# ARQUITECTURA BÁSICA DE UN NAVEGADOR DVB-HTML PARA MÚLTIPLES TERMINALES


ANEXO G

**José Wilmer Castillo Obando**
**Flavio Andrés Martínez Erazo**

Director
Ing. RODRIGO ALBERTO CERÓN MARTÍNEZ
Asesores
Ing. VICTOR MANUEL MONDRAGÓN MACA
Ing. FRANCO ARTURO URBANO ORDOÑEZ

Universidad del Cauca
**Facultad de Ingeniería Electrónica y Telecomunicaciones**
**Departamento de Telemática**
Línea de investigación: Sistemas telemáticos a la tele-educación
Popayán, Junio de 2009

**TABLA DE CONTENIDO**

# ANEXO G. CARACTERÍSTICAS DEL NAVEGADOR

Este anexo presenta las características del navegador EDiTVBW basado en el estándar DVB-HTML, creado a partir de la arquitectura planteada en el capítulo 4 del documento de monografía. Además, presenta los documentos descriptivos de aplicaciones DVB-HTML soportados y sus características.

## G.1 DTD del Navegador EDiTVBW

La definición de tipo de documento (DTD) es una descripción de estructura y sintaxis de un documento XML. Su función básica es la descripción del formato de datos, para usar un formato común y mantener la consistencia entre todos los documentos que utilicen la misma DTD. De esta forma, dichos documentos, pueden ser validados y conocer la estructura de los elementos y la descripción de los datos que trae consigo cada documento, además, pueden compartir la misma descripción y forma de validación dentro de un grupo de trabajo que usa el mismo tipo de información.

Debido a los problemas presentados con el DTD de la especificación MHP 1.2 como es mencionado en la sección 5.1.2 (paquete dom) del documento de monografía, es presentado a continuación el DTD utilizado para las aplicaciones DVB-HTML ejecutadas en el navegador:

```
<!DOCTYPE part-dvb-html [
<!ATTLIST div id ID #IMPLIED>
<!ATTLIST div top CDATA #IMPLIED>
<!ATTLIST div right CDATA #IMPLIED>
<!ATTLIST div width CDATA #IMPLIED>
<!ATTLIST div heigh CDATA #IMPLIED>
<!ATTLIST div font-size CDATA #IMPLIED>
<!ATTLIST div font-family CDATA #IMPLIED>
<!ATTLIST div text-align CDATA #IMPLIED>
<!ATTLIST div color CDATA #IMPLIED>
<!ATTLIST div opacity CDATA #IMPLIED>
<!ATTLIST div background-image CDATA #IMPLIED>
<!ATTLIST div background-color CDATA #IMPLIED>

<!ATTLIST p id ID #IMPLIED>
<!ATTLIST p font-size CDATA #IMPLIED>
<!ATTLIST p font-family CDATA #IMPLIED>
<!ATTLIST p text-align CDATA #IMPLIED>
<!ATTLIST p color CDATA #IMPLIED>

<!ATTLIST img id ID #IMPLIED>
<!ATTLIST img width CDATA #IMPLIED>
<!ATTLIST img heigh CDATA #IMPLIED>
<!ATTLIST img font-size CDATA #IMPLIED>
<!ATTLIST img font-family CDATA #IMPLIED>
<!ATTLIST img src CDATA #IMPLIED>

<!ATTLIST a id ID #IMPLIED>
<!ATTLIST a font-size CDATA #IMPLIED>
<!ATTLIST a font-family CDATA #IMPLIED>
```

```
<!ATTLIST a text-align CDATA #IMPLIED>
<!ATTLIST a color CDATA #IMPLIED>
<!ATTLIST a nav-up CDATA #IMPLIED>
<!ATTLIST a nav-down CDATA #IMPLIED>
<!ATTLIST a nav-left CDATA #IMPLIED>
<!ATTLIST a nav-right CDATA #IMPLIED>
<!ATTLIST a nav-index CDATA #IMPLIED>
<!ATTLIST a nav-first CDATA #IMPLIED>
<!ATTLIST a onfocus CDATA #IMPLIED>
<!ATTLIST a onblur CDATA #IMPLIED>
<!ATTLIST a href CDATA #IMPLIED>

<!ATTLIST form id ID #IMPLIED>
<!ATTLIST form top CDATA #IMPLIED>
<!ATTLIST form right CDATA #IMPLIED>
<!ATTLIST form width CDATA #IMPLIED>
<!ATTLIST form heigh CDATA #IMPLIED>
<!ATTLIST form action CDATA #IMPLIED>
<!ATTLIST form method CDATA #IMPLIED>

<!ATTLIST object id ID #IMPLIED>
<!ATTLIST object top CDATA #IMPLIED>
<!ATTLIST object right CDATA #IMPLIED>
<!ATTLIST object width CDATA #IMPLIED>
<!ATTLIST object heigh CDATA #IMPLIED>
>
<!ATTLIST object data CDATA #IMPLIED>
<!ATTLIST object type CDATA #IMPLIED>
]>
```

