM.Mgrs.imageFieldRestMgr"/>
  </imagefield>
</fieldset>
```

Seguidamente, hay que copiar la etiqueta en el archivo personalizado [namespaces/custom/Lectra.PDM.CollectionPlan.ListCollectionPlan.xml](#), en lugar de la etiqueta fieldset heredada, y modificarla como corresponda.

Ejemplo: casilla de verificación situada encima de la imagen.

```
<fieldset name="ListCollectionDetailsLeft" padding="10" border="false" margin="-4 0 0 0">
  <checkbox name="CP_Boolean" fieldLabelI18n="custom.cp.label.boolean" readOnly="true" />
  <imagefield fieldLabelI18n="action.toggle.storyBoard" hideLabel="true" name="attachment" readOnly="true" useREST="true"
    inherits="http://Lectra.com/pdm/common#defaultImageFieldEXT5" managerRef="attachmentImageField"
    imageWidth="450" imageHeight="450" width="460" height="460" imageSize="full">
    <managerInstance name="Lectra.PDM.Mgrs.imageFieldRestMgr"/>
  </imagefield>
</fieldset>
```

To improve ✓



Name	Surf Collection
Description	Collection 2016 - Play the colors of the SunSet.
Comments	Do not mix with dark colors.
Status	Ongoing
Created	by pj on 12/04/2014 at 11:30 AM
Modified	by pj on 12/09/2014 at 5:29 PM

5.2.1.6 Añadir campos debajo de las etiquetas fieldset de listCollectionDetailsRight



Añada los campos y funciones que haya creado en el módulo de Administración y configuración, en la etiqueta fieldset:

```
<fieldset inherits="ListCollectionDetails/ListCollectionDetailsRight" >
    <!-- See Dictionary.xml for the available kinds of fields -->
</fieldset>
```

Actualice su explorador para ver la modificación.

Ejemplo: debajo de las etiquetas fieldset aparecen un área de texto, una casilla de verificación, un número entero, una fecha, una combinación múltiple, una combinación única, una imagen y una combinación envitem.

```
<fieldset inherits="ListCollectionDetails/ListCollectionDetailsRight" >
    <!-- See Dictionary.xml for the available kinds of fields -->
    <fieldset padding="10" border="false" margin="-3 0 0 0">
        <textarea name="CP_LongString" fieldLabelI18n="custom.cp.Label.Longstring" readOnly="true" />
        <checkbox name="CP_Boolean" fieldLabelI18n="custom.cp.Label.boolean" readOnly="true" />
        <integerfield name="CP_Int" fieldLabelI18n="custom.cp.Label.int" readOnly="true" />
        <datefield name="CP_Date" fieldLabelI18n="custom.cp.Label.date" readOnly="true" />
        <nodecombo name="CP_Country_Multi" multiSelect="true" propertyName="Country" fieldLabelI18n="custom.cp.Label.hvl.country.multi" >
            <bind readOnly="true" />
        </nodecombo>
        <nodecombo name="CP_Country_Single" multiSelect="false" propertyName="Country" fieldLabelI18n="custom.cp.Label.hvl.country.single" >
            <bind readOnly="true" />
        </nodecombo>
        <imagefield name="CP_Doc" fieldLabelI18n="custom.cp.Label.document" useREST="true"
            inherits="http://Lectra.com/pdm/common/defaultImageFieldEXT5" managerRef="selSmallImageField" imageWidth="96" imageHeight="96" readOnly="true" >
            <managerInstance name="Lectra.PDM.Mgrs.ImageFieldRestMgr" />
        </imagefield>
        <envitemcombo name="CP_Color" envItemType="MarketingColor" fieldLabelI18n="custom.cp.Label.envitem.color" hideTrigger2="true" readOnly="true" />
    </fieldset>
</fieldset>
```

	Name	Surf Collection
	Description	Collection 2016 - Play the colors of the SunSet.
	Comments	Do not mix with dark colors.
	Status	Ongoing
	Created	by pj on 12/04/2014 at 11:30 AM
	Modified	by pj on 12/09/2014 at 5:29 PM
	Other comment	Do not forget to see with designers
	To improve	✓
	Prevision	45600
	End	04/08/15
Countries	Italie Espagne	
Original country	France	
Add idea		
Main color	YELLOW-1	

5.2.1.7 Añadir campos debajo de la primera etiqueta fieldset de listCollectionDetailsRight


Si tiene que añadir campos y funciones justo debajo de esta etiqueta fieldset, ya no puede utilizar el mecanismo de herencia. A continuación hay que redefinir la etiqueta fieldset, que abarca listCollectionDetailsRight, copiando la del archivo [namespaces/Lectra.PDM.CollectionPlan.ListCollectionPlan.xml](#) original:

```
<fieldset name="listCollectionDetailsRight" width="520" padding="10" border="false" margin="-1 0 0 -20">
  <fieldset padding="10" margin="0" border="false" cls="pdm-identitycard">
    <defaults labelSeparator=" " />
    <displayfield name="name" fieldLabelI18n="name" fieldCls="pdm-header-1" />
    <displayfield name="description" fieldLabelI18n="description" />
  </fieldset>
  <fieldset padding="10" border="false" margin="-3 0 0 0">
    <defaults labelSeparator=" " />
    <displayfield name="comment" fieldLabelI18n="comment" />
    <displayfield name="statusName" fieldLabelI18n="status" />
    <displayfield name="formatCreatedDateWithUser" fieldLabelI18n="Label.authoring.created.nosemicolon" />
    <displayfield name="formatModifiedDateWithUser" fieldLabelI18n="Label.authoring.modified.nosemicolon" />
  </fieldset>
</fieldset>
```

Seguidamente, hay que pegar la etiqueta en el archivo personalizado [namespaces/custom/Lectra.PDM.CollectionPlan.ListCollectionPlan.xml](#), en lugar de la etiqueta fieldset heredada, y modificarla como corresponda.

Ejemplo: casilla de verificación situada después del campo de descripción.

```
<fieldset name="listCollectionDetailsRight" width="520" padding="10" border="false" margin="-1 0 0 -20">
  <fieldset padding="10" margin="0" border="false" cls="pdm-identitycard">
    <defaults labelSeparator=" " />
    <displayfield name="name" fieldLabelI18n="name" fieldCls="pdm-header-1" />
    <displayfield name="description" fieldLabelI18n="description" />
    <checkbox name="CP_Boolean" fieldLabelI18n="custom.cp.label.boolean" readOnly="true" />
  </fieldset>
  <fieldset padding="10" border="false" margin="-3 0 0 0">
    <defaults labelSeparator=" " />
    <displayfield name="comment" fieldLabelI18n="comment" />
    <displayfield name="statusName" fieldLabelI18n="status" />
    <displayfield name="formatCreatedDateWithUser" fieldLabelI18n="Label.authoring.created.nosemicolon" />
    <displayfield name="formatModifiedDateWithUser" fieldLabelI18n="Label.authoring.modified.nosemicolon" />
  </fieldset>
</fieldset>
```

	Name Surf Collection
	Description Collection 2016 - Play the colors of the SunSet.
	To improve ✓
	Comments Do not mix with dark colors.
	Status Ongoing
	Created by pj on 12/04/2014 at 11:30 AM
	Modified by pj on 12/09/2014 at 5:29 PM

5.2.1.8 Añadir campos y cambiar la etiqueta global fieldset listCollectionDetails

Si desea añadir campos y vínculos, para disponer de otra representación de diseño, consulte la sección Modificación del diseño de página.

5.2.1.9 Observaciones relativas a los campos y vínculos de esta pantalla

A continuación figuran varios campos y vínculos (con ejemplos) que se pueden añadir a esta pantalla y que son de sólo lectura:

Campo de cadena

```
<textfield name="CP_String" fieldLabel18n="custom.cp.label.string" readOnly="true" />
```

LongString field (Campo de cadena larga)

```
<textarea name="CP_LongString" fieldLabel18n="custom.cp.label.longstring" readOnly="true" />
```

Boolean field (Campo booleano)

```
<checkbox name="CP_Boolean" fieldLabel18n="custom.cp.label.boolean" readOnly="true" />
```

Long field (Campo largo)

```
<longfield name="CP_Long" fieldLabel18n="custom.cp.label.long" readOnly="true" />
```

Integer field (Campo de entero)

```
<integerfield name="CP_Int" fieldLabel18n="custom.cp.label.int" readOnly="true" />
```

Double field (Campo doble)

```
<doublefield name="CP_Double" fieldLabel18n="custom.cp.label.double" readOnly="true" />
```

Float field (Campo flotante)

```
<floatfield name="CP_Float" fieldLabel18n="custom.cp.label.float" readOnly="true" />
```

Date field (Campo de fecha)


```
<datefield name="CP_Date" fieldLabel18n="custom.cp.label.date" readOnly="true" />
```

Ejemplo de HVL pick list (lista de selección HVL) con Países y Selección múltiple.

```
<nodecombo name="CP_Country_Multi" multiSelect="true" propertyName="Country"
fieldLabel18n="custom.cp.label.hvl.multi" >
<bind readOnly="true" />
</nodecombo>
```

Ejemplo de HVL pick list (lista de selección HVL) con Países y Selección única.

```
<nodecombo name="CP_Country_Single" multiSelect="false" propertyName="Country"
fieldLabel18n="custom.cp.label.hvl.single" >
<bind readOnly="true" />
</nodecombo>
```

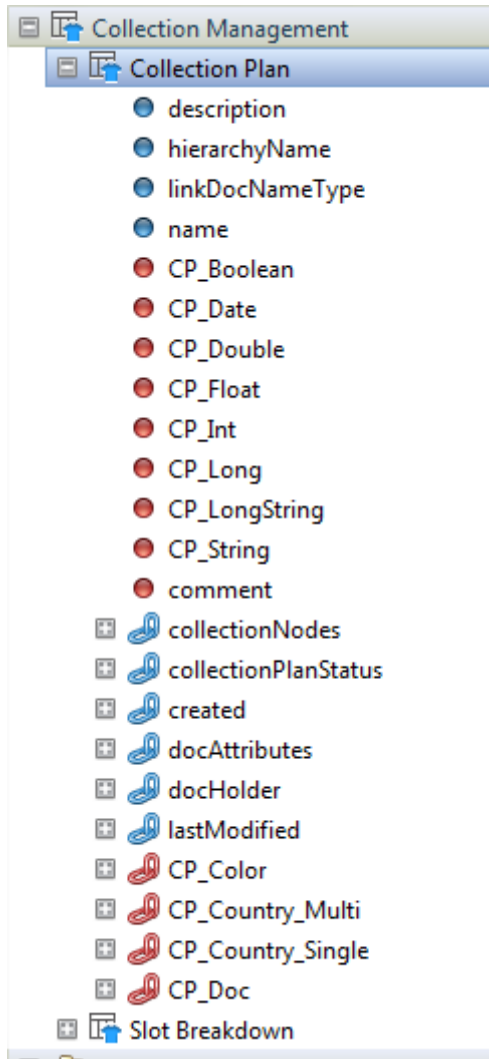
Selección del documento de destino

```
<imagefield name="CP_Doc" fieldLabel18n="custom.cp.label.document" useREST="true"
inherits="http://lectra.com/pdm/common#defaultImageFieldEXT5"
managerRef="selSmallImageField" imageWidth="96" imageHeight="96" readOnly="true" >
<managerInstance name="Lectra.PDM.Mgrs.imageFieldRestMgr" />
</imagefield>
```

Envitem (selección múltiple no permitida): ejemplo con Color de marketing

```
<envitemcombo name="CP_Color" envItemtype="MarketingColor"
fieldLabel18n="custom.cp.label.envitem.color" hideTrigger2="true" readOnly="true" />
```

Observe que el atributo del valor *name* es idéntico al que se ha creado en el módulo de Administración y configuración:



Tenga en cuenta que el atributo del valor *fieldLabel18n* debe definirse en el archivo [PLM-Fashion\PDM\LectraPLMParam\messagesCusto_en.properties](#) (y en otro archivo de idioma si es necesario, por ejemplo [messagesCusto_es.properties](#) para el español).

