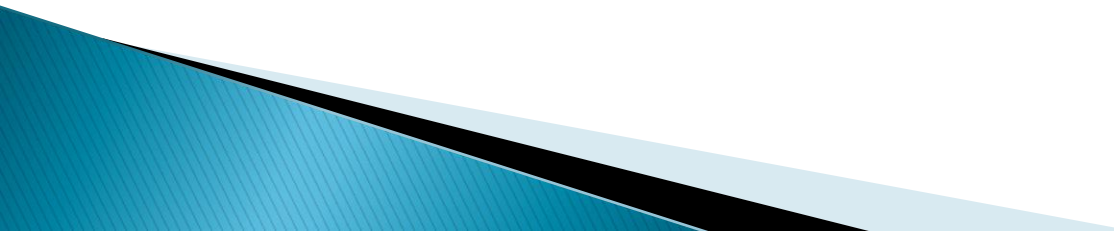


Xslt Taller

Para desarrollos con FE

Contenido

- ▶ Alcance
 - ▶ Definiciones
 - ▶ Procesadores Xslt
 - ▶ Elementos principales
 - ▶ Ejemplos
 - ▶ Caso Factura Electrónica
 - Diseños
 - Reusabilidad
 - Funcionalidades esperadas
- 

Alcance

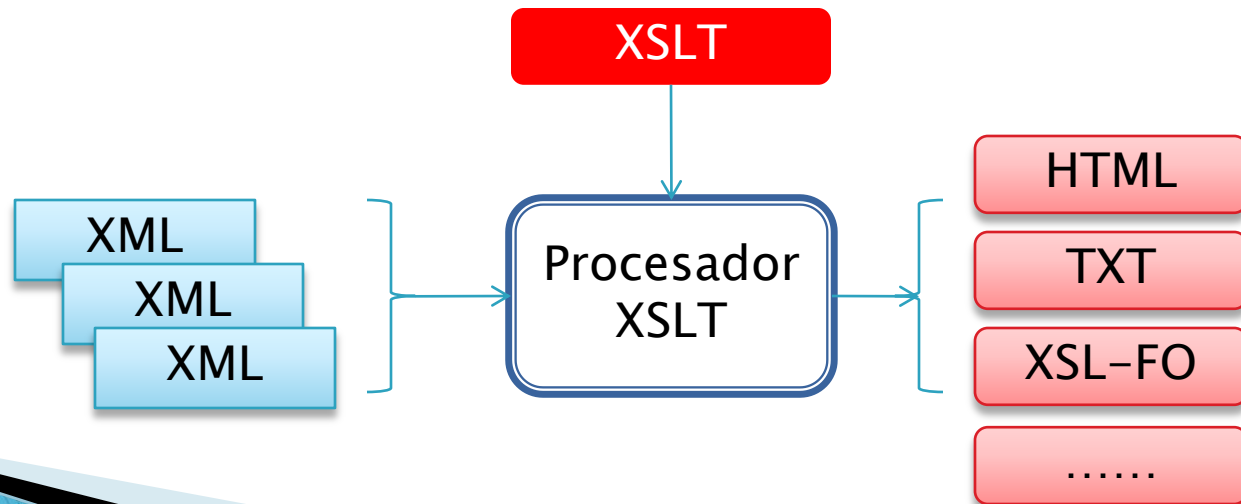
- ▶ Revisar los conceptos básicos que permitan generar páginas Html desde XML vía XSLT
- ▶ Con el ejemplo básico se puede completar el desarrollo de FE.

Definiciones

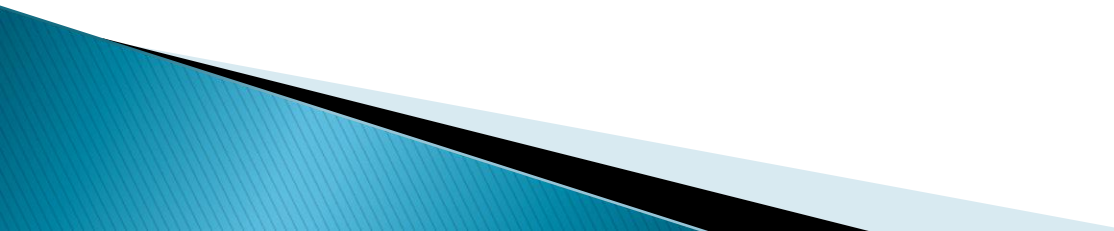
- ▶ **Qué es XSLT**
 - XSL (eXtensible Stylesheet Language) lenguaje de estilos para XML
 - XSLT se refiere a Transformaciones.
- ▶ **Conocimiento básico**
 - XML, Xpath, Html (y sus otros componentes como js, css, etc)

Cómo funciona ?

