

Henry Alberto
Diosa

RESUMEN

XML(eXtensible Markup Language), el lenguaje de marcado extensible permite separar la abstracción estructural de un conjunto de documentos de su contenido y presentación; esto quiere decir, que se pueden determinar tres capas o niveles de un documento:

- Su abstracción estructural o Definición de Tipo de Documento (DTD).
- Su contenido basado en etiquetado o documento XML.
- Su presentación, definida a través de hojas de estilo.

Esta visión modular de un documento permite plantear una solución para llevar a la WEB la información de proyectos de grado e investigaciones respetando los estándares de documentación y alcance de contenidos que fije una institución universitaria; en este artículo, por claridad y facilidad de comprensión, se desarrolla un ejemplo muy básico de como difundir fichas resumen de investigación por medio de un sitio WEB, que den a conocer a la comunidad externa e interna a la institución sobre los trabajos que se vienen desarrollando o se han terminado por parte de estudiantes o grupos de trabajo específicos y con la connotación de ser un procedimiento sencillo y relativamente fácil de implantar.

Palabras clave: Lenguaje de Marcado Extensible (XML), Definición de Tipo de Documento, Hojas de Estilo, Internet, Documento de Investigación.

ABSTRACT

XML, the eXtensible Markup Language allows to separate the structural abstraction of a group of documents of their content and presentation; this means that three layers or levels of a document can be determined:

- Their structural abstraction or Document Type Definition(DTD).
- Their content based on having labeled or XML document.
- Their defined presentation through Style Sheets.

This modulate vision of a document allows to outline a solution to show in the WEB the information of degree projects and investigations

according to the documentation standards and scope of contents defined by an institution; in this article, for clarity and easiness of understanding, a very basic example is developed, diffusing of records investigation summary by means of a WEB site that give to know to the external and internal community of the university around the works that one comes developing or they have ended, developed by students or specific work groups, with the connotation of being a simple and relatively easy procedure of implanting.

Key words: eXtensible Markup Language (XML), Document Type Definition, Style Sheets, Internet, Document of Investigation.

INTRODUCCIÓN

Preguntas que surgen cuando se empieza a trabajar en el estado del arte de un tema particular son:

- ¿Qué ha desarrollado una Universidad específica al respecto?
- ¿Que han hecho otras Universidades o empresas?
- ¿Se podrá encontrar la información de rastreo de temas en formatos resumidos que permitan posteriormente enfocar los esfuerzos hacia profundizar en aquellos temas que realmente interesan para un proyecto en particular?
- ¿Se podrá hacer una investigación preliminar del estado del arte a través de Internet?

Este artículo intenta vislumbrar un camino sencillo y de bajo costo para que las instituciones universitarias y de investigación modelen e implementen un sistema de publicaciones WEB de sus trabajos de investigación con tecnologías estandarizadas o en vía de estandarización a nivel internacional. Se presentan los pasos requeridos para llevar a la WEB una **Ficha Resumen de Investigación**(con base en un modelo básico para Grupos de Trabajo) que el autor ha modelado usando DTDs, XML y XSL (eXtensible Style Language); luego se propone un modelo procedimental para lograr implementar esta solución en una institución universitaria y se presentan proyecciones de las tecnologías inherentes a esta propuesta. Por el corto espacio y objetivo propio de este escrito, no se hace una revisión detallada de los estándares de las tecnologías aquí utilizadas; no obstante, quien esté intere-

¿Se podrá encontrar la información de rastreo de temas en formatos resumidos que permitan posteriormente enfocar los esfuerzos hacia profundizar en aquellos temas que realmente interesan para un proyecto en particular?

Un DTD es el modelo que el diseñador de formatos electrónicos de documentos debe establecer para validar la estructura de contenidos que se deseen publicar.

sado en profundizar en el tema y localizar herramientas que apoyen estos desarrollos puede consultar el sitio del World Wide Web Consortium(W3C)[1], organización a cargo de promover y estandarizar dichas tecnologías a nivel mundial.