```

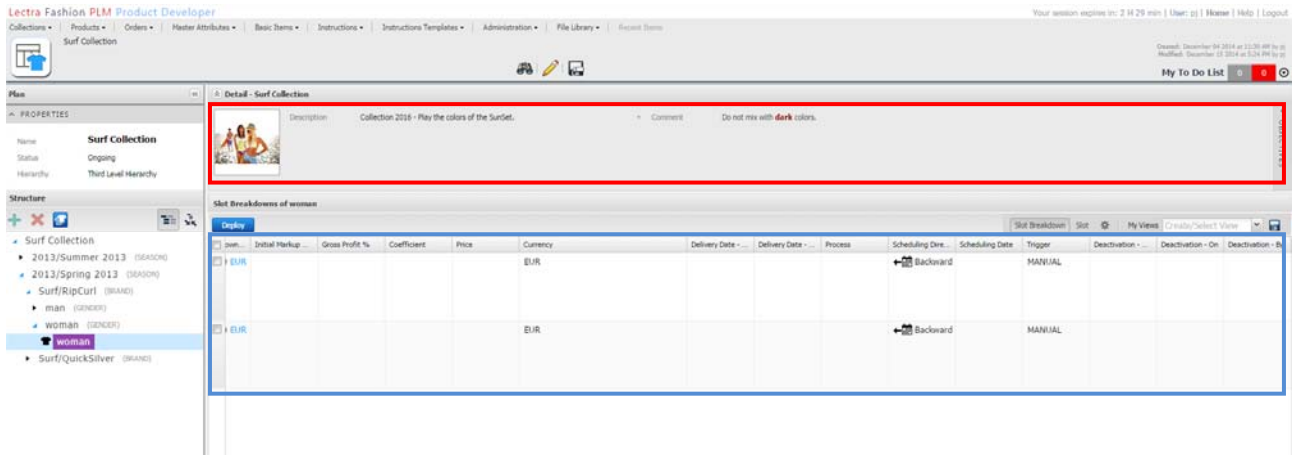
1 custom.cp.label.longstring=Other comment
2 custom.cp.label.boolean=To improve
3 custom.cp.label.int=Prevision
4 custom.cp.label.date=End
5 custom.cp.label.hvl.country.multi=Countries
6 custom.cp.label.hvl.country.single=Original country
7 custom.cp.label.document=Add idea
8 custom.cp.label.envitem.color=Main color
9 custom.cp.rp.label.date=Begining
10 custom.cp.rp.label.boolean=Stop
11 custom.cp.rp.label.string=Comment 2
12 custom.cp.rp.label.hvl.country.multi=Countries
13 custom.cp.rp.label.hvl.country.single=Country
14 custom.cp.rp.label.document=Recent image
15 custom.cp.rp.label.envitem.color=Main color
16 custom.cp.rp.label.envitem.unit=Unit

```

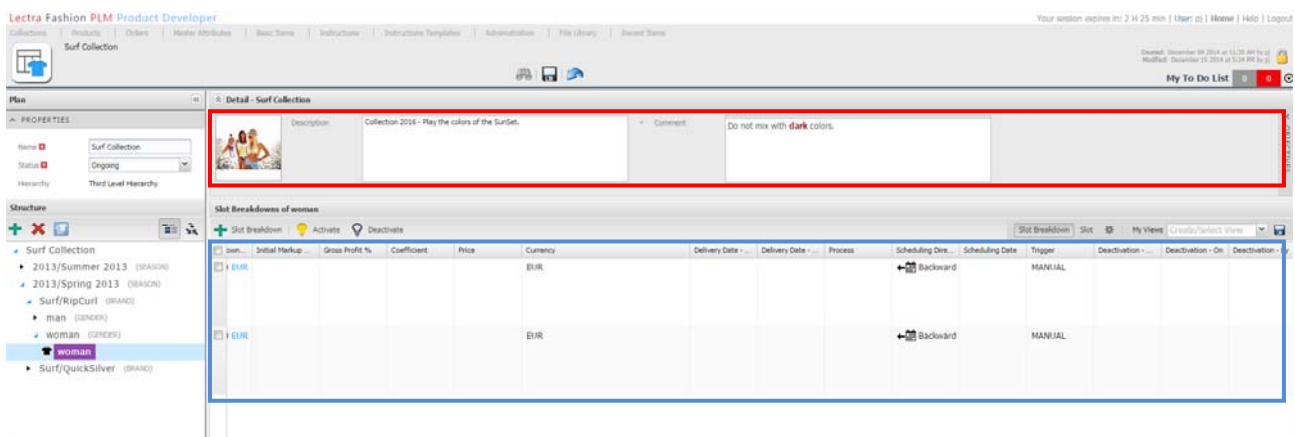

5.2.2 Detalle del plan de colección

5.2.2.1 Presentación de la pantalla

Modo de consulta



Modo de edición:



Esta pantalla tiene dos partes personalizables: los datos generales (**marco rojo**) y la cuadrícula de las variantes (**marco azul**). Tenga en cuenta que existen dos modos: consulta y edición.

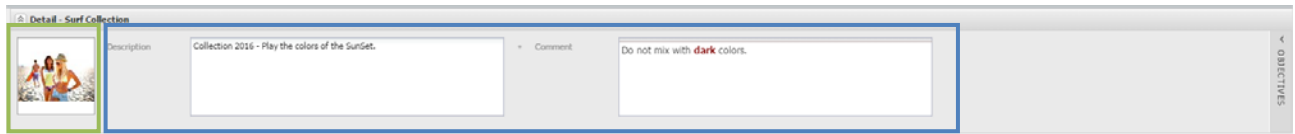
5.2.2.2 Archivo XML personalizable

La pantalla de lista del plan de colección se puede personalizar modificando el archivo [Lectra.PDM.CollectionPlan.CollectionPlan.xml](#). Es necesario completar el archivo que se encuentra en la subcarpeta **custom namespaces/custom/** [Lectra.PDM.CollectionPlan.CollectionPlan.xml](#) (y no en la original [namespaces/](#) [Lectra.PDM.CollectionPlan.CollectionPlan.xml](#)).

Si no existe, es necesario crearlo en la carpeta custom y estructurarlo del siguiente modo:

```
<namespace xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
uri="http://Lectra.com/pdm/colLectionPlan/custom"
    prefix="colLectionPlan">
    <!-- General Data Form - Display and Edit Mode -->
    <form name="detailsEditDefault" layout="border" id="editForm"
reference="collectionPlanDetailsForm" border="false">
    <panel inherits="detailsEdit/detailsEditWest" />
    <panel inherits="detailsEdit/detailsEditCenter" overflowX="auto" overflowY="auto"
layout="form">
        <!-- See Dictionary.xml for the available kinds of fields -->
    </panel>
</form>
    <!-- Grid Range Plan Slot Breakdown -->
    <grid name="rangePlanslotbreakdowngridDefault" inherits="rangePlanslotbreakdowngrid" >
        <!-- See Dictionary.xml for the available kinds of fields -->
    </grid>
</namespace>
```

5.2.2.3 Identificación de la parte que hay que modificar



La sección dedicada a los datos generales tiene la siguiente estructura:

detailsEdit (form)

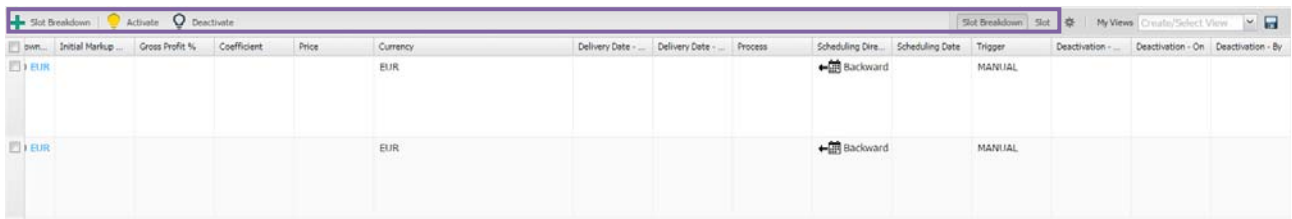
 |_ detailsEditWest (panel)

 |_ imagefield

 |_ detailsEditCenter (panel)

 |_ description (textarea)

 |_ comment (htmleditor)



Item	Initial Markup	Gross Profit %	Coefficient	Price	Currency	Delivery Date	Delivery Date	Process	Scheduling Dire.	Scheduling Date	Trigger	Deactivation	Deactivation - On	Deactivation - By
EUR					EUR				Backward		MANUAL			
EUR					EUR				Backward		MANUAL			

La parte dedicada a la cuadrícula de variantes tiene la siguiente estructura:

rangeplanslotbreakdowngrid (rangeplangrid)

 |_ no name (toolbar containing buttons)

 |_ (column 1)

 |_ (column 2)

 |_ etc.

5.2.2.4 Añadir campos debajo de imagefield de detailsEditWest (datos generales)

Añada los campos y funciones que haya creado en el módulo de Administración y configuración, en la etiqueta panel:

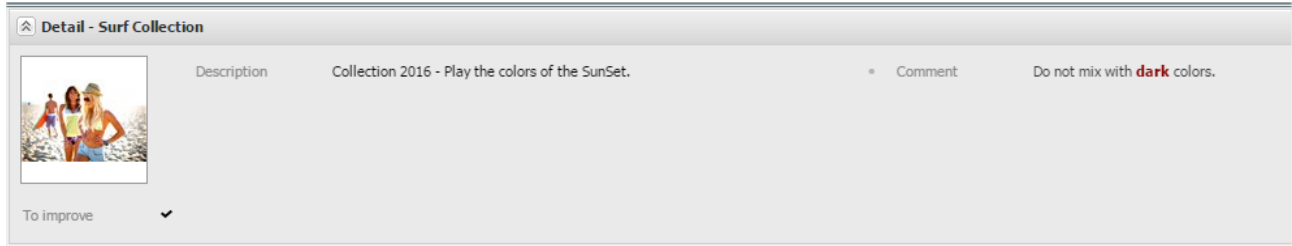
```
<panel inherits="detailsEdit/detailsEditWest" />
```

Y actualice su explorador para ver la modificación.

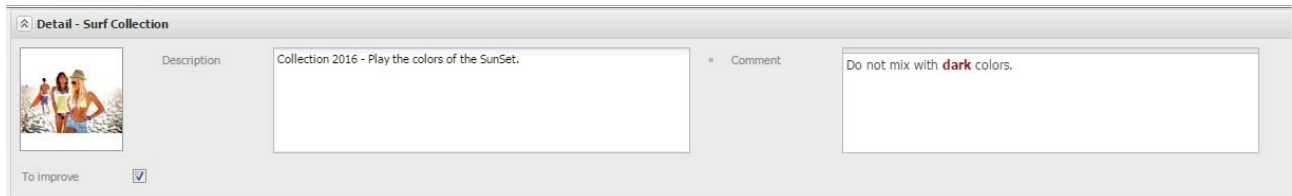
Ejemplo: casilla de verificación situada debajo de la imagen.

```
<panel inherits="detailsEdit/detailsEditWest" >
  <checkbox name="CP_Boolean" fieldLabelI18n="custom.cp.label.boolean" margin="10 0 0 10">
    <bind readOnly="{!isPageEditable}" value="{collectionPlan.CP_Boolean}" />
  </checkbox>
</panel>
```

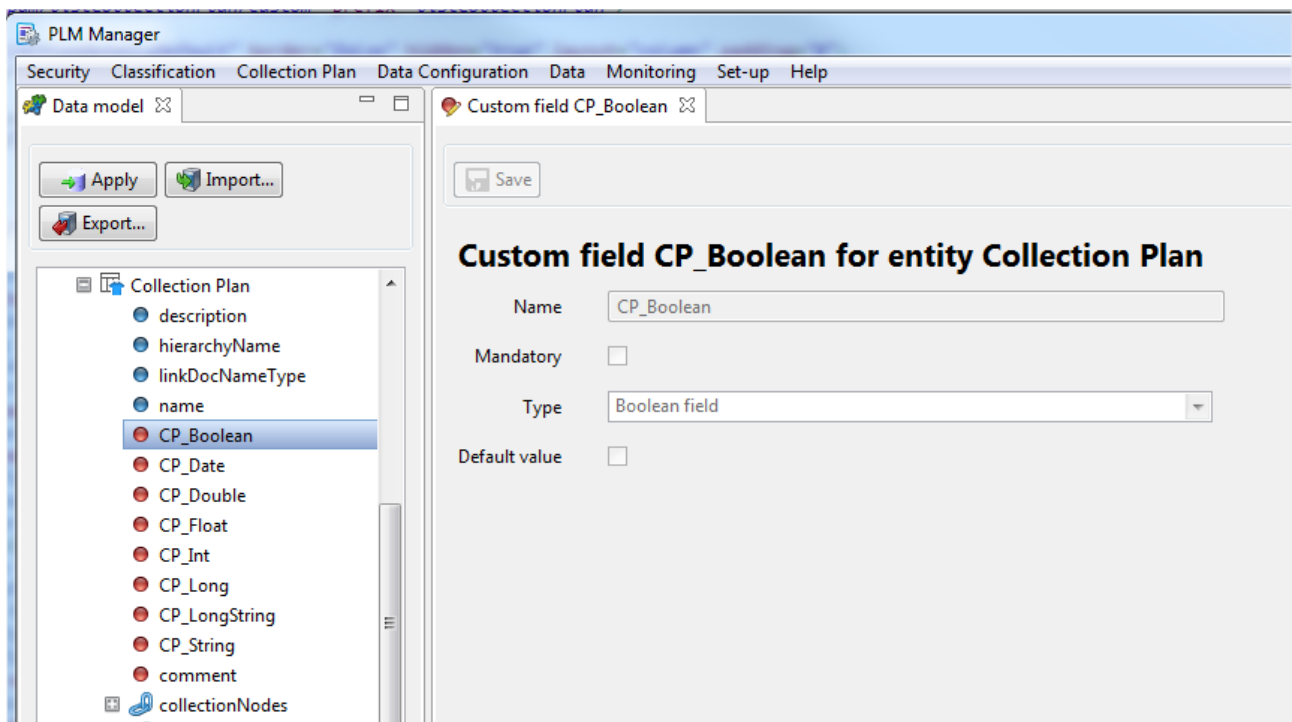
Modo de consulta



Modo de edición:



Nota: el nombre de la casilla de verificación es idéntico al que se ha creado en el módulo de Administración y configuración:



Observe que el atributo del valor *bind* es "{collectionPlan.created_name}". Debe utilizar este modelo.

Nota: el paso del modo de consulta al modo de edición, y viceversa, está condicionado por la etiqueta *bind* con el atributo *readOnly*. Debe utilizar este mecanismo.

5.2.2.5 Añadir campos hasta imagefield de "detailsEditWest" (datos generales)

Si tiene que añadir campos y funciones hasta imagefield, ya no puede utilizar el mecanismo de herencia. El panel debe redefinirse copiando el del archivo original [namespaces/Lectra.PDM.CollectionPlan.CollectionPlan.xml](#):

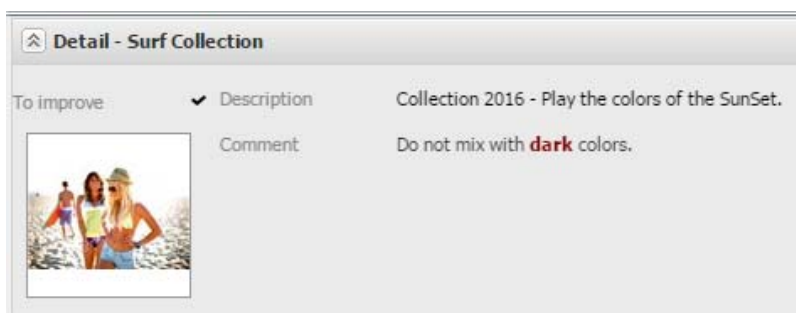
```
<panel name="detailsEditWest" border="false" region="west">
  <imagefield fieldLabelI18n="action.toggle.storyBoard" hideLabel="true" name="attachment"
    useREST="true" inherits="http://Lectra.com/pdm/common#defaultImageFieldEXT5" managerRef="selSmallImageField"
    imageSize="medium" imageWidth="96" imageHeight="96" padding="8 0 0 8" >
    <managerInstance name="Lectra.PDM.Mgrs.imageFieldRestMgr" />
    <bind fileLinkObj="{collectionPlan.attachment}" readOnly="{!isPageEditable}"/>
  </imagefield>
</panel>
```

Péguelo a continuación del archivo personalizado [namespaces/custom/Lectra.PDM.CollectionPlan.CollectionPlan.xml](#), en el lugar del panel heredado, y modifíquelo como corresponda.

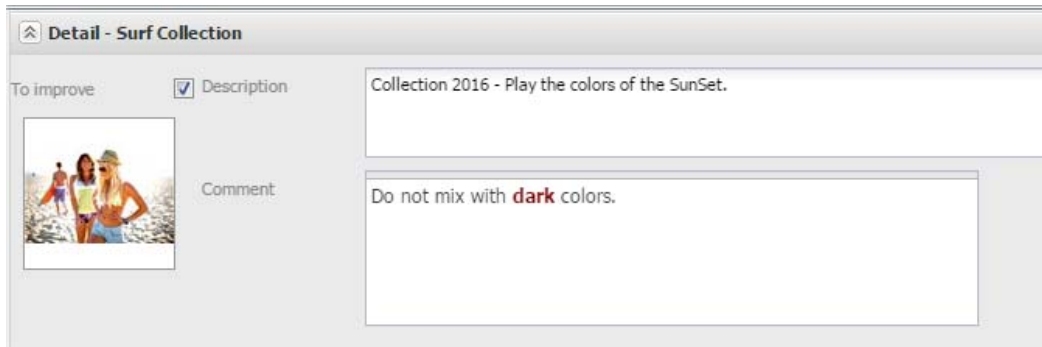
Ejemplo: casilla de verificación que aparece hasta la imagen.

