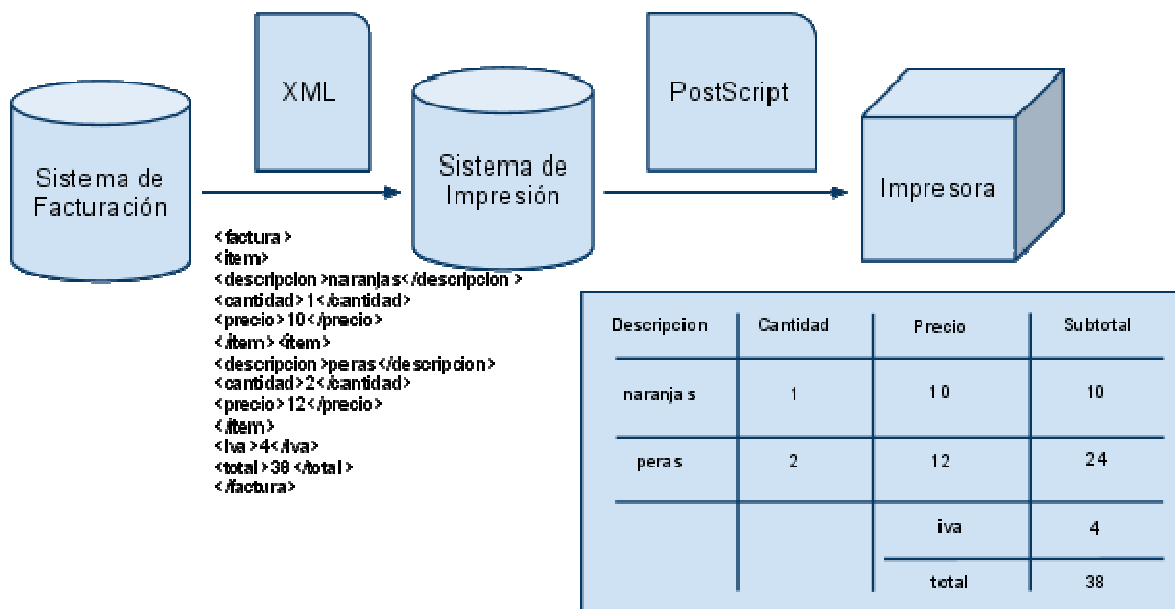


# Trabajo Práctico Final - Impresor Masivo

## Introducción

Las compañías con gran cantidad de clientes frecuentemente requieren de la impresión en lote de miles de documentos. Generalmente se trata de la factura de servicios como luz, gas, teléfono; pero podrían ser también campañas de marketing masivas. Para imprimir esta cantidad de documentos se utilizan grandes impresoras que reciben, mediante un formato específico los documentos a imprimir, por ejemplo [PostScript](#). Las compañías utilizan software específico para convertir los datos generados por sus sistemas comerciales a un formato que la impresora pueda procesar. Es en esta transformación que se le da formato gráfico a los documentos.

El típico ejemplo de esto es el siguiente:



En este ejemplo el **sistema de facturación** genera un archivo [XML](#) con los datos de la factura generada. Estos datos pasan por el **sistema de impresión** y éste genera un archivo PostScript con la forma que se le envía la **impresora**. Es importante notar algunos detalles aquí:

1. Los datos pueden ser repetitivos o listas, como en este caso que recibe una lista de ítems de la factura.
2. El formato gráfico de la factura esta en el **sistema de impresión**.

3. No todos los datos expresados en la factura están explícitos en los datos de entrada:
  - a. Los títulos de las columnas están en el formato
  - b. Los subtotales fueron calculados de los datos de entrada.

## ***Descripción***

Se pide implementar un Software de Impresión masivo que a partir de XML genere PostScript. Se requiere:

1. Implementar un editor gráfico del formato del documento a generar.
  - a. El editor debe poder crear, editar, guardar, probar versiones de plantillas de documentos. Específicamente probar quiere decir poder levantar datos de un XML y aplicarlos a la plantilla que está actualmente en edición.
  - b. Las plantillas de documentos tendrán fechas de validez desde-hasta.
2. Implementar un servidor que procesará los documentos aplicando el témpate y generando el PostScript. El servidor puede trabajar de dos modos:
  - a. **Lote** o batch, en este modo recibirá por línea de comando el nombre del archivo a procesar y generará el PostScript en otro archivo.
  - b. **En línea** publicando un servicio [SOAP/HTTP](#). En este modo recibirá el contenido del documento y lo escribirá a una carpeta en un .ps nuevo. Como respuesta al servicio indicara el nombre del archivo generado.
  - c. Tanto en el modo en **lote** o **en línea** se recibirá la fecha de validez del documento y se utilizará versión de plantilla que corresponda a esa fecha.

## ***Características***

Las platillas deben poder incluir:

1. Nombre del tag XML que agrupa los datos.
2. Texto tomado del XML de origen.
3. Fecha de validez desde-hasta (hasta puede ser indefinido, pero modificable)
4. El editor debe poder tomar un archivo JPG de fondo para uso de hojas preimpresas.
5. Se debe poder realizar operaciones entre los datos recibidos en el XML. Para esto se utilizará el lenguaje de desarrollo de PostScript
6. Posibilidad de dibujar líneas gráficas en la plantilla.
7. Como lenguaje de especificación de los campos en el documento se utilizará [xpath](#). Las peticiones serán relativas al tag XML que contiene los datos. Ver el ejemplo para más información.
8. El sistema debe poder almacenar más de una plantilla, estas serán accedidas por nombre, y el nombre será parámetro del proceso batch o del servicio en línea.