I. LA ABSTRACCIÓN ESTRUCTURAL DE DOCUMENTOS A PUBLICAR

El primer paso para estandarizar el alcance de contenidos de lo que se quiere publicar es determinar la estructura que debe respetar cualquier documento; este modelo estructural lo define un DTD o Definición de Tipo de Documento. Un DTD es el modelo que el diseñador de formatos electrónicos de documentos debe establecer para validar la estructura de contenidos que se deseen publicar; en el ejemplo que aquí se desarrolla se establece un DTD para una **Ficha Resumen de Investigación o de Grupo de Trabajo**. En el Cuadro No. 1 se muestra el código requerido para nuestro caso en estudio. Como se puede observar, lo que se intenta especificar con un DTD es la estructura genérica de la **FICHA RESUMEN DE INVESTIGACIÓN** que se requiere estandarizar a nivel institucional, de esta manera cualquier documento de esta naturaleza debe respetar la abstracción estructural genérica y será validado contra ésta; si no es válido, debe rechazarse porque no respetaría el estándar de documentación definido. El DTD es importante en el sentido de determinar cuántos elementos de información y atributos (con sus tipos, algo flexibles) pueden publicarse y permitir así normalizar lo que se puede publicar. Es notorio que el ejercicio de definición de un DTD no es simple porque debe recoger de manera, lo más completa posible, la especificación estructural de un tipo de documento.

II. CONTENIDOS CON XML RESPECTANDO EL DTD RESPECTIVO

Después de elaborar el molde que se debe respetar para un conjunto de documentos se puede iniciar la elaboración de documentos con datos concretos que respeten el estándar fijado(DTD) por la organización; para nuestro caso se presenta en el Cuadro No. 2 un ejemplo de una posible **FICHA RESUMEN DE INVESTIGACIÓN** que respeta el DTD definido en el Cuadro No.1.

CUADRO No.1. DTD DE UNA FICHA RESUMEN DE INVESTIGACIÓN

Nombre del archivo: ficha_proyecto.dtd

```
<!ELEMENT proyectos (proyecto)+>
<!ELEMENT proyecto (obj_esp+,resumen,director+,investigador*,sector_obj*,estudiante*)>
<!ATTLIST proyecto
  titulo CDATA #REQUIRED
  obj_gen CDATA #REQUIRED
  estado (concepcion|formulado|modelado|desarrollo|pruebas|terminado) "concepcion"
  fecha_inicio CDATA #REQUIRED
  facultad CDATA #REQUIRED
  proy_curricul CDATA #REQUIRED
  fecha_aprob CDATA #REQUIRED>
<!ELEMENT obj_esp (#PCDATA)>
<!ELEMENT resumen (#PCDATA)>
<!ELEMENT director EMPTY>
<!ATTLIST director
  nombres CDATA #REQUIRED
  apellidos CDATA #REQUIRED
  cedula CDATA #REQUIRED
  grado (auxiliar|asistente|asociado|titular) "auxiliar"
  mail CDATA #REQUIRED
  telefono CDATA #REQUIRED
  imagen ENTITY #IMPLIED>
<!ELEMENT investigador EMPTY>
<!ATTLIST investigador
  nombres CDATA #REQUIRED
  apellidos CDATA #REQUIRED
  cedula CDATA #REQUIRED
  grado (auxiliar|asistente|asociado|titular) "auxiliar"
  e-mail CDATA #REQUIRED
  telefono CDATA #IMPLIED
  imagen ENTITY #IMPLIED>
<!ELEMENT sector_obj (empresa*, concepto_viabil+)>
<!ATTLIST sector_obj
  descripcion CDATA #REQUIRED
  oferta CDATA #REQUIRED>
<!ELEMENT empresa EMPTY>
<!ATTLIST empresa
  razon_social CDATA #REQUIRED
  contacto CDATA #REQUIRED
  direccion CDATA #REQUIRED
  telefono CDATA #REQUIRED
  e-mail CDATA #REQUIRED>
<!ELEMENT concepto_viabil EMPTY>
<!ATTLIST concepto_viabil
  responsable CDATA #REQUIRED
  fecha CDATA #REQUIRED
  telefono CDATA #REQUIRED
  e-mail CDATA #REQUIRED
  concepto CDATA #REQUIRED>
<!ELEMENT estudiante EMPTY>
<!ATTLIST estudiante
  nombres CDATA #REQUIRED
  apellidos CDATA #REQUIRED
  telefono CDATA #IMPLIED
  e-mail CDATA #REQUIRED
  imagen ENTITY #IMPLIED
  semestre CDATA #REQUIRED
  subproyecto CDATA #REQUIRED
  descripcion CDATA #REQUIRED
  estado (c|f|m|d|p|t) "c">
```