```
<panel name="detailsEditWest" border="false" region="west">
  <checkbox name="CP_Boolean" fieldLabelI18n="custom.cp.label.boolean" margin="10 0 0 10">
    <bind readOnly="{!isPageEditable}" value="{collectionPlan.CP_Boolean}"/>
  </checkbox>
  <imagefield fieldLabelI18n="action.toggle.storyBoard" hideLabel="true" name="attachment"
    useREST="true" inherits="http://Lectra.com/pdm/common#defaultImageFieldEXT5" managerRef="selSmallImageField"
    imageSize="medium" imageWidth="96" imageHeight="96" padding="8 0 0 8" >
    <managerInstance name="Lectra.PDM.Mgrs.imageFieldRestMgr" />
    <bind fileLinkObj="{collectionPlan.attachment}" readOnly="{!isPageEditable}"/>
  </imagefield>
</panel>
```

Modo de consulta



Modo de edición:



5.2.2.6 Añadir campos debajo del comentario y la descripción de detailsEditCenter (datos generales)

Añada los campos y funciones que haya creado en el módulo de Administración y configuración, en la etiqueta panel:

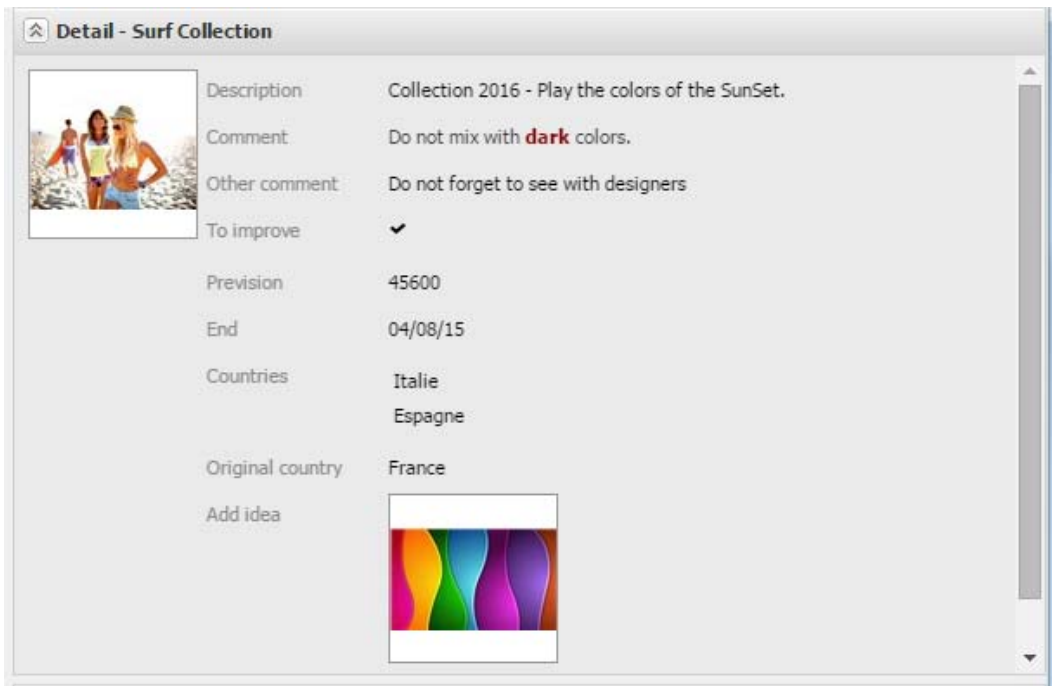
```
<panel inherits="detailsEdit/detailsEditCenter" overflowX="auto" overflowY="auto" layout="form">
```

Y actualice su explorador para ver la modificación.

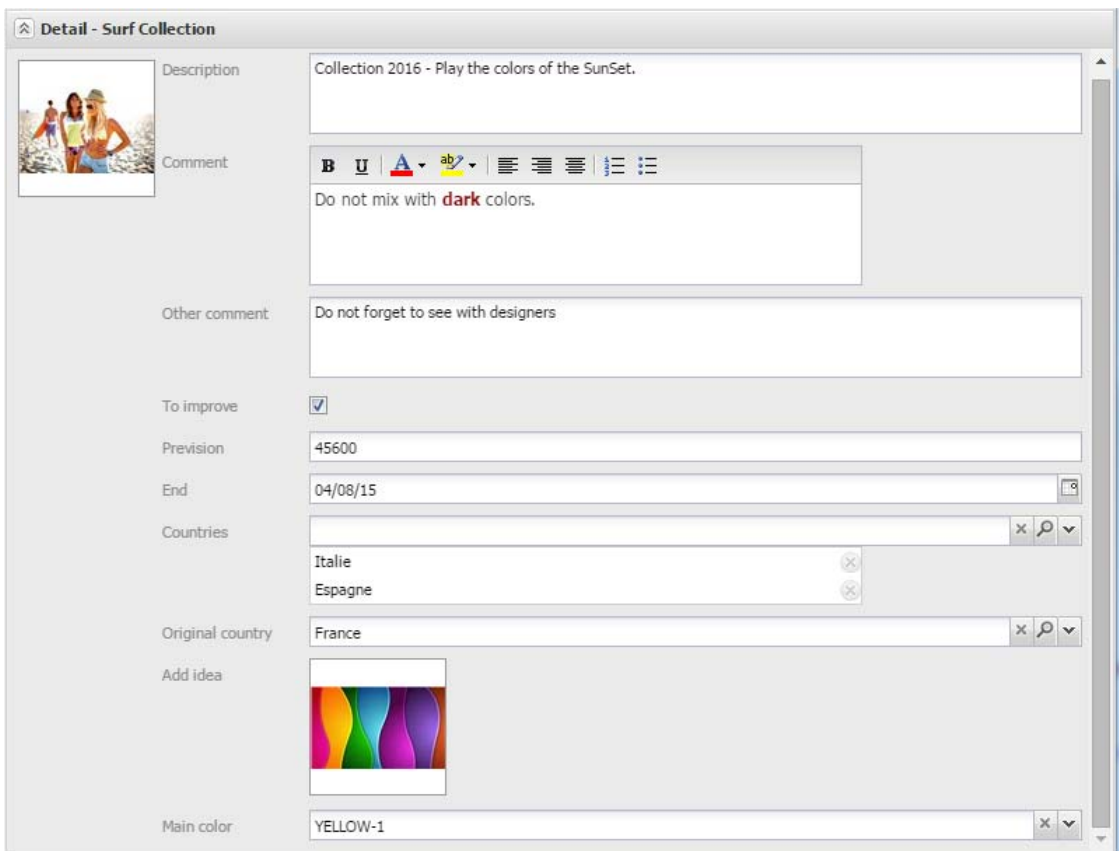
Ejemplo: debajo del comentario y la descripción aparecen un área de texto, una casilla de verificación, un número entero, una fecha, una combinación múltiple, una combinación única, una imagen y una combinación envitem.

```
<panel inherits="detailsEdit/detailsEditCenter" overflowX="auto" overflowY="auto" layout="form">
  <!-- See Dictionary.xml for the available kinds of fields -->
  <textarea name="CP_LongString" fieldLabelI18n="custom.cp.Label.Longstring" height="100" width="500" margin="10 0 0 10">
    <bind readOnly="{!isPageEditable}" value="{collectionPlan.CP_LongString}" />
  </textarea>
  <checkbox name="CP_Boolean" fieldLabelI18n="custom.cp.Label.boolean" margin="10 0 0 10">
    <bind readOnly="{!isPageEditable}" value="{collectionPlan.CP_Boolean}" />
  </checkbox>
  <integerfield name="CP_Int" fieldLabelI18n="custom.cp.Label.int" margin="10 0 0 10">
    <bind readOnly="{!isPageEditable}" value="{collectionPlan.CP_Int}" />
  </integerfield>
  <datefield name="CP_Date" fieldLabelI18n="custom.cp.Label.date" margin="10 0 0 10" width="250">
    <bind readOnly="{!isPageEditable}" value="{collectionPlan.CP_Date}" />
  </datefield>
  <nodecombo name="CP_Country_Multi" multiSelect="true" propertyName="Country" formatQuery="false"
  fieldLabelI18n="custom.cp.Label.hvl.country.multi" width="300" margin="10 0 0 10">
    <bind readOnly="{!isPageEditable}" value="{collectionPlan.CP_Country_Multi}" />
  </nodecombo>
  <nodecombo name="CP_Country_Single" multiSelect="false" displayField="path" propertyName="Country" formatQuery="false"
  fieldLabelI18n="custom.cp.Label.hvl.country.single" width="300" margin="10 0 0 10">
    <bind readOnly="{!isPageEditable}" value="{collectionPlan.CP_Country_Single}" />
  </nodecombo>
  <imagefield name="CP_Doc" fieldLabelI18n="custom.cp.Label.document" useREST="true"
  inherits="http://Lectra.com/pdm/common#defaultImageFieldEXT5" managerRef="selSmallImageField" imageWidth="96" imageHeight="96">
    <managerInstance name="Lectra.PDM.Mgrs.imageFieldRestMgr" />
    <bind readOnly="{!isPageEditable}" fileLinkObj="{collectionPlan.CP_Doc}" />
  </imagefield>
  <envitemcombo name="CP_Color" hiddenName="CP_Color" envItemType="MarketingColor" hideTrigger2="true"
  fieldLabelI18n="custom.cp.Label.envitem.color" pageSize="25">
    <bind readOnly="{!isPageEditable}" value="{collectionPlan.CP_Color}" />
  </envitemcombo>
</panel>
```

Modo de consulta



Modo de edición:



Tenga en cuenta que el diseño ha cambiado: hemos elegido un diseño de formulario para este ejemplo, con una barra de desplazamiento, si es necesario (atributo *overflow*).

5.2.2.7 Notas relativas a los campos y funciones de esta pantalla (parte dedicada a los datos generales)

A continuación figuran varios campos y funciones (con ejemplos) que se pueden añadir a esta pantalla. Aparecen en modo de consulta o edición, según la acción del usuario (pulse el botón Editar):

String field (Campo de cadena)

```
<textfield name="CP_String" fieldLabelI18n="custom.cp.Label.string" maxLength="255">  
  <bind readOnly="{!isPageEditable}" value="{collectionPlan.CP_String}"/>  
</textfield>
```

Long String field (Campo de cadena larga)

```
<textarea name="CP_LongString" fieldLabelI18n="custom.cp.Label.Longstring" >  
  <bind readOnly="{!isPageEditable}" value="{collectionPlan.CP_LongString}"/>  
</textarea>
```

Boolean field (Campo booleano)

```
<checkbox name="CP_Boolean" fieldLabelI18n="custom.cp.Label.boolean">  
  <bind readOnly="{!isPageEditable}" value="{collectionPlan.CP_Boolean}"/>  
</checkbox>
```

Long field (Campo largo)

```
<longfield name="CP_Long" fieldLabelI18n="custom.cp.Label.Long" >  
  <bind readOnly="{!isPageEditable}" value="{collectionPlan.CP_Long}"/>  
</longfield>
```

Integer field (Campo de entero)

```
<integerfield name="CP_Int" fieldLabelI18n="custom.cp.Label.int" >  
  <bind readOnly="{!isPageEditable}" value="{collectionPlan.CP_Int}"/>  
</integerfield>
```

Double field (Campo doble)

```
<doublefield name="CP_Double" fieldLabelI18n="custom.cp.Label.double" >  
  <bind readOnly="{!isPageEditable}" value="{collectionPlan.CP_Double}"/>  
</doublefield>
```

Float field (Campo flotante)

```
<floatfield name="CP_Float" fieldLabelI18n="custom.cp.Label.float" >  
  <bind readOnly="{!isPageEditable}" value="{collectionPlan.CP_Float}"/>  
</floatfield>
```

Date field (Campo de fecha)