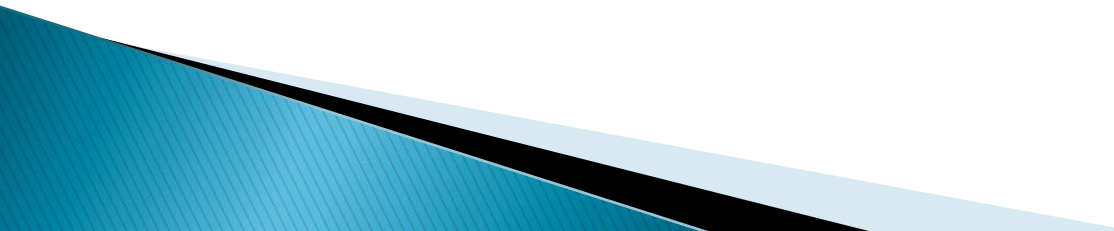
- ▶ Se usa como fuente un XML, Se define un archivo XSLT y se genera un archivo de salida en el formato que se desee
- ▶ Se debe utilizar un procesador XSLT
- ▶ **Muy poderoso, permite transformar xml en múltiples formatos de salida**



Ventajas

- ▶ Es un estándar de la industria
 - ▶ Independiente del lenguaje de programación.
 - ▶ Los resultados se pueden alterar alterando el archivo xsl. No se requiere cambio de código.
 - ▶ Con los procesadores en cliente, el diseño se puede hacer sin necesidad de compilador y servidor y ver los resultados de inmediato.
 - ▶ Hay procesadores para C#, java, oracle, etc.
- 

Algunas herramientas

- ▶ Comerciales
 - Stylus Studio
 - Altova XmlSpy
 - Visual Studio
 - ▶ Open source
 - Saxon processor
 - ▶ Embebidos en los navegadores
 - Chrome, Firefox
- 

Versiones

▶ Xslt 1.0

- Incluye funciones básicas de transformación
- Está incluido en la mayor parte de las librerías

▶ Xslt 2.0

- Incluye extensiones
- Incluye mayor número de funciones propias
- Otros
- Existen pocos procesadores que lo soportan, pero java tiene su librería open source saxon

Ejercicio: Transformación básica

- ▶ Usar el ejemplo:
 - XSLT: ejemplo.xsl
 - XML: ejemplo.xml
- Verificar que funciona en los browsers

Elementos del XSL

- ▶ Template
- ▶ Apply-templates
- ▶ value-of
- ▶ for-each
- ▶ sort
- ▶ if
- ▶ choose

<xsl:template>

- ▶ <xsl:template> usado para aplicar “plantillas” a secciones del xml
- ▶ Sintaxis:
 - <xsl:template
 - name = QName
 - match = expression
 - priority = number
 - mode = QName >
 - </xsl:template>

<xsl:apply-template>

- ▶ <xsl:apply-template> da la señal al procesador xslt para buscar el template adecuado para aplicar, basado en el contexto actual del nodo en el que se encuentra.
- ▶ Sintaxis:
 - <xsl:apply-template
 - select = Expression >
 - </xsl:apply-template>

<xsl:value-of>

- ▶ <xsl:value-of> permite extraer el valor de un campo o atributo
- ▶ Debe incorporar el atributo “select” usando un xpath
- ▶ Sintaxis:
 - <xsl:value-of
 - select = Expression
 - disable-output-escaping = "yes" | "no" >
 - </xsl:value-of>

<xsl:for-each>

- ▶ <xsl:for-each> permite iterar sobre los elementos iguales de un mismo nivel
- ▶ Debe incorporar el atributo “select” usando un xpath
- ▶ Sintaxis:
 - <xsl:for-each
 - select = Expression >
 - </xsl:for-each>

<xsl:sort>

- ▶ <xsl:sort> permite ordenar sobre los elementos iguales de un mismo nivel
- ▶ Debe incorporar el atributo “select” para indicar el campo o campos por los que se ordena
- ▶ Sintaxis:
 - <xsl:sort
 - select = string-expression
 - lang = { nmtoken }
 - data-type = { "text" | "number" | QName }
 - order = { "ascending" | "descending" }
 - case-order = { "upper-first" | "lower-first" } >
 - </xsl:sort>