Nombre de archivo: proyectos.xml

```

< ?xml version="1.0" encoding="US-ASCII"?>
< ?xml-stylesheet href="proyectos.xml" type="text/xml"?>
<!DOCTYPE proyectos SYSTEM "ficha_proyecto.dtd"
[ <!NOTATION JPEG SYSTEM "IExplore.exe">
<!ENTITY foto SYSTEM "foto.jpg" NDATA JPEG> ]>
< proyectos>
  < proyecto
    titulo="GRUPO DE TRABAJO CORBA-OSM"
    obj_gen="Consolidar una infraestructura tecnológica (hardware, software, comunicaciones de datos, recurso humano calificado y de gestión de conocimiento) en un período de dos años sobre los temas de CORBA Y OBJETOS DE SOFTWARE MÓVILES que posicionen la Universidad como líder en estos ámbitos y sea punto de referencia a nivel nacional e internacional"
    estado="formulado"
    fecha_inicio="SEPTIEMBRE / 2000"
    facultad="INGENIERIA"
    proy_curricul="INGENIERIA DE SISTEMAS"
    fecha_aprob="ENERO 2001">
    < obj_esp> Implementar los laboratorios de Procesamiento Basado en Objetos Distribuidos y Objetos de Software Móviles con base en recursos de software gestionados a través de la Universidad y desarrollos propios. Desarrollar proyectos de I+ D aplicados a la solución de problemas del ámbito colombiano con base en las dos tecnologías abordadas. Generar tradición de investigación en los temas eje del Grupo de Trabajo. Fortalecer la academia tanto en pregrado como en postgrado.
    < /obj_esp>
    < resumen> La evolución de la Ingeniería de Software y de las facilidades en las herramientas de programación junto con el avance vertiginoso de las redes de computadores con la consecuencia inherente de una mayor complejidad para el análisis, diseño, implementación y mantenimiento de sistemas soportados en el computador han traído a la arena tecnológica los enfoques del Análisis, Diseño y Programación Orientada a Objetos. ...
    < /resumen>
    < director
      nombres="HENRY ALBERTO"
      apellidos="DIOSA"
      cedula="80264735"
      grado="asistente"
      mail="hdiosa@asenet.com.co"
      telefono="2545170"
      imagen="foto" />
    < investigador
      nombres="HENRY ALBERTO"
      apellidos="DIOSA"
      cedula="80264735"
      grado="asistente"
      e-mail="hdiosa@asenet.com.co" />
    < sector_obj
      descripcion="SALUD"
      oferta="SOFTWARE PARA GESTIÓN DE HISTORIAS CLÍNICAS">
    < empresa
      razon_social="EPSs e IPSs"
      contacto="alguien"
      direccion="alguna"
      telefono="alguno"
      e-mail="alguno" />
    < concepto_viabil
      responsable="alguien"
      fecha="alguna"
      telefono="alguno"
      e-mail="alguno"
      concepto="alguno" />
    < /sector_obj>
    < estudiante
      nombres="DIANA"
      apellidos="TOLOSA"
      telefono="ALGUNO"
      e-mail="ALGUNO"
      semestre="DECIMO"
      subproyecto="GESTION DE HC GENERAL"
      descripcion="PROYECTO DE MODELAMIENTO E IMPLEMENTACIÓN..."
      estado="d" />
    < /proyecto>
  < /proyectos>

```

A este documento se le puede aplicar un proceso de validación automático frente al DTD respectivo usando analizadores de documentos XML. Podrían aparecer muchos documentos XML que se validarían contra el DTD respectivo y obligaría al creador del documento XML a respetar la norma estructural establecida a través del DTD; se podría decir que un DTD puede tener muchas instancias de documentos

XML, similar al concepto de clase en la programación orientada a objetos que puede tener muchas instancias de clase u objetos. La línea :

```
<!DOCTYPE proyectos SYSTEM "ficha_proyecto.dtd"...
```

permite validar el documento XML anterior contra su especificación estructura **ficha_proyecto.dtd**. Un

documento que respeta un DTD se denomina documento XML válido; así mismo, se pueden elaborar documentos XML que no se validen frente a un DTD y se denominan **bien formados** si cumplen la especificación XML del W3C. Hay programas analizadores que permiten determinar el cumplimiento de la **validez** y **buena formación** o sólo de la **buena formación** si se requiere; dichos programas también se podrían desarrollar con base en los estándares que se han promulgado.