```
<datefield name="CP_Date" fieldLabelI18n="custom.cp.Label.date" >  
  <bind readOnly="{!isPageEditable}" value="{collectionPlan.CP_Date}"/>  
</datefield>
```

HVL Pick list multi field (Campo múltiple de lista de selección HVL)

```
<nodecombo name="CP_Country_Multi" multiSelect="true" propertyName="Country"  
fieldLabelI18n="custom.cp.Label.hvl.country.multi" formatQuery="false" >  
  <bind readOnly="{!isPageEditable}" value="{collectionPlan.CP_Country_Multi}"/>  
</nodecombo>
```

HVL Pick list single field (Campo único de lista de selección HVL)


```
<nodecombo name="CP_Country_Single" multiSelect="false" displayField="path"
propertyName="Country" fieldLabelI18n="custom.cp.Label.hvl.country.single"
formatQuery="false" >
    <bind readOnly="{!isPageEditable}" value="{collectionPlan.CP_Country_Single}"/>
</nodecombo>
```

Selección del documento de destino

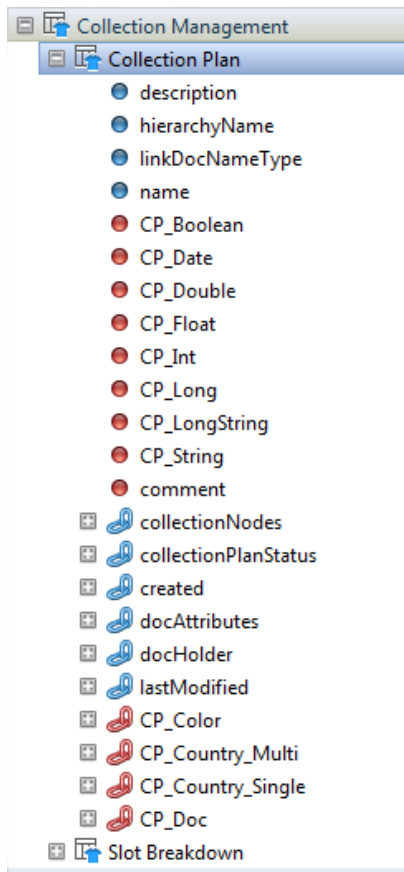
```
<imagefield name="CP_Doc" fieldLabelI18n="custom.cp.Label.document"
useREST="true" inherits="http://lectra.com/pdm/common#defaultImageFieldEXT5"
managerRef="selSmallImageField" imageWidth="96" imageHeight="96">
    <managerInstance name="Lectra.PDM.Mgrs.imageFieldRestMgr" />
    <bind readOnly="{!isPageEditable}" fileLinkObj="{collectionPlan.CP_Doc}" />
</imagefield>
```

Envitem field (Campo Envitem)

```
<envitemcombo name="CP_Unit" hiddenName="CP_Unit" envItemType="Unit"
hideTrigger2="true" fieldLabelI18n="custom.cp.Label.envitem.unit" pageSize="25"
formatQuery="false">
    <bind readOnly="{!isPageEditable}" value="{collectionPlan.CP_Unit}" />
</envitemcombo>
```

Tenga en cuenta que el atributo `formatQuery="false"` es necesario para poder buscar en los campos de lista.

Observe que el atributo del valor `name` es idéntico al que se ha creado en el módulo de Administración y configuración:



Tenga en cuenta que el atributo del valor *fieldLabelI18n* debe definirse en el archivo [PLM-Fashion\PDMLectraPLMParam\messagesCusto_en.properties](#) (y en otro archivo de idioma si es necesario, por ejemplo [messagesCusto_es.properties](#) para el español).

5.2.2.8 Añadir columnas después de las columnas de origen de la cuadrícula de variantes (parte dedicada a la cuadrícula)

Añada los campos y funciones que haya creado en el módulo de Administración y configuración, en la etiqueta grid:

```
<!-- Grid Range Plan Slot Breakdown -->
<grid name="rangeplanslotbreakdowngridDefault" inherits="rangeplanslotbreakdowngrid" >

    <!-- See Dictionary.xml for the available kinds of fields -->

</grid>
```



Y actualice su explorador para ver la modificación.

Ejemplo: en la cuadrícula de variantes aparecen una columna de fecha, una columna de casilla de verificación, una columna de texto, una columna de combinación múltiple, una columna de combinación única, una columna de imagen, una columna de combinación envitem de unidad y una columna de combinación envitem de color marketing.

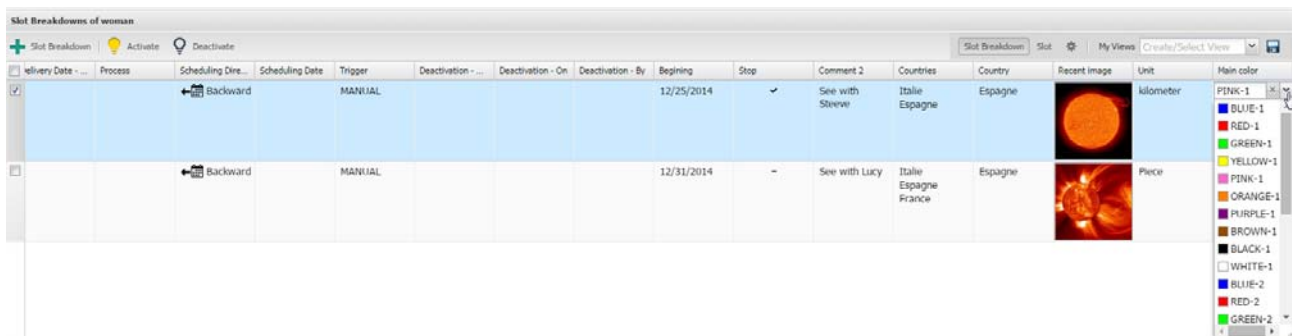
```
<!-- Grid Range Plan Slot Breakdown -->
<grid name="rangeplanslotbreakdowngridDefault" inherits="rangeplanslotbreakdowngrid">
  <column xtype="datecolumn" dataIndex="CP_RP_Date" headerI18n="custom.cp.rp.label.date" format="{#{i18n['i18n.extdateformat']}" editable="true">
    <editorField xsi:type="datefield" format="{#{i18n['i18n.extdateformat']}" />
  </column>
  <column xtype="checkboxcolumn" dataIndex="CP_RP_Boolean" headerI18n="custom.cp.rp.label.boolean" headerId="CP_RP_Boolean" editable="true"/>
  <column xtype="textcolumn" dataIndex="CP_RP_String" headerI18n="custom.cp.rp.label.string" headerId="CP_RP_String" editable="true"/>
  <column dataIndex="CP_RP_Country_Multi" headerI18n="custom.cp.rp.label.hvl.country.multi" headerId="CP_RP_Country_Multi" massEditable="false">
    <editorField xsi:type="nodecombo" multiSelect="true" propertyName="Country" formatQuery="false" />
    <rendererFn name="Lectra.PDM.Format.nodesRenderer()" />
  </column>
  <column dataIndex="CP_RP_Country_Single" headerI18n="custom.cp.rp.label.hvl.country.single" headerId="CP_RP_Country_Single">
    <editorField xsi:type="nodecombo" multiSelect="false" propertyName="Country" formatQuery="false" />
    <rendererFn name="Lectra.PDM.Format.nodeRenderer()" />
  </column>
  <column xsi:type="imagecolumn" dataIndex="CP_RP_Doc" headerI18n="custom.cp.rp.label.document" headerId="CP_RP_Doc" imageSize="medium"
  useREST="true" editable="true">
    <imageEditor inherits="http://Lectra.com/pdm/common#defaultImageFieldEXT5" managerRef="attachmentImageField">
      <managerInstance name="Lectra.PDM.Mgrs.imageFieldRestMgr" />
    </imageEditor>
  </column>
  <column xsi:type="envitemcolumn" envItemType="Unit" headerI18n="custom.cp.rp.label.envitem.unit" dataIndex="CP_RP_Unit"
  headerId="CP_RP_Unit" sortable="false" editable="true" formatQuery="false">
    <rendererFn name="Lectra.PDM.Rest.Format.otherEnvItemRenderer(false)" />
    <editorConfig>
      <listConfig cls="pdm-collectionplan-rangeplanelditor" resizable="true" />
    </editorConfig>
  </column>
  <column xsi:type="envitemcolumn" envItemType="MarketingColor" headerI18n="custom.cp.rp.label.envitem.color" dataIndex="CP_RP_Color"
  headerId="CP_RP_Color" editable="true" formatQuery="false">
    <rendererFn name="Lectra.PDM.Rest.Format.colorEnvItemRenderer(false)" />
    <editorConfig>
      <listConfig cls="pdm-collectionplan-rangeplanelditor"
        resizable="true" />
      <store pageSize="25">
        <dataField name="name" type="string" />
        <proxy xsi:type="rest" type="rest" url="/marketing-colors">
          <extraParams restrequest="true" />
          <reader type="json" />
        </proxy>
      </store>
    </editorConfig>
  </column>
</grid>
```



Modo de consulta



Delivery Date	Process	Scheduling Dir.	Scheduling Date	Trigger	Deactivation - On	Deactivation - By	Beginning	Stop	Comment 2	Countries	Country	Recent image	Unit	Main color
		Backward		MANUAL			12/25/2014	✓	See with Steeve	Italie Espagne	Espagne		kilometer	PINK-1
		Backward		MANUAL			12/31/2014	-	See with Lucy	Italie Espagne France	Espagne		Piece	RED-2

Modo de edición:



Delivery Date	Process	Scheduling Dir.	Scheduling Date	Trigger	Deactivation - On	Deactivation - By	Beginning	Stop	Comment 2	Countries	Country	Recent image	Unit	Main color
		Backward		MANUAL			12/25/2014	✓	See with Steeve	Italie Espagne	Espagne		kilometer	PINK-1
		Backward		MANUAL			12/31/2014	-	See with Lucy	Italie Espagne France	Espagne		Piece	RED-2

5.2.2.9 Notas relativas a los campos y funciones de esta cuadrícula de variantes (parte dedicada a la cuadrícula)

A continuación figuran varios campos y funciones (con ejemplos) que se pueden añadir a esta cuadrícula. Aparecen en modo de consulta o edición, según la acción del usuario (pulse el botón Editar):

Date field in grid (Campo de fecha en la cuadrícula)

```
<column xtype="datecolumn" dataIndex="CP_RP_Date" headerI18n="custom.cp.rp.Label.date"
format="{#{i18n['i18n.extdateformat']}" editable="true">
  <editorField xsi:type="datefield" format="{#{i18n['i18n.extdateformat']}" />
</column>
```

Boolean field in grid (Campo booleano en la cuadrícula)

```
<column xtype="checkboxcolumn" dataIndex="CP_RP_Boolean"
headerI18n="custom.cp.rp.Label.boolean" headerId="CP_RP_Boolean" editable="true"/>
```

String field in grid (Campo de cadena en la cuadrícula)

```
<column xtype="textcolumn" dataIndex="CP_RP_String"
headerI18n="custom.cp.rp.Label.string" headerId="CP_RP_String" editable="true"/>
```

Lista de selección HVL en la cuadrícula: ejemplo con países y selección múltiple

```
<column dataIndex="CP_RP_Country_Multi" headerI18n="custom.cp.rp.Label.hvl.multi"
headerId="CP_RP_Country_Multi" massEditable="false" >
  <editorField xsi:type="nodecombo" multiSelect="true" propertyName="Country"
formatQuery="false" />
  <rendererFn name="Lectra.PDM.Format.nodesRenderer()" />
</column>
```

Lista de selección HVL en la cuadrícula: ejemplo con países y selección única

```
<column dataIndex="CP_RP_Country_Single"
headerI18n="custom.cp.rp.Label.hvl.country.single" headerId="CP_RP_Country_Single" >
  <editorField xsi:type="nodecombo" multiSelect="false" propertyName="Country"
formatQuery="false" />
  <rendererFn name="Lectra.PDM.Format.nodeRenderer()" />
</column>
```

Documento de destino en la cuadrícula

