

Jasper Report

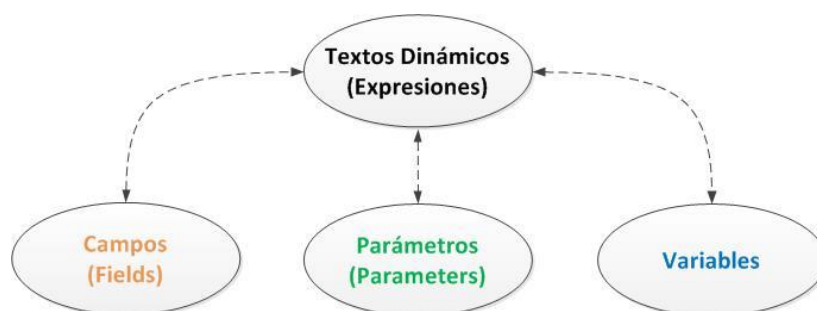
Introducción:

Jasper Report, funciona compilando y generando reportes, por medio de dos partes esenciales, un Data Resource y un Template. En el Template se define el estilo, la forma, y el conjunto de datos que se van a mostrar de forma estática, y los datos que serán cargados dinámicamente en el momento de generación del reporte; estos datos serán obtenidos del Data Resource, el cual podría ser definido como una tabla de la base de datos, o por medio de código java (la opción que usaremos nosotros).

Templates: Explicación de los componentes y su funcionamiento

Textos:

En el template/documento de JR, hay dos tipos de textos, los estáticos y los dinámicos (expresiones), los estáticos como su nombre indica se definen una vez, y se mantienen sin mutar incluso durante el periodo de compilación del informe final, mientras que el dinámico es el que va cambiando dependiendo de la información ingresada (la parte copada y mágica de la herramienta). Por ahora nos centraremos en los dinámicos, los cuales a su vez se dividen en tres grupos:



A continuación pasaremos a definir cada uno:

Campos: también conocidos como Fields, le dicen al reporte donde colocara los valores/datos obtenidos a través del data source, que en caso de hacerlo por medio de java y utilizando el API de JR o JCore, serian por ejemplo las propiedades de un objeto. Para hacer uso de estos textos dinámicos, en el reporte se deben invocar de la siguiente forma: ***\$F{nombreDeLaPropiedad}***.

Parámetros: Los parámetros (***\$P{nombreDelParametro}***) son valores que usualmente se pasan al reporte directamente desde el programa que crea el JasperPrint del reporte (en nuestra aplicación Java). Aunque también existen algunos parámetros internos que podemos leer pero no modificar. Para hacer uso de estos parámetros simplemente indicamos el nombre del parámetro en el lugar que queremos colocarlo. Pueden encontrar los nombres y significados de los parámetros internos en la documentación de JasperReports.

Variables: son objetos usados para almacenar valores calculados, y al igual que los Parámetros, JR tiene algunas variables internas que podemos usar. Para hacer uso de estas variables o valores calculados, en el reporte se deben invocar de la siguiente forma:

`$V{}`.

Grupos:

Dentro de los templates también existen lo que podemos definir como Grupos o Grupos de Reporte, que básicamente es una asociación de datos, que trabajan en unidad/conjunto, o sea que tienen sentido únicamente estando juntos, al menos en este reporte. Por cada nuevo grupo de datos que creemos/agreguemos al reporte, automáticamente se nos generaran dos nuevas secciones en el reporte "*NuevoGrupo Group Header*" y "*NuevoGrupo Group Footer*" y una variable llamada "*nuevoGrupo_count*". Los grupos suelen ser útiles cuando queremos particionar la información, por ejemplo a la hora de hacer una gráfica.

Sub-Reportes:

Los sub-reportes son reportes que luego se incorporan a otro reporte, al que normalmente se lo llama reporte maestro. Se suelen usar cuando la complejidad de un reporte es muy grande, y se lo divide en varios reportes más chicos, que luego son "pegados" en un reporte maestro (principal), debido a esto nos provee de una mayor flexibilidad y legibilidad, y a su vez nos permite crear mejor los templates.

Data Source: Definición

El data Source, no es más ni menos que la fuente de información de Jasper, se podría decir que es la estructura de datos de la que saca la información en el momento de compilar/ generar un reporte; por lo que sin data Source el template y Jasper por sí solo no serían nada.

Hay dos estrategias a la hora de definir un Data Source:

1. Definirlo como una conexión a una BBDD (básicamente como una tabla de la base de datos, donde cada variable usada por el generador y el template sea una columna de la base de datos).
2. Definirlo vía java como un objeto.