III. FORMATEAR LA PRESENTACIÓN DEL DOCUMENTO

Para llevar la información a la WEB, ésta se debe trasladar a un formato entendible por los navegadores en las máquinas cliente, normalmente este formato es HTML (Hypertext Markup Language). Los últimos navegadores buscan soportar la lectura de documentos XML y puede asociarse a dichos documentos un formato de presentación predefinido a través de hojas de estilo que permiten determinar: tamaño de fuentes, color, ubicación, fondos, etc... Para un documento XML se asocia una hoja de estilo, en nuestro caso la línea:

```
<?xml-stylesheet href="proyectos.xsl" type="text/xsl"?>
```

relaciona el documento XML anterior con la hoja de estilo **proyectos.xsl**. El código que conforma esta hoja de estilo se presenta en el Cuadro No. 3.

A través de esta hoja de estilo un navegador obtendrá la información suficiente para presentar la información que se asocie a la misma. Actualmente se está en proceso de estandarizar un lenguaje de hojas de estilo denominado XSL(eXtensible Stylesheet Language) por parte del W3 Consortium; esto permitirá a todos los navegadores soportar XML con más propiedad; sin embargo, un camino que se puede seguir es usar XSLT[2] ((eXtensible Stylesheet

Nombre de archivo: *proyectos.xsl*

```
<?xml version="1.0"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
<xsl:template match="/">
  <html> <head> <title> EJEMPLO APLICACION XML DE FICHA DE INVESTIGACION< /title> </head>
  <body bgcolor="#FFFFFF">
    <table>
      <tr>
        <td width="40">
          </td>
        <td width="580">
          <h1 style="padding-left: 15px; background-color: Blue; color: #ffff; font-family: Verdana, Arial; font-size: 18pt; text-align: justify; font-style: italic; letter-spacing: 0.5em"> FICHA DE INVESTIGACION< /h1>
          <xsl:for-each select="proyectos">
            <xsl:apply-templates select="proyecto"/>
          </xsl:for-each>
          <xsl:for-each select="proyectos/proyecto">
            <xsl:apply-templates select="obj_esp"/>
            <xsl:apply-templates select="resumen"/>
            <xsl:apply-templates select="director"/>
            <xsl:apply-templates select="investigador"/>
            <xsl:apply-templates select="sector_obj"/>
            <xsl:apply-templates select="estudiante"/>
            <xsl:apply-templates select="fecha_aprob"/>
          </xsl:for-each>
        </td>
      </tr>
    </table>
  </body>
</html>
</xsl:template>

<xsl:template match="proyecto">
<h1>
<xsl:value-of select="@titulo" />
<p>
<font face="arial, geneva, lucida sans unicode, helvetica" size="-1"> OBJETIVO GENERAL:
<xsl:value-of select="@obj_gen" />
<h2 style="padding-left: 15px; background-color: Chocolate; color: #ffff; font-family: Verdana, Arial; font-size: 14pt; text-align: justify; font-style: italic; letter-spacing: 0.5em"> ESTADO:<xsl:value-of select="@estado" /> </h2>
FECHA INICIO:<xsl:value-of select="@fecha_inicio" />
<p> FACULTAD:<xsl:value-of select="@facultad" /> </p>
<p> PROYECTO CURRICULAR:<xsl:value-of select="@proj_curricul"/> </p>
<p> FECHA APROBACION:<xsl:value-of select="@fecha_aprob"/> </p>
</font>
</p>
</h1>
</xsl:template>

<xsl:template match="obj_esp">
<h1 style="padding-left: 15px; background-color: #000000; color: #ffff; font-family: Verdana, Arial; font-size: 12pt; text-align: justify; font-style: italic; letter-spacing: 0.5em"> OBJETIVOS ESPECÍFICOS< /h1>
<h4>
<xsl:apply-templates/>
</h4>
</xsl:template>

</xsl:stylesheet>
```

Language Transformation), un lenguaje de transformación de hojas de estilo a otros códigos, en este caso a HTML, que es leído por cualquier navegador actual.