```
<column xsi:type="imagecolumn" dataIndex="CP_RP_Doc"
headerI18n="custom.cp.rp.Label.document" headerId="CP_RP_Doc" imageSize="medium"
useREST="true" editable="true" >
  <imageEditor inherits="http://lectra.com/pdm/common#defaultImageFieldEXT5">
    <managerInstance name="Lectra.PDM.Mgrs.imageFieldRestMgr" />
  </imageEditor>
</column>
```

Envitem (selección múltiple no permitida) en la cuadrícula: ejemplo con Unidad

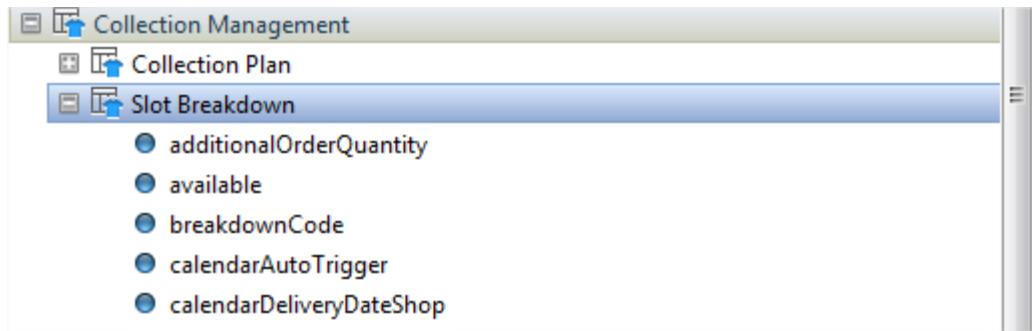
```
<column headerI18n="custom.cp.rp.Label.envitem.unit" xsi:type="envitemcolumn"
envItemType="Unit" dataIndex="CP_RP_Unit" headerId="CP_RP_Unit" sortable="false"
editable="true" formatQuery="false">
  <rendererFn name="Lectra.PDM.Rest.Format.otherEnvItemRenderer(false)" />
  <editorConfig>
    <listConfig cls="pdm-collectionplan-rangeplaneditor" resizable="true" />
  </editorConfig>
</column>
```

Envitem (selección múltiple no permitida) en la cuadrícula con Color de marketing únicamente

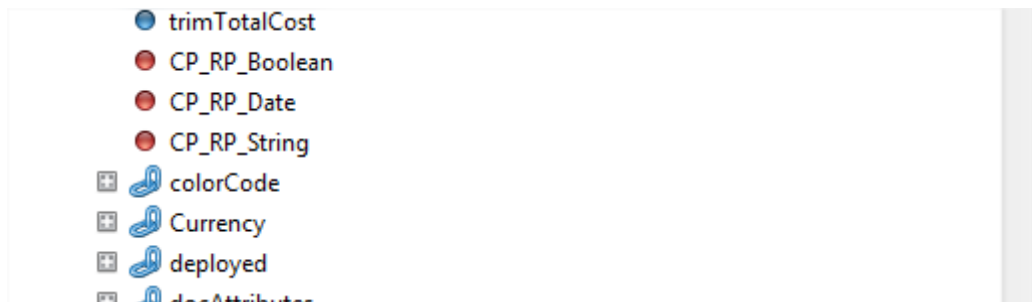
```
<column headerI18n="custom.cp.rp.Label.envitem.color" xsi:type="envitemcolumn"
envItemType="MarketingColor" dataIndex="CP_RP_Color" headerId="CP_RP_Color"
editable="true" formatQuery="false">
  <rendererFn name="Lectra.PDM.Rest.Format.colorEnvItemRenderer(false)" />
  <editorConfig>
    <listConfig cls="pdm-collectionplan-rangeplaneditor"
      resizable="true" />
    <store pageSize="25">
      <dataField name="name" type="string" />
      <proxy xsi:type="rest" type="rest" url="/marketing-colors">
        <extraParams restrequest="true" />
        <reader type="json" />
      </proxy>
    </store>
  </editorConfig>
</column>
```

Tenga en cuenta que el atributo `formatQuery="false"` es necesario para poder buscar en los campos de lista.

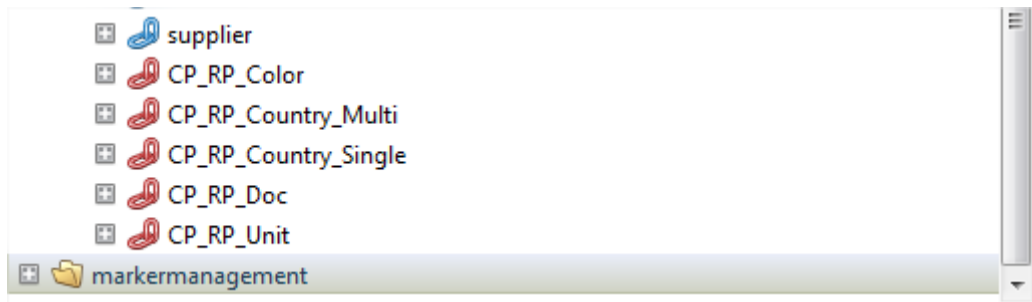
Observe que el atributo del valor `dataIndex` es idéntico al que se ha creado en el módulo de Administración y configuración:



...



...



Tenga en cuenta que el atributo del valor *header18n* debe definirse en el archivo [PLM-Fashion\PDM\LectraPLMParam\messagesCusto_en.properties](#) (y en otro archivo de idioma si es necesario, por ejemplo [messagesCusto_es.properties](#) para el español).

Tenga en cuenta que *marketingColor envitem* tiene una etiqueta store en función de la url `/marketing-colors`; esto es necesario para mostrar las miniaturas de color.

6. CONFIGURACION AVANZADA PARA AÑADIR CAMPOS Y PANELES

6.1 Uso de la herencia

En lugar de copiar descripciones de paneles completas del archivo XML de base en la carpeta "custom" del archivo XML, si se está seguro de la descripción XML, es posible utilizar el concepto de herencia para definir la descripción personalizada.

La herencia se define en la sección [2.2 - Herencia](#).

Cualquier declaración de un elemento gráfico como <panel> se puede aplicar a otras definiciones conservando la definición por defecto que exista en el archivo.

Si ya se ha definido la herencia correspondiente a **#type** o **#topCategoryName**, o no se ajusta a sus necesidades, es posible extender la herencia para definir un nuevo panel que se tenga en cuenta de manera prioritaria.

6.2 Ejemplo concreto

La descripción del panel hereda de 3 definiciones de descripción de posibles paneles:

```
<panel name="description"
inherits="description_{$#type},description_{$#topCategoryName},description" />
```

Si el panel que desea aplicar es diferente de <panel name="descripcion_todos_los_tipos">, diferente de <panel name="description_Styles, Fabrics, Trims, PackagingLabel">:

- se puede crear <panel name= « MaDescription_Custom »>.
- la nueva herencia se puede añadir a la declaración de la descripción del panel anterior.

```
<panel name="description"
inherits="MaDescription_Custom" description_{$#type},description_{$#topCategory
Name},description" />
```

Estando en la primera posición, es la que primero se busca, antes de la descripción por **type** y **topCategoryName**.

Es inútil duplicar el código XML: basta con añadir el código de

```
<panel name="MaDescription_Custom" .../... />
```

Constraint (Restricción):

Si se hereda de una definición del explorador, es necesario definir la etiqueta <contextparameter> siguiente:

```
<panel name="Explorer_CustomStyle" inherits="Explorer">
  <contextparameter name="explorerTitle" value="model.Style" />
  <contextparameter name="categoryName" value="CustomStyle" />
  <contextparameter name="topCategoryName" value="Style" />
  <contextparameter name="explorerName" value="Style" />
</panel>
```

7. MODIFICACION DEL DISEÑO DE PAGINA

Las pantallas también se pueden configurar modificando la presentación de las pantallas.

Para ello, modifique los diseños de página utilizados. Sin embargo, esto implica modificar las definiciones XML de pantallas y respetar una sintaxis XML avanzada.

Los elementos gráficos utilizados en nuestra sintaxis XML provienen de la biblioteca ExtJS. Lectra también ha desarrollado la estructura LPFExt que define los demás elementos gráficos que heredan de la estructura ExtJS.

En los párrafos siguientes se mencionan algunos elementos de la biblioteca ExtJS o de la estructura LPFExt.

7.1 Definición del diseño de página

El diseño de página define el modo de organización de los diferentes bloques de datos en la pantalla: las columnas, las tablas, las cuadrículas, etc.

Siga este enlace para ver algunos Ejemplos [de diseño de página](#).

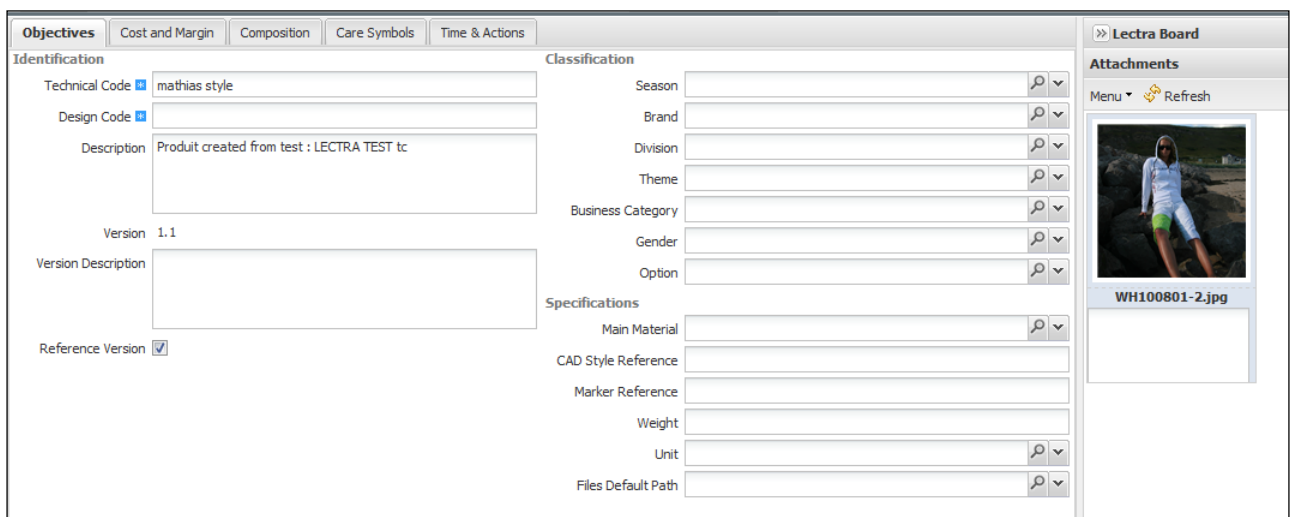
7.2 La estructura ExtJS Erreur ! Source du renvoi introuvable.es la única que tiene un "layout (diseño de página)"

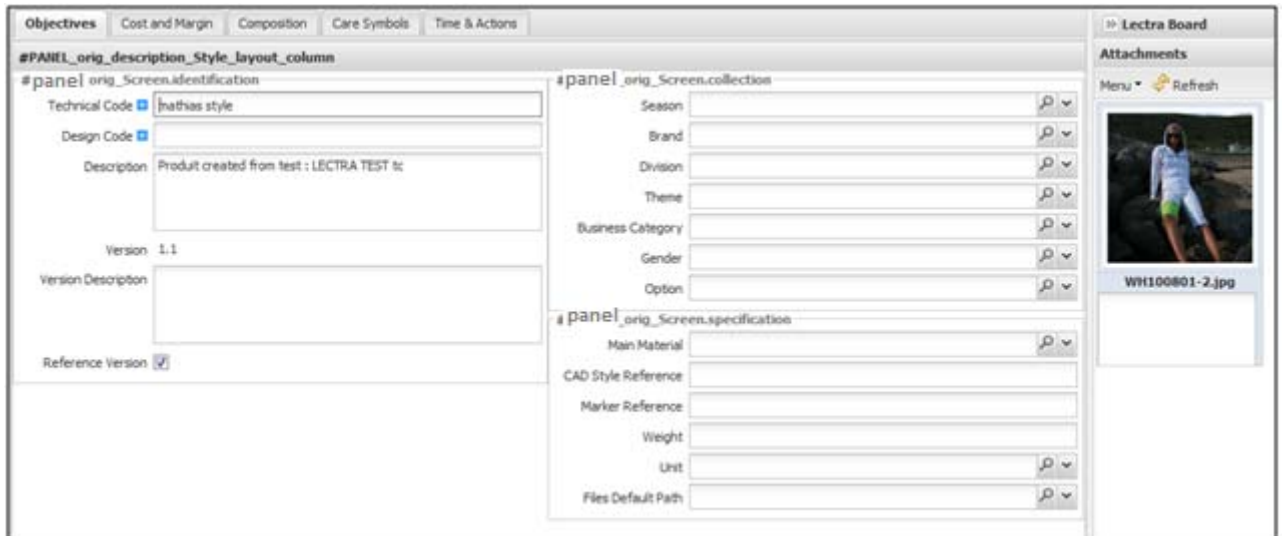
Los contenedores utilizados en las pantallas estudiadas (en el párrafo anterior) son "panels", "tabpanel", "gridpanels". Para obtener la lista completa de contenedores, consulte el Anexo Erreur ! Source du renvoi introuvable. - Erreur ! Source du renvoi introuvable..

7.2.1 Diseño de página de las columnas del panel y de los paneles de la pantalla GO de un Style

Como ya se ha mencionado y discutido en los párrafos anteriores, los paneles son bloques de datos utilizados habitualmente en las pantallas.

La modificación de los paneles se ilustra en las 2 capturas de pantalla de la pantalla "style" GeneralObjectives. Por lo tanto es más fácil visualizar los paneles.





El panel de la pestaña **Objectives** se define mediante el diseño de página de **column (columna)**,

```
<panel name="description_Style" titleI18n="PANEL orig description_Style layout column" layout="column" autoScroll="true" border="true">
```

... y los 3 paneles que la componen se definen con el atributo `columnWidth="0.5"`, que significa que cada uno de ellos ocupa un 50% de la anchura principal del panel.