<xsl:if>

- ▶ <xsl:if> permite validar un valor o condición de un elemento u atributo y actuar acorde
- ▶ No hay “else”, en caso de requerir usar <xsl:choose>
- ▶ Sintaxis:
 - <xsl:if
 - test = boolean-expression >
 - </xsl:if>

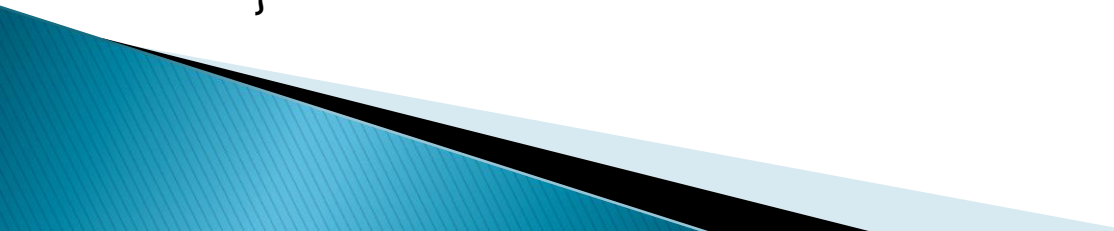
<xsl:choose>

- ▶ <xsl:choose> permite decidir sobre uno u otro valor, se usa en conjunto con <xsl:when> y <xsl:otherwise>
- ▶ Ver Ejemplo:
 - <xsl:choose >
 - <xsl:when expression>
 - <xsl:when expression>
 - <xsl:otherwise>
 - </xsl:choose>

Transformación en c#

```
public ActionResult VerFactura()
{
    string html = String.Empty;
    var myXslTrans = new XslCompiledTransform();
    /* Reemplazar por path adecuado */
    myXslTrans.Load(Server.MapPath("/static/FESimple.xsl"));

    using (StringWriter sw = new StringWriter())
    using (XmlWriter xwo =
        XmlWriter.Create(sw, myXslTrans.OutputSettings))
    {
        /* Reemplazar por xml obtenido de la base de datos */
        myXslTrans.Transform(Server.MapPath("/static/rFE.xml"), xwo);
        html = sw.ToString();
    }
    return Content(html);
}
```



Ejercicios:

- ▶ Ejercicio 01:
 - Crear un html con los siguientes campos: dVerForm, dId, dNroDF, dFechaEm
- ▶ Ejercicio 02:
 - Crear un html con los siguientes campos: iAmb, iTpEmis, iDoc
- ▶ Ejercicio 03:
 - Crear un html con los detalles de la factura en tabla:
 - dDescProd, cUnidad, dPrUnit, dValTotItem

El diseño de la factura

- ▶ Múltiples formas de visualización
 - Generalmente una básica con información general
 - Una con todos los detalles
 - Una versión de impresión
 - Pueden agregarse otras
- ▶ Se debe controlar el nivel de acceso
 - Los detalles solo deberían verse por parte de los contribuyentes y los funcionarios de la DGI

Ejemplo

► Revisar el código del ejemplo

SFE Panama

Ver Factura

Ver Detalles


Imprimir

Cerrar

No. 111093698

FACTURA

2017-09-18T19:07:00-05:00



CÓDIGO ÚNICO DE FACTURA ELECTRÓNICA [CUFE]
FE01100023824-0930-21262328001201707150000151340010117450183430

CÓDIGO AUTORIZACIÓN
12712721-12-212

MODALIDAD EMISIÓN
NORMAL

FECHA AUTORIZACIÓN
2017-07-12T08:09:10

EMISOR

RUC
121212-2-212121

NOMBRE
DISTRIBUIDORA ENORME S.A.

DIRECCIÓN
Av Del Sitio Escondido y la que cruza

TELÉFONO
234-2323

RECEPTOR

CONTRIBUYENTE

RUC
8-NT-12121

NOMBRE
MINORISTA SUPER.

DIRECCIÓN
Calle Principal No. 1221

TELÉFONO
534-2323

Detalle

Línea	Código	Descripción	Cantidad	Precio Unidad	Monto	Impuesto	Total
1	910202012	ZUCARITAS 840 GRAMOS (E)	1	5.90	5.90	0.00	5.90
2	010202012	NEVADA FREE UHT LTRO (E)	1	1.65	1.65	0.00	1.65
3	510202012	NEVADA FREE UHT LTRO (E)	1	1.65	1.65	0.00	1.65
4	010675012	HS SH SUAVE M 180ML (A)	1	2.99	2.99	0.21	2.99
Totales					12.19	0.21	12.40

Ejercicios

- ▶ Completar el xslt de factura
- ▶ Incluir en código