La Figura 1 presenta parte del documento visualizado en un navegador WEB (Internet Explorer 5.5) con el formato dado por la hoja de estilo, para este caso se usó XSLT para transformar automáticamente el documento XML a código HTML y así permitir su visualización a través del navegador en la máquina cliente.

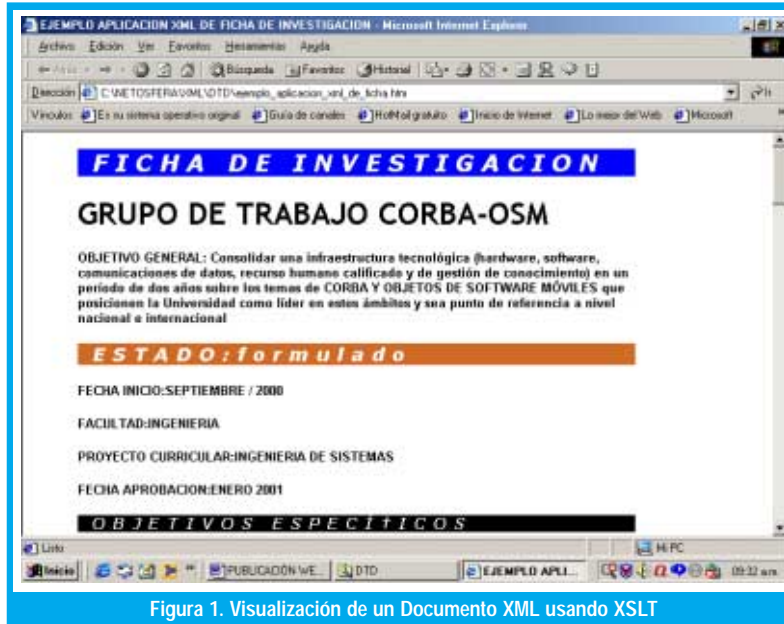


Figura 1. Visualización de un Documento XML usando XSLT

IV. PROPUESTA PROCEDIMENTAL PARA LA PUBLICACIÓN WEB DE TRABAJOS DE INVESTIGACIÓN EN UNA INSTITUCIÓN UNIVERSITARIA

Como lo ilustra el flujograma de la Figura 2, la institución debe determinar el estándar estructural de los documentos de investigación que requiere publicar, en esta labor se generarán los repositorios de DTDs para validar contra éstos cualquier información que quiera ir a la WEB; en el caso ejemplo, la Universidad podría haber definido **ficha_proyecto.dtd** como su modelo estructural de las fichas resumen de proyectos de investigación de Grupos de Trabajo.

Un segundo paso es permitir que los productores de información, en este caso: docentes, estudiantes, grupos de trabajo, investigadores, etc..., tengan acceso al formato que deben diligenciar soportado en etiquetado XML. Se podría brindar acceso a este formato a través de la WEB (haciendo una descarga del mismo) o suministrarlo a través de las coordinaciones de carrera en medio electrónico. Así mismo, se podría entregar un corto instructivo y un programa (como Microsoft XML Notepad, es software de uso

libre) que facilite la introducción de contenidos al formato XML a través de un árbol jerárquico que dé acceso a los nodos que conforman la estructura del documento, basado en el modelo **DOM (Document Object Model)**[2], tal como lo muestra la Figura 3; también se podría facilitar un analizador de documentos XML que permita, previo a la entrega, validar por parte del productor de información su documento contra el DTD correspondiente.