```
<panel name="identification" titleI18n="fieldset orig Screen.identification" columnWidth="0.5" border="true">
```

```
<panel name="collection" titleI18n="fieldset orig Screen.collection" columnWidth="0.5" border="true">
<panel name="collection" titleI18n="fieldset orig Screen.collection" columnWidth="0.5" border="true">
```

En el módulo de desarrollo del producto, los atributos de diseño de página más utilizados son el margen y el ancho.

ANEXOS

A. ExtJS 3.4 y ExtJS 5.0.1

PDM V4R2 utiliza ExtJS 3.4, salvo para las pantallas del plan de colección, de validación de tabla y las pestañas Composición y Símbolo de lavado, que se basan en ExtJS 6.0.2.

La capa de abstracción LPFExt le permite ser parcialmente independiente de la versión de ExtJS utilizada.

En el archivo xml de descripción LPFExt, los atributos de una etiqueta son interpretados por LPFExt o ExtJS.

Con respecto a los atributos ExtJS, algunos sólo son compatibles con ExtJS 3.4 y otros con ExtJS 3.4 y 6.0.2.

Ejemplo:

```
<displayfield fieldLabelI18n="objective" topLabelCls="" labelAlign="top" labelSeparator=":" bind="{objectives.processDevelopmentType}" />
```

- fieldLabelI18n: interpretado por la capa LPFExt.
- labelAlign y labelSeparator: interpretados por ExtJS 3.4 o ExtJS 6.
- topLabelCls: interpretado por ExtJS 6.

Con la instrucción siguiente:

```
<displayfield fieldLabelI18n="objective" topLabelCls="" labelAlign="top" labelSeparator=":" bind="{objectives.processDevelopmentType}" />
```

Es totalmente interpretado por una pantalla del plan de colección (modo ExtJS 5).

En las demás pantallas (modo ExtJS 3.4), se ignora el atributo topLabelCls.

B. Alternar entre ExtJS 3.4 y ExtJS 6.0.2

Product Developer gestiona la coexistencia de ExtJS 3.4 y ExtJS 6. Dependiendo de la URL procesada por el servidor, el sistema conmuta entre el modo ExtJS 6 y el modo ExtJS 3.4:

- URL del plan de colección → Ext 6 mod.
- Otra URL: Ext 3 mod.

Si quiere utilizar expresamente la biblioteca ExtJS 5 (y superior) en ExtJS 3.4 mod, debe aplicar el sufijo EXT5 al nombre del panel.

Ejemplo:

```
<!-- BEGIN WIDGET EXT5 DEV MODE -->  
<panel name="detail_Panel_PickerEXT5">  
  <managerInstance name="Lectra.PDM.Mgrs.aiPickerMgr"/>  
  <graphicobjectsviewer name="localgr" width="300" itemId="bdetail"  
titleI18n="action.filelink.info" managerRef="boardAIDetail"  
inherits="http://lectra.com/pdm/picker#graphicContainer">  
    <managerInstance name="Lectra.PDM.Mgrs.aiPickerMgr"/>  
    </graphicobjectsviewer>  
  <panel itemId="other" managerRef="emptyDetail" border="false"/>  
</panel>
```

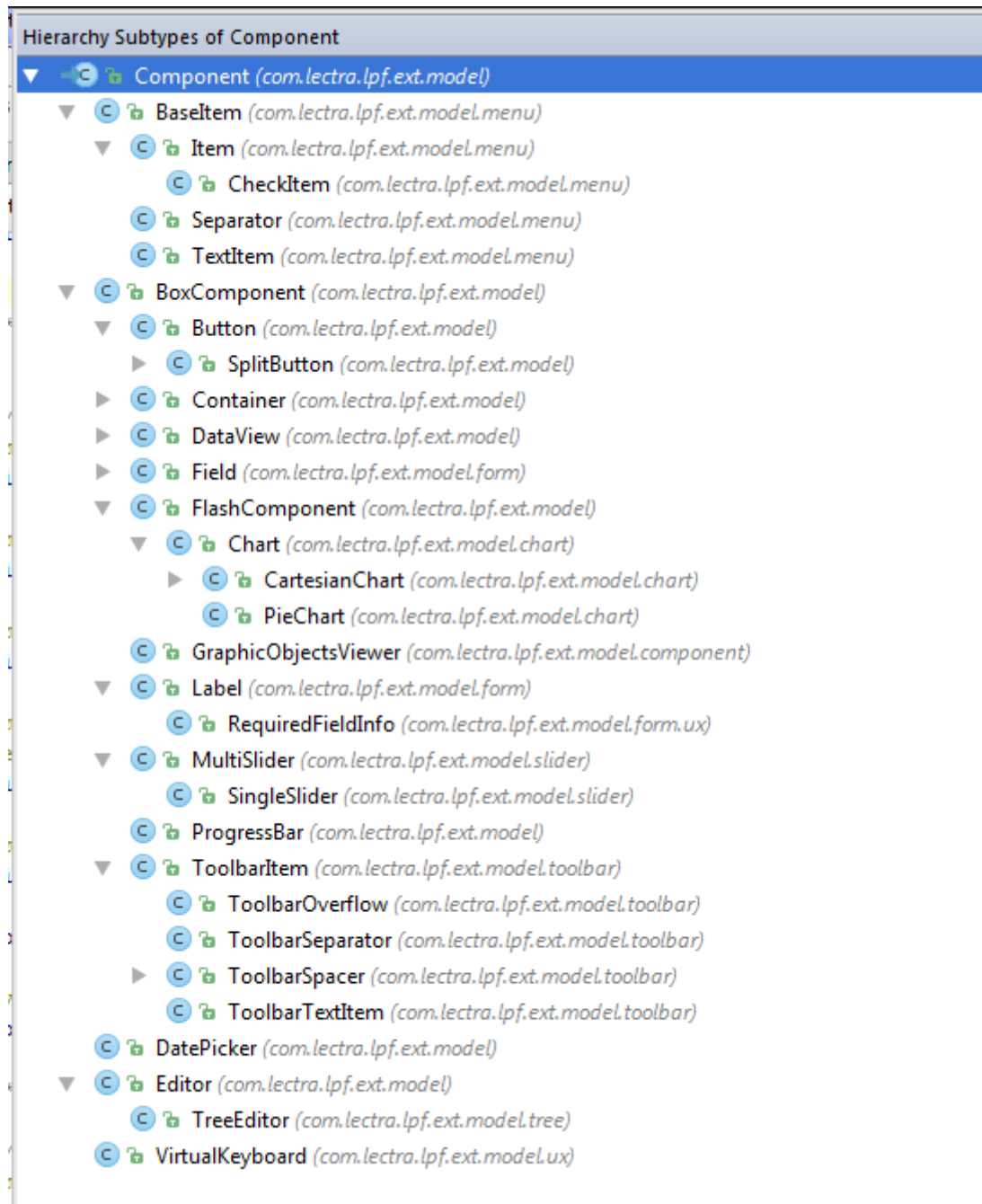
C. Lista de widgets de LPFExt

Un widget de LPFExt es un componente gráfico que se puede crear utilizando la sintaxis xml de LPFExt.

Ejemplo de widget ExplorerGrid:

```
<explorerGrid modePick="false"  
inherits="http://lectra.com/pdm/search/product#searchGrid">  
  <title>${#i18n['search.result']}</title>  
</explorerGrid>
```

La lista de widgets de LPFExt es la siguiente:

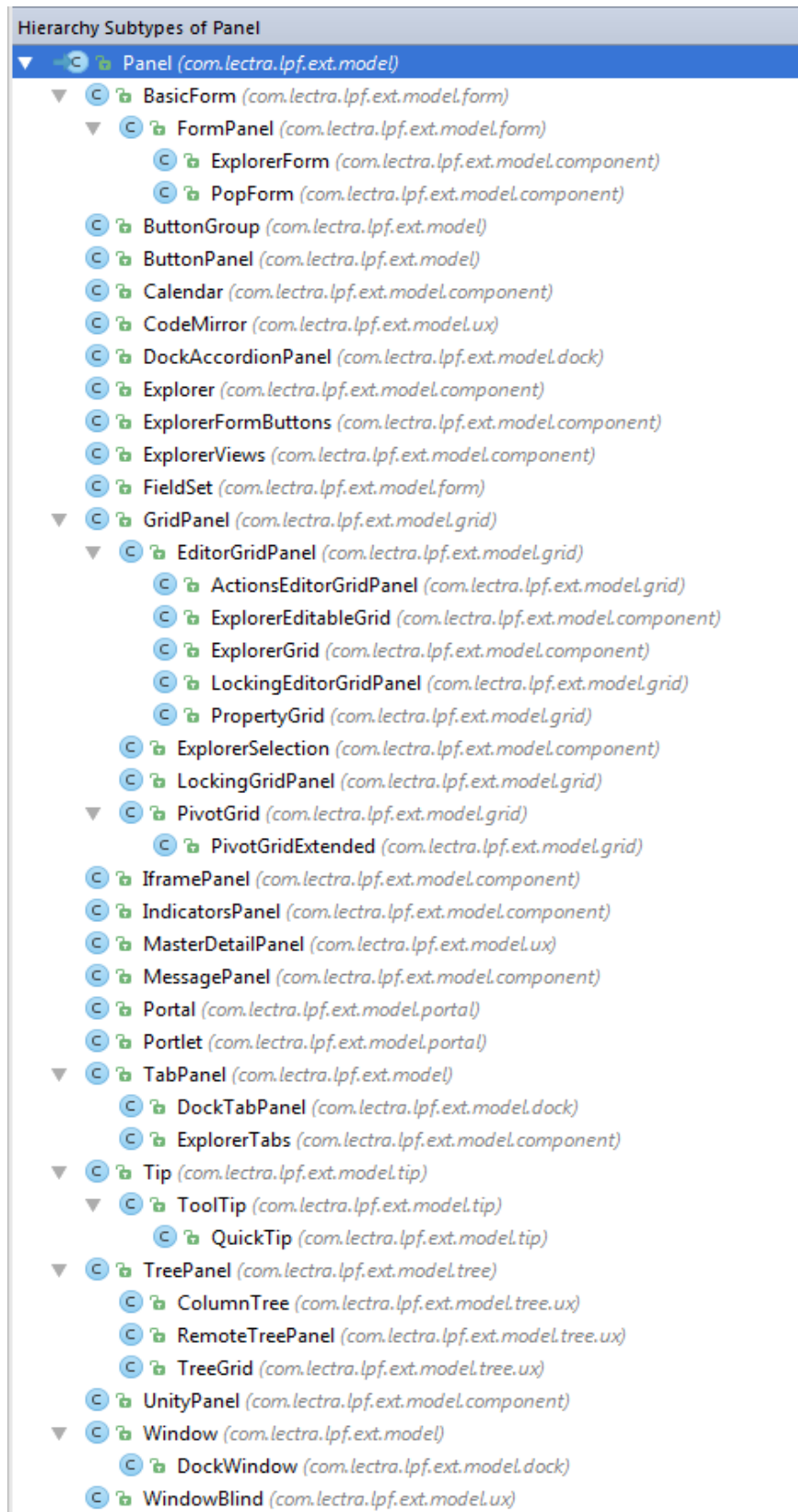


Detalles de los subcomponentes del contenedor:

Hierarchy Subtypes of Container

- ▼ Container (*com.lectra.lpf.ext.model*)
 - ▼ FieldContainer (*com.lectra.lpf.ext.model.form*)
 - MultiSelectExt5 (*com.lectra.lpf.ext.model.form*)
 - GroupComponent (*com.lectra.lpf.ext.model*)
 - ▼ Menu (*com.lectra.lpf.ext.model.menu*)
 - StoreMenu (*com.lectra.lpf.ext.model.menu.ux*)
 - ▼ Panel (*com.lectra.lpf.ext.model*)
 - ▼ BasicForm (*com.lectra.lpf.ext.model.form*)
 - ▶ FormPanel (*com.lectra.lpf.ext.model.form*)
 - ButtonGroup (*com.lectra.lpf.ext.model*)
 - ButtonPanel (*com.lectra.lpf.ext.model*)
 - Calendar (*com.lectra.lpf.ext.model.component*)
 - CodeMirror (*com.lectra.lpf.ext.model.ux*)
 - DockAccordionPanel (*com.lectra.lpf.ext.model.dock*)
 - Explorer (*com.lectra.lpf.ext.model.component*)
 - ExplorerFormButtons (*com.lectra.lpf.ext.model.component*)
 - ExplorerViews (*com.lectra.lpf.ext.model.component*)
 - FieldSet (*com.lectra.lpf.ext.model.form*)
 - ▶ GridPanel (*com.lectra.lpf.ext.model.grid*)
 - IframePanel (*com.lectra.lpf.ext.model.component*)
 - IndicatorsPanel (*com.lectra.lpf.ext.model.component*)
 - MasterDetailPanel (*com.lectra.lpf.ext.model.ux*)
 - MessagePanel (*com.lectra.lpf.ext.model.component*)
 - Portal (*com.lectra.lpf.ext.model.portal*)
 - Portlet (*com.lectra.lpf.ext.model.portal*)
 - ▶ TabPanel (*com.lectra.lpf.ext.model*)
 - ▶ Tip (*com.lectra.lpf.ext.model.tip*)
 - ▶ TreePanel (*com.lectra.lpf.ext.model.tree*)
 - UnityPanel (*com.lectra.lpf.ext.model.component*)
 - ▶ Window (*com.lectra.lpf.ext.model*)
 - WindowBlind (*com.lectra.lpf.ext.model.ux*)
 - PanelHeader (*com.lectra.lpf.ext.model*)
 - PortalColumn (*com.lectra.lpf.ext.model.portal*)
 - ▼ Toolbar (*com.lectra.lpf.ext.model.toolbar*)
 - Paging (*com.lectra.lpf.ext.model.toolbar*)
 - ▶ PagingToolbar (*com.lectra.lpf.ext.model.toolbar*)
 - StatusBar (*com.lectra.lpf.ext.model.toolbar*)
 - ViewPort (*com.lectra.lpf.ext.model*)

Detalles de los subcomponentes del panel:

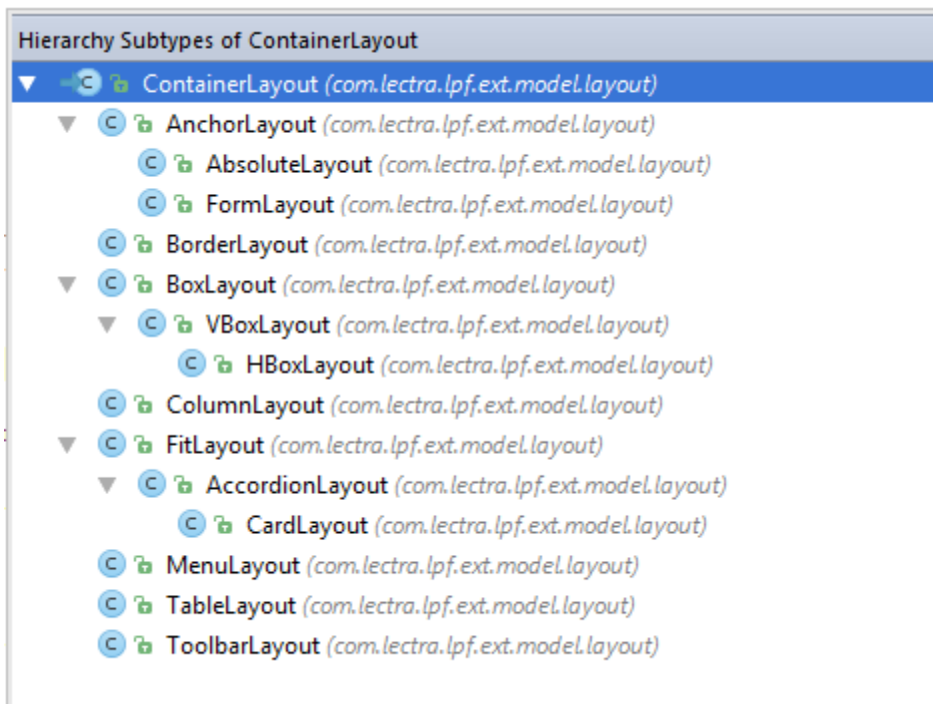


Detalles de los subcomponentes del campo:

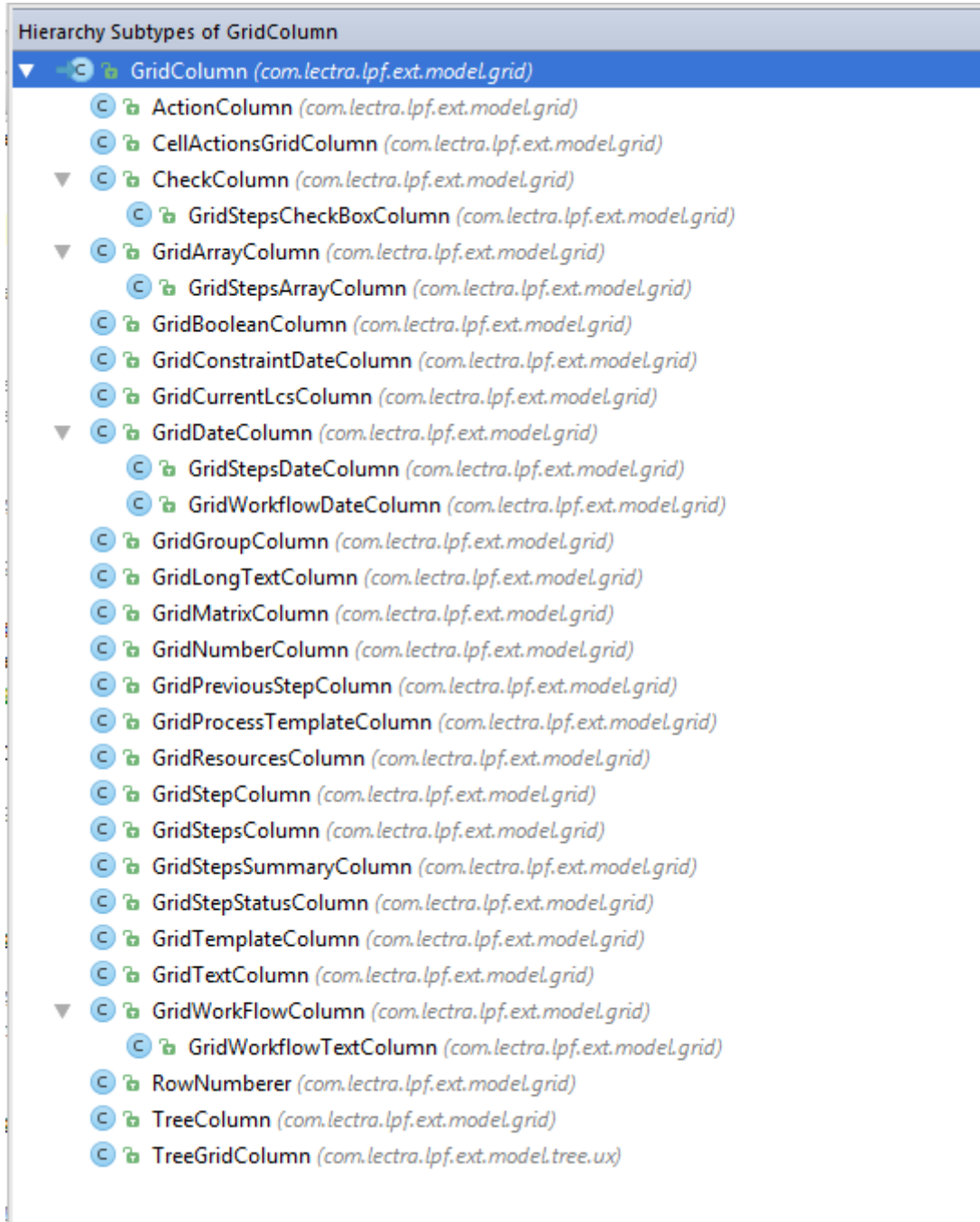


Ciertos widgets corresponden exactamente al widget ExtJS (como un botón), mientras que otros no (como TreePickerField).

D. "Diseños de página" LPFExt



E. Diferentes tipos de columnas en una cuadrícula LPFExt



F. Documentación de ExtJS completa

- Para ExtJS 3.4: <http://docs.sencha.com/ext-js/3-4/#!/api>
- Para ExtJS 5.0.1: <http://docs-origin.sencha.com/extjs/5.0/5.0.1-apidocs/>

y ejemplos

- Para ExtJS 3.4: <http://dev.sencha.com/deploy/ext-3.4.0/examples/>
- Para ExtJS 6.0.2: <http://examples.sencha.com/extjs/6.0.2/examples/>

G. Documentación de los widgets de LPFExt

Hay que considerar dos casos:

1. Los widgets de LPFExt que se corresponden con un componente de ExtJS.
2. Los widgets de LPFExt que no se corresponden con un componente de ExtJS.

Caso 1:

Lea la documentación del widget de ExtJS correspondiente.

Ejemplo: la documentación del widget botón LPFExt.

- Para ExtJS 6: <http://docs.sencha.com/extjs/6.0/6.0.2-classic/#!/api/Ext.button.Button>
- Para ExtJS 3.4: <http://docs.sencha.com/extjs/3.4.0/#!/api/Ext.Button>

Caso 2:

Lea la documentación de ExtJS del componente primario del LPFExt que se corresponde con el componente de ExtJS.

Ejemplo: Documentación del widget de FloatField LPFExt.

FloatField no tiene equivalente ExtJS. El componente primario de FloatField es NumberField, que tiene un equivalente ExtJS. Por consiguiente, lea la documentación de NumberField ExtJS.

- Para ExtJS 5.0.1:
 - <http://docs.sencha.com/extjs/6.0/6.0.2-classic/#!/api/Ext.form.field.Number>
- Para ExtJS 3.4:
 - <http://docs.sencha.com/extjs/3.4.0/#!/api/Ext.form.NumberField>

Por último, puede consultar el archivo xml de la pantalla para saber cómo se configura un widget. Hay numerosos ejemplos de uso.

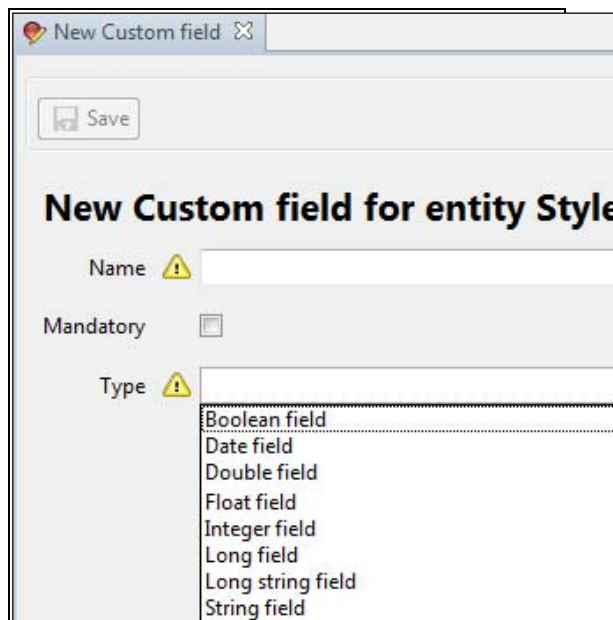
H. Cambio de ubicación del directorio de LPFExt

En PLM V5R1, los archivos LPFExt se encuentran en lpf/ext3/nombre del directorio.

I. Recordatorio de la aplicación de Administración y configuración

Para completar con los valores los nuevos campos que se añaden a una pantalla, es necesario crear los valores correspondientes en la base de datos.

Pueden añadirse campos del siguiente tipo:

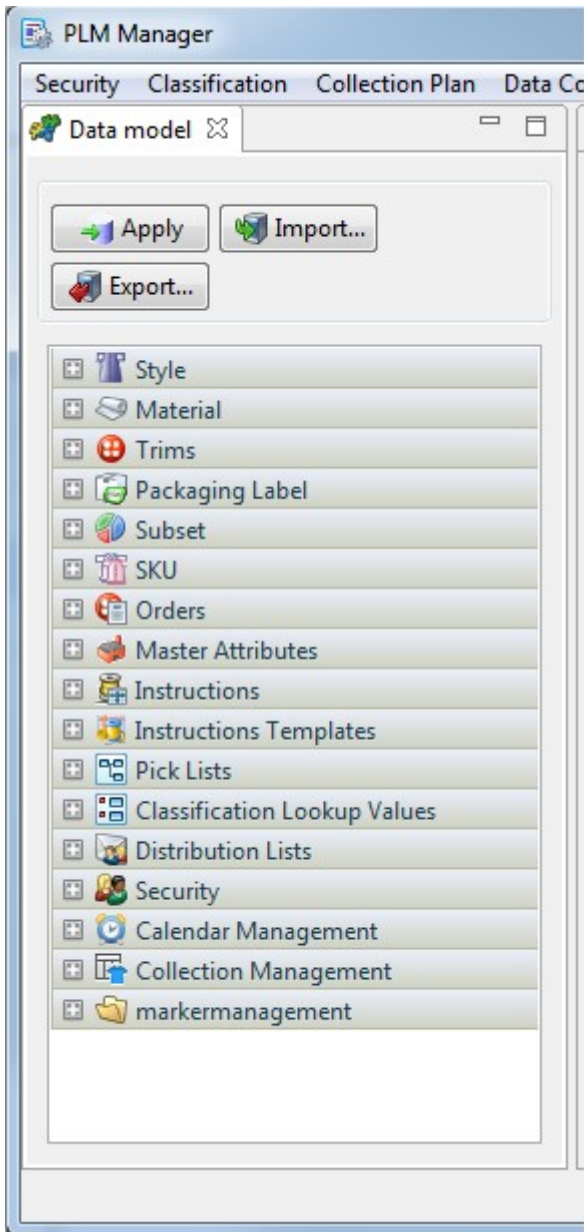


Por ejemplo: "MyCustomStringField" en "Style" debe desencadenar la creación de datos "MyCustomStringField" en el tipo "String Field" de la base de datos.

Para ello:

Abra el módulo de Administración y configuración

y el menú **Configuración de datos/Modelo de datos extendidos** para mostrar el panel que describe todos los datos:



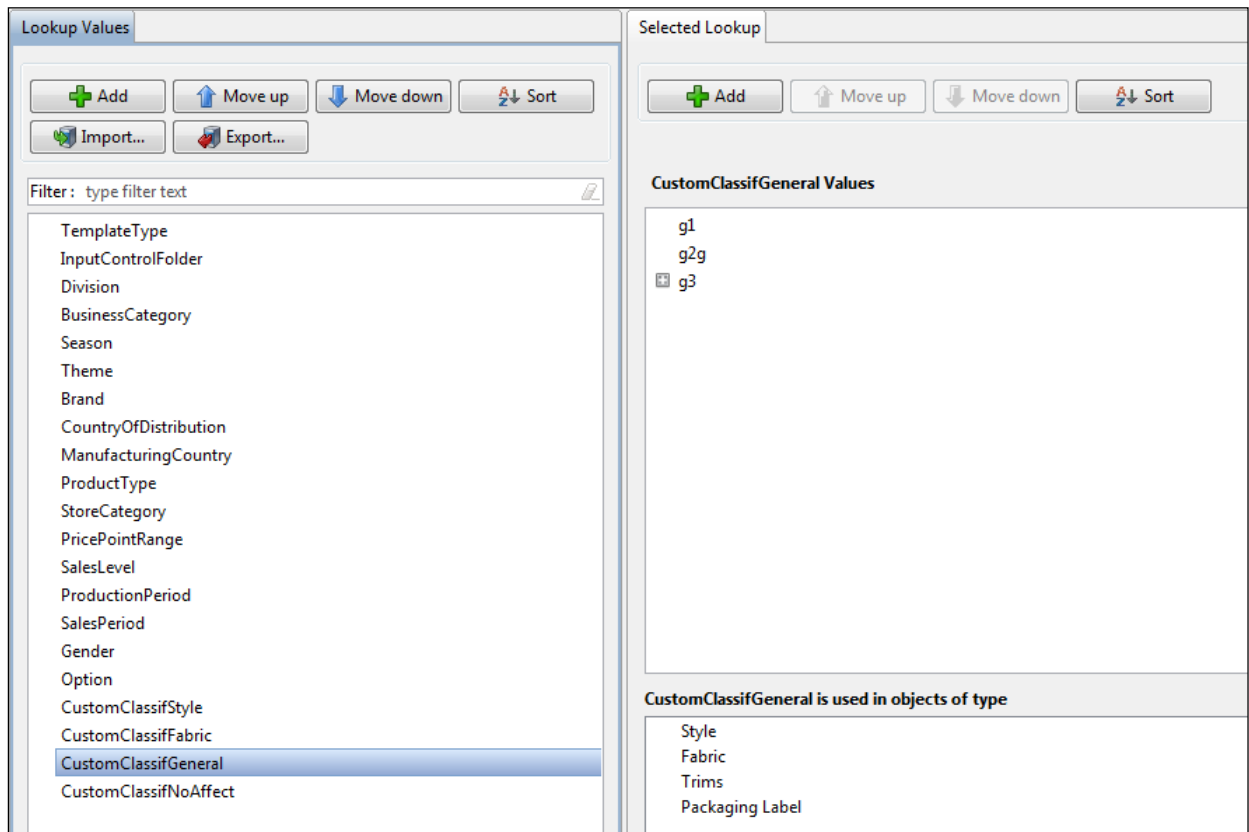
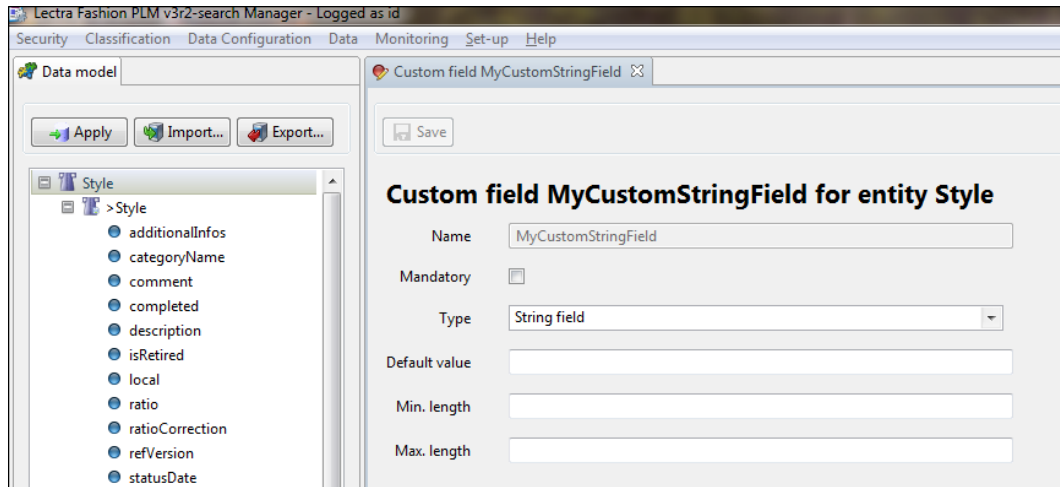
Haga clic derecho en Style (Modelo).

En el menú contextual: haga clic en Add Custom Field (Crear una configuración).

Asigne al valor el nombre MyCustomStringField.

Seleccione su tipo: String field (Campo de cadena)

Haga clic en Guardar y aplicar la configuración a todas las subcategorías si necesita extender la definición a las subcategorías.



J. Añadir otros tipos de campos simples en xml

Todos los tipos de campos que se pueden definir en el módulo de Administración y configuración tienen su correspondiente definición en el archivo XML, conforme a las siguientes asociaciones:

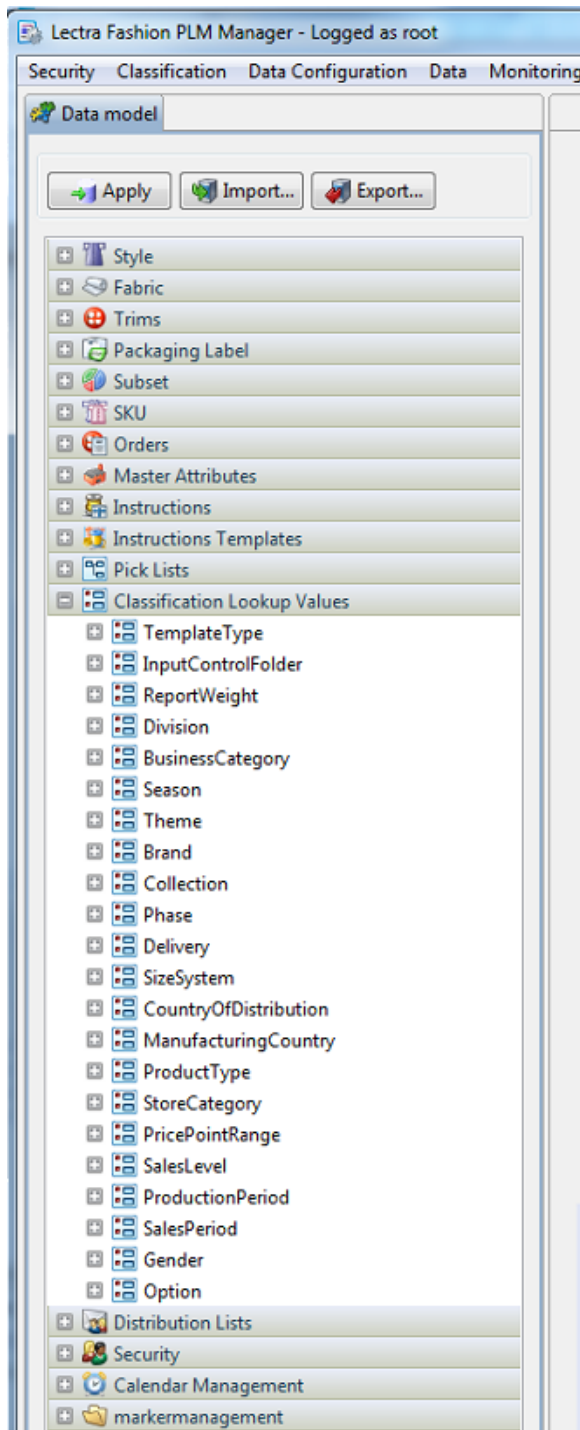
Boolean field	Twincombo / checkbox
Date field	datefield
Double field	doublefield
Float field	floatfield
Integer field	integerfield
Long field	longfield
Long string field	textarea
String field	textfield



Para introducir los distintos tipos de campos en el archivo xml de descripción hay que hacerlo a través del archivo [namespace/custom/Lectra.PDM.User.xml](#).

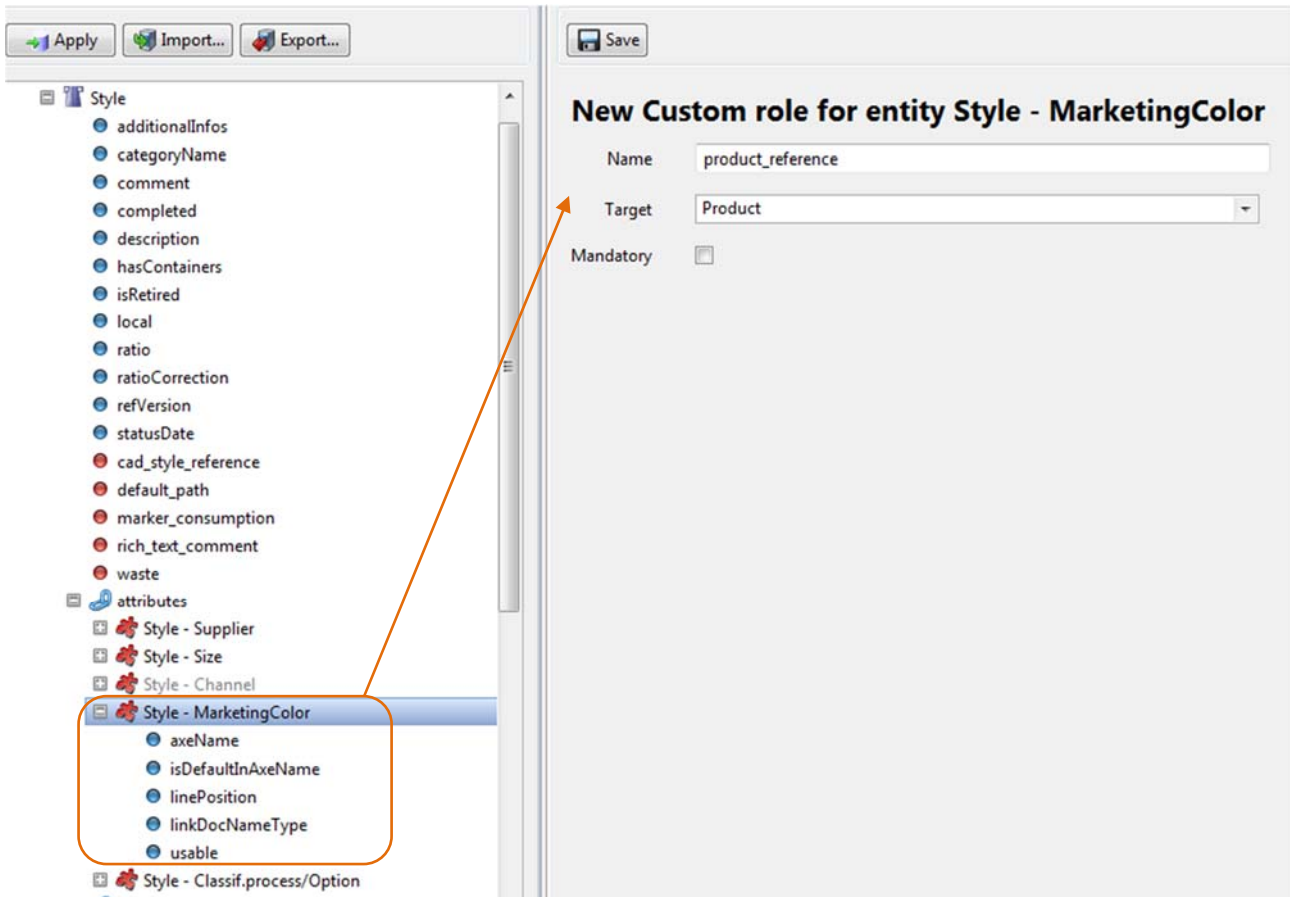
K. Añadir "Pick Lists" (Listas de selección) y "Classification Lookup Values (Valores de clasificación)"

Observando el modelo de datos del módulo de Administración y configuración, puede ver qué es lo que se ha creado:



L. Añadir el "Product Picker (Selector de productos)" en los atributos de los productos

Observando el modelo de datos del módulo de Administración y configuración, puede crear una función personalizada que haga referencia al modelo de datos del producto en uno de sus atributos de productos:



Cambiar su screensCusto.xml:

```
<screen name="Axis.Style" extends="Axis">
  .../...
  <tab name="colorsTab"...>
    <table name="colors"...>
      .../...
      <column name="product_reference"
i18n="product_reference" editable="true"/>
    </table>
  </tab>
  .../...
```

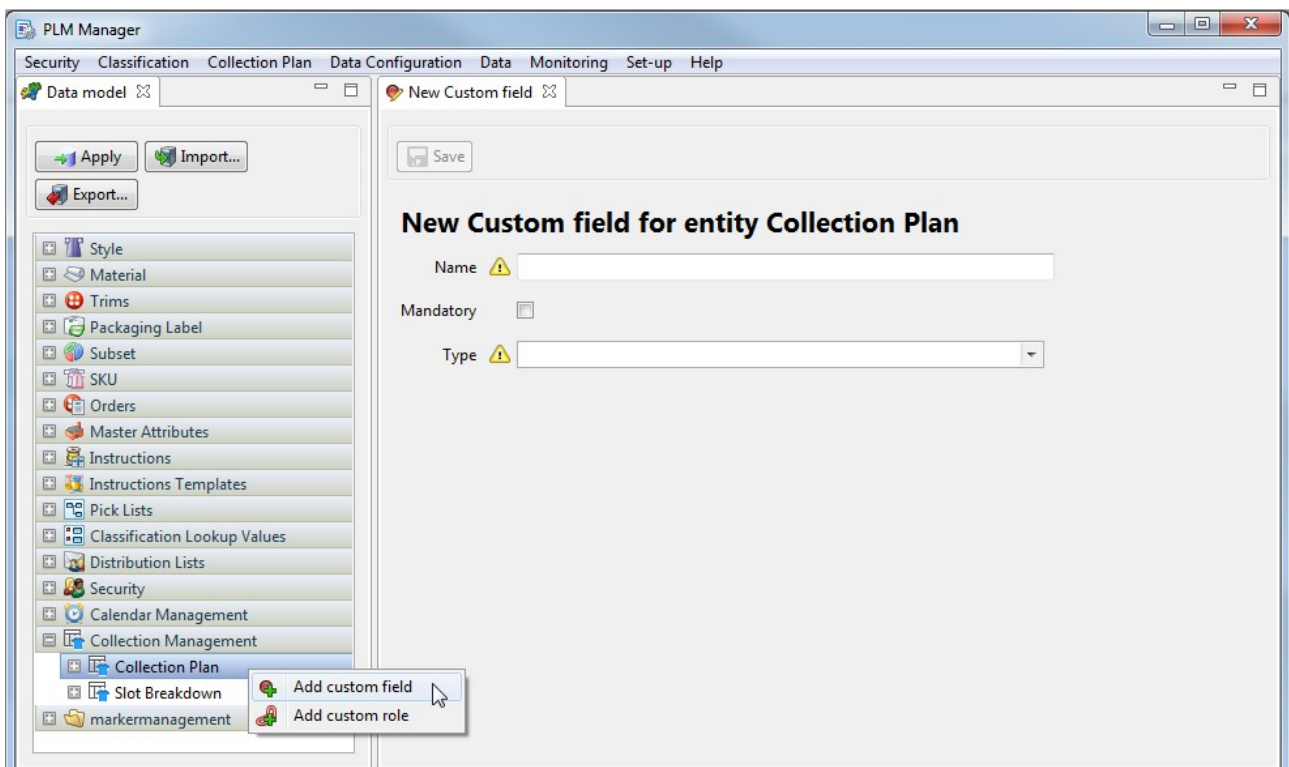
</screen>

Resultado en la pantalla de Atributos del Style:



M. Añadir campos y vínculos personalizados a los datos generales de un plan de colección

Haga clic derecho en la entrada "Collection Plan" y seleccione "Añadir campo personalizado" o "Añadir vínculo personalizado".



N. Añadir campos y vínculos personalizados a la cuadrícula de variantes de un plan de colección

Haga clic derecho en la entrada "Slot Breakdown (Variante)" y seleccione "Añadir campo personalizado" o "Añadir vínculo personalizado".