## G.2 Etiquetas soportadas por el navegador EDiTVBW

Las etiquetas que maneja el navegador pueden tener los atributos en ella misma (aunque no lo establece el estándar) o en la hoja de estilos. El navegador soporta posiciones fijas en sus componentes, por tanto, los elementos de estructura (*div*) deben tener atributos como *right, top, width, heigh* con el número de pixeles. Las etiquetas soportadas con algunas limitaciones son las siguientes:

- **html**
- **body**
- **head**
- **link**
- **script**
- **p**
- **a**
- **img**
- **object**
- **div**
- **form**
- **input**

Pueden existir más etiquetas en el documento XML, sin embargo, estas son ignoradas y no tienen ninguna representación. Las etiquetas clasificadas de estructura son: *html, body, head, div;* para texto y contenidos (*p, a, img, input*).

Todas las etiquetas para texto o contenido deben tener un nodo padre de estructura, más concretamente un *div*. editvBW implementa el algoritmo de obtención de atributos en cascada, sólo con un nivel de profundidad para obtener valores para: *top, right, background-color, opacity, width* y *heigh.*

### G.2.1 Etiqueta <html>

Define el *root* de documento, sus dos nodos hijos son <*head*> y <*body*>, no tiene ningún atributo

### G.2.2 Etiqueta <head>

A esta etiqueta pertenecen <*scrip*> y <*link* >

### G.2.3 Etiqueta <script>

En su atributo *src* debe tener la URL al un archivo con extensión *.js* donde es encuentrado el código ECMAScript. En el atributo *language* el valor de “JavaScript":
<scrip language="JavaScript" src="url"/>

### G.2.4 Etiqueta <link>

En el atributo *href* debe tener la URL completa a un archivo de hojas de estilo, con la extensión *.css* y los dos atributos con los valores mostrados a continuación:
<link rel="stylesheet" type="text/css" href="url"/>

### G.2.5 Etiqueta <body>

Contiene todos las etiquetas a ser representadas, soporta el atributo background-image, en el cual debe estar la url al una imagen iFrame, para desplegarse en la capa de fondo.

### G.2.6 Etiqueta <object>

Etiqueta para el despliegue de contenidos de video, tiene los atributos *id* y *type* con el valor video/mpeg (type="video/mpeg"). Por medio de la etiqueta <*param*> (dentro de <*object*>) son establecidas las URLs para los flujos a ser reproducidos, también deben tener las siguientes características:

- **right**
- **top**
- **width**
- **heigh**

La etiqueta <*param*> contiene la URL del flujo de video. En su atributo *name* tiene el nombre de “*locator*” y en el atributo *value* tiene el valor de la URL al flujo de video.

### G.2.7 Etiqueta <img>

Etiqueta para el soporte de imágenes. En su atributo *src* debe tener la URL completa a la imagen en los formatos gif, png o jpeg, además, debe tener los atributos de *width* y *high*. Los demás atributos los hereda de su etiqueta contenedora padre.

### G.2.8 Etiqueta <p>

Es una etiqueta de texto, despliega el texto entre sus etiquetas de apertura y cierre, dentro sólo puede haber texto sin ningún otro tipo de etiqueta o contenido XML, debe

tener los valores para los siguientes atributos, ya sea dentro de la etiqueta directamente o en la hoja de estilos:

- **font-size**
- **font-family**

### G.2.9 Etiqueta <a>

Es una etiqueta de texto, en su atributo *href* tiene la URL donde el navegador empieza la carga de una nueva página DVB-HTML. Esta URL debe estar completa, incluyendo el protocolo, el servidor o dirección DVB, además, el camino al documento.

- **dvb://x.x.x.x/camino/recurso**
- **http://servidor/camino/recurso**
- **file://camino/recurso**

El documento para que sea identificado como una página nueva debe tener extensión .html, .xml o .jsp. Estas dos últimas agregadas para facilitar el desarrollo y uso de un servidor con soporte para jsp.

La siguiente información es necesaria para la etiqueta <*a*>, puede estar en la etiqueta o en la hoja de estilos (excepto las atributos *nav* que sólo pueden estar en la etiqueta directamente).

- **nav_index**
- **nav-right**
- **nav-left**
- **nav-down**
- **nav-up**

Los siguientes son opcionales:

- **id**
- **onfocus**
- **onblur**

### G.2.10 Etiqueta <form>

Identifica al formulario, dentro de ella estan las etiquetas <*input*>, debe tener los atributos de *action* con la URL del recurso al cual es dirigida la información capturada por el formulario, además del atributo *method* con el valor GET. El navegador editvBW sólo soporta envió por el método GET. A continuación un ejemplo.

<form action="http://172.16.200.200:8080/editvServer/form1.jsp" method="GET">

### G.2.