El tercer paso corresponde a la entrega del documento XML (diligenciado) al responsable de la publicación WEB (Web Master, bibliotecólogo, administrador de la red UDN, etc...); éste procedería a validarlo contra el DTD respectivo y, como es su responsabilidad tener diseñadas e implementadas las hojas de estilo en XSL, ejecuta un programa transformador a HTML generando automáticamente el contenido (etiquetado para su presentación) de la nueva página, inserta el enlace en la página principal o nodo desde donde se accede a los documentos en cuestión y queda publicado un nuevo

trabajo. En el caso ejemplo de este escrito, el documento **proyectos.xml** es entregado al publicador WEB que agregará una línea de enlace a la hoja de estilos respectiva (Ver Sección III) y a través de un programa de conversión a HTML que soporte XSLT genera la página HTML, la encadena agregando un enlace hipertexto a la página nodo que enlaza con las fichas resumen y quedará lista para ser visualizada; si los navegadores ya incorporan lectura directa de documentos XML el paso de transformación sería innecesario, sin olvidar que hay que asociar la hoja de estilo al documento XML.

V. ALGUNAS CONSIDERACIONES DE IMPLEMENTACIÓN

Construir bases de datos WEB puede ser bastante costoso si se trabaja con herramientas comerciales de ciertas casas de software; un repositorio de documentos consultables a través de Internet con el procedimiento aquí propuesto requiere (fuera de la plataforma de servidores Internet, ya existente en la mayoría de universidades):

- Un editor de DTDs que puede ser un editor de texto convencional o una herramienta de software libre.

La institución debe determinar el estándar estructural de los documentos de investigación que requiere publicar, en esta labor se generarán los repositorios de DTDs para validar contra éstos cualquier información que quiera ir a la WEB.

- Un editor de documentos XML, que también existen en versiones libres o se puede usar un simple editor de texto.
- Un analizador XML que valide frente a un DTD, también se encuentran versiones de uso libre.
- Un convertidor a HTML que soporte XSLT, se encuentran algunas herramientas de uso libre que logran la transformación de manera bastante flexible y transparente a código estándar HTML.

En Internet se encuentra bastante software comercial y libre que permite desarrollar lo aquí abordado; el software libre, en su mayoría intenta cumplir las especificaciones que el W3C ha promulgado, no obstante hay que ir con cuidado con estas herramientas. Algunas herramientas útiles pueden ser:

- Cooktop 2.200[3]
- Easy XML[4]
- Microsoft XML Notepad[5]
- XML Styler[6]
- StyleAssistant[7]