En general, una plantilla debe poseer objetos que permitan cumplir con las características mencionadas. Al principio del trabajo deberá entonces presentarse,

como parte del diseño, la lista de objetos que soportará la plantilla y el comportamiento de cada uno. Para demostrar que el sistema será suficientemente flexible.

## ***Restricciones***

Es necesario que la solución al procesar archivos de texto en ningún momento suba a memoria el archivo. Esta restricción es importante ya que este sistema debe poder procesar archivos masivos sin restricción de la capacidad del equipo.

Se permite utilizar bibliotecas de código abierto que implementen parseo de XML y XPATH.

## ***Ejemplo plantilla***

Como se explicó anteriormente la plantilla contiene el nombre del tag XML que agrupo los datos para imprimir un ejemplar. Un archivo determinado contendrá N instancias de ese tag. Y cada instancia contendrá una cantidad arbitraria de información que será solicitada vía xpath.

Para el caso del ejemplo que recibe el siguiente XML:

```
<factura>

<item>

  <descripcion>naranjas</descripcion>

  <cantidad>1</cantidad>

  <precio>10</precio>

</item>
<item>

  <descripcion>peras</descripcion>

  <cantidad>2</cantidad>

  <precio>12</precio>

</item>

<iva>4</iva>

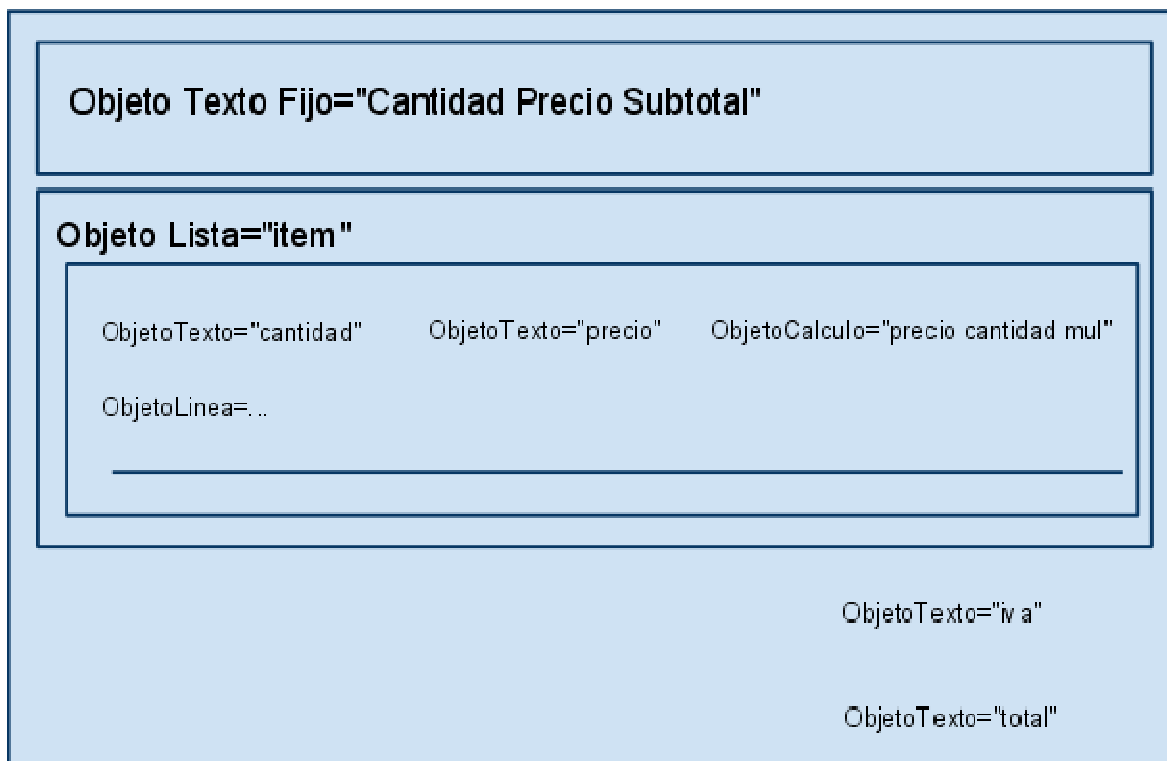
<total>38</total>

</factura>

<factura>...
```

Una posible plantilla sería la siguiente:

**Documento="factura"**



En esta se especifica que el documento aparece como "factura". Por lo tanto aplicará la plantilla a cada tag factura del archivo. Dentro de este pondrá una cabecera de texto fijo con los títulos de las columnas. A continuación tendrá una lista de todos los tags "ítem". Dentro de cada tag obtendrá los elementos de la lista para las columnas y en la columna de subtotal realizara la multiplicación entre cantidad y precio. Cada objeto ítem también tendrá una línea que separa las filas. Finalmente aparece el monto de IVA y total de la factura.

## ***Referencias***

[1] [http://partners.adobe.com/public/developer/ps/index\\_specs.html](http://partners.adobe.com/public/developer/ps/index_specs.html)

[2] <http://www-cdf.fnal.gov/offline/PostScript/BUEBOOK.PDF>