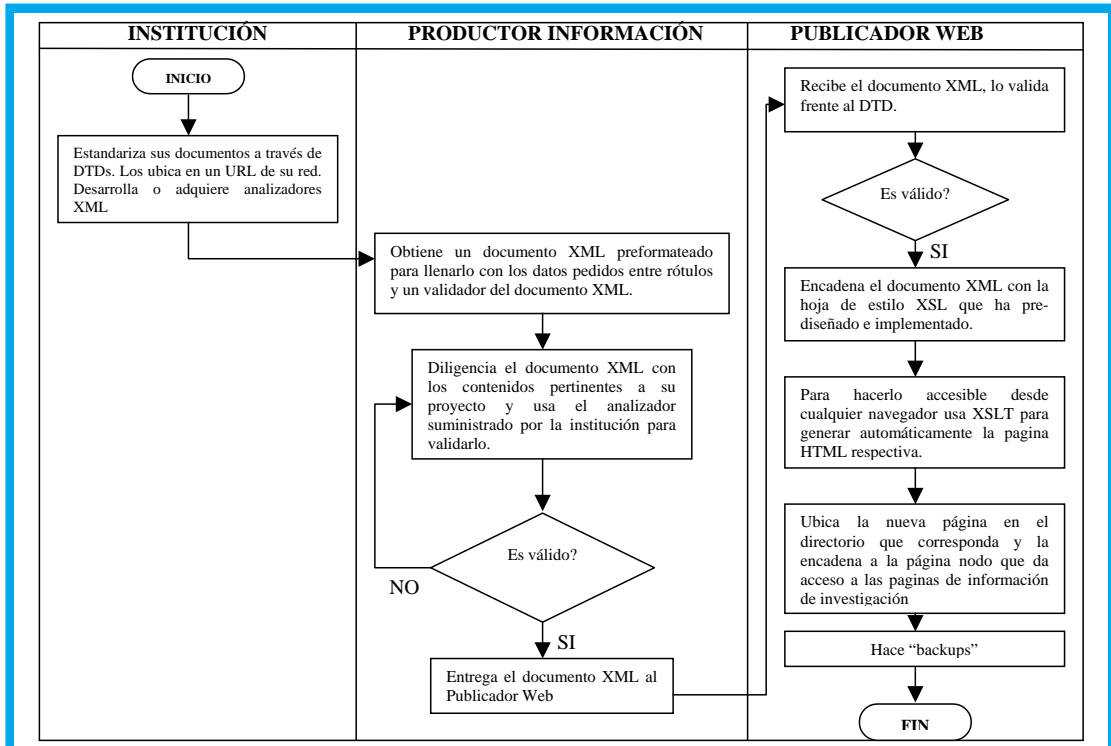


Figura 2. Flujograma de Procedimiento Publicación WEB

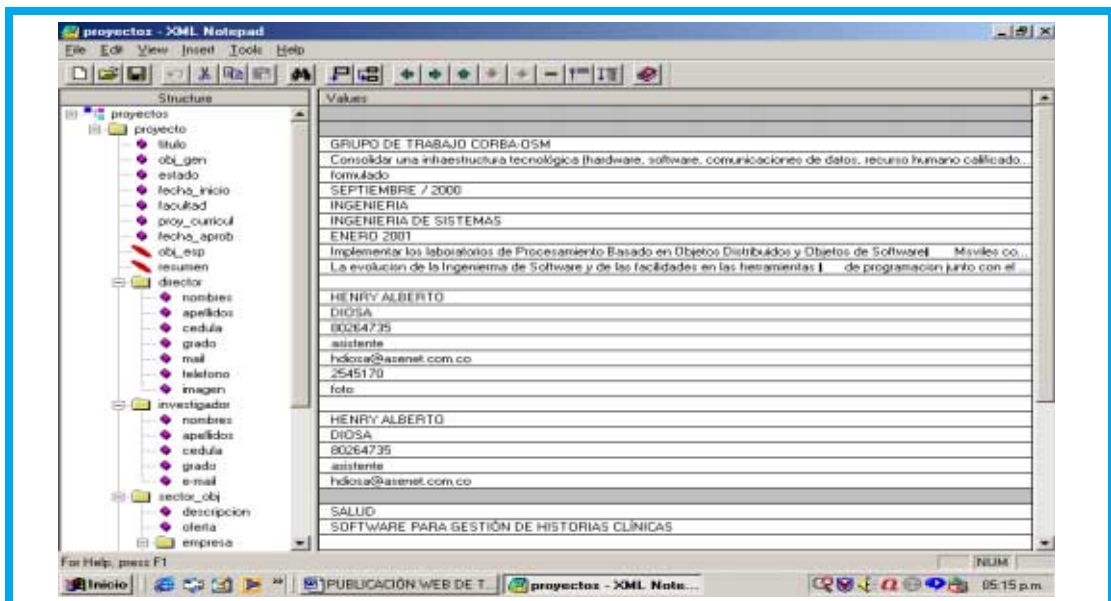


Figura 3. Herramienta para Introducir Datos Entre Rútilos XML como un árbol DOM

La lectura directa de documentos XML por parte de los navegadores, sin conversiones intermedias, facilitarán el proceso aquí tipificado y darán a XML el espaldarazo para convertirse en el lenguaje de publicación WEB por excelencia.

VI. CONCLUSIONES

Estandarizar la estructura de los documentos que una organización maneja propende por una administración más eficiente de la misma y proyecta un manejo organizado de la información. El ejemplo aquí expuesto puede generalizarse a otro tipo de información que requiere socializarse de manera fácil y a bajo costo; las DTDs son una muy buena herramienta para lograr esta normalización, requiere un estudio concienzudo y una buena abstracción del alcance estructural de los contenidos; no obstante, es sencillo para organizaciones que han establecido normas que aplican en alguna medida a sus documentos. El ejemplo aquí expuesto, podría optimizarse agregando atributos del tipo ID, ID_REF o IDREFS a ciertos elementos, esto permitiría efectuar búsquedas y normalizar mejor los datos. XML permite personalizar el etiquetado, aspecto que no podía lograrse con HTML; este enfoque de metalenguaje aplicado a dominios de problema específicos permite construir bases de datos WEB donde las búsquedas pueden ser más eficientes y menos engorrosas. CSS (Cascade Style Sheet) y sobre todo XSL, permiten jugar con diversos tipos de presentación, personalizadas y adaptadas a la imagen institucional sin importar el contenido.

De acuerdo a las anteriores consideraciones se pueden vislumbrar oportunidades de trabajo en:

- Definir DTDs o estandarizar la estructura de contenidos; aquí el modelo DOM (Document Object Model)[8] jugará un papel importante para facilitar la visualización y estructuración de documentos como árboles con nodos jerarquizados de acuerdo a su posición y composición en cada documento. Una de las tecnologías que propone un mayor alcance de las DTDs es XML Schema[9], propuesta por Microsoft y en estudio para ser estándar por parte del W3C, ya que maneja una gama más amplia y estricta de tipos de datos; además busca soportar la orientación a objetos.
- Elaborar documentos XML válidos y bien formados; se requieren analizadores tanto a nivel de validez como de buena formación.
- Usar repositorios de documentos XML como bases de datos WEB requerirá indexar de manera similar a lo que se logra con los manejadores de bases de datos, desarrollar motores de búsqueda y algún nivel transaccional sobre estos datos.
- La lectura directa de documentos XML por parte de los navegadores, sin conversiones intermedias, facilitarán el proceso aquí tipificado y darán a XML el espaldarazo para convertirse en el lenguaje de publicación WEB por excelencia.

- El desarrollo de hojas de estilo atractivas, amigables y que hagan la interfaz del lado cliente más útil y más flexible hacen que esta tecnología tenga mucho futuro. Como ejemplo, el diseñador gráfico podrá vender sus propuestas a instituciones o personas y no responsabilizarse de los contenidos a publicar.
- El W3C viene trabajando en estandarizar XML para animación 3D[10], para comunicación por voz con dispositivos[11] y otras iniciativas a aplicar en diversos campos como: matemáticas, medicina, etc...que le dan aún más fuerza a esta tecnología WEB.

Estos son algunos de los aspectos que hacen prometedores para el ámbito WEB estos avances. Hay que tener claro que alrededor de ver la información surgen otro tipo de problemas: ¿cómo buscar de manera fácil y amigable?, ¿cómo ordenar?, ¿cómo eliminar y actualizar datos?, ¿cómo proveer seguridad y privacidad, en algunos casos?¿cómo administrarla?, ¿cómo automatizar más el proceso después de adquirir la cultura de publicación WEB usando estas tecnologías?,etc... Lo que si es cierto es que después de tener la información estructurada y bien organizada será más fácil solucionar cualquier problema adicional.

REFERENCIAS

- [1] <http://www.w3c.org>
- [2] XSLT se ha venido estandarizando a través de la Recomendación para XSLT 1.0 publicada el 16 de Noviembre de 1999 por el W3C junto con el borrador de trabajo de la misma organización, que mejora y reemplaza la versión 1.1. y que fue liberada el 20 de Diciembre de 2001.
- [3] Editor de archivos .dtd, .xml y .xsl; incluye XSLT. Este es un ambiente de desarrollo y colaboración XML/XSL, de versión libre y desarrollado por Víctor Pavlov.
- [4] Herramienta para construcción rápida de aplicaciones XML de TanyiTech Systems. Se puede obtener más información en: <http://www.tanyitech.com>
- [5] Editor de archivos XML basado en DOM; se puede descargar de <http://msdn.microsoft.com/xml/notepad/intro.asp>
- [6] Herramienta de la empresa Arbortext . Se puede obtener de: <http://www.arbortext.com/xmlstyler>
- [7] Generador de hojas de estilo CSS desarrollado por Dr. Thomas Meinike. Se puede obtener desde <http://www.StyleAssistant.de>
- [8] <http://www.w3.org/TR/REC-DOM-Level-1>
- [9] <http://www.w3.org/1999/05/06-xmlschema-1>
- [10] <http://www.flatland.com>
- [11] <http://www.voxml.com>

Henry Alberto Diosa

Ingeniero de Sistemas de la Universidad Nacional de Colombia, Msc. Teleinformática de la Universidad Distrital Francisco José de Caldas, profesor facultad de Ingeniería Universidad Distrital Francisco José de Caldas. Actualmente es Ingeniero Desarrollador y Gerente de Netosfera Ltda. Coordinador Grupo de Trabajo CORBA-OSM hdiosa@atlas.udistrital.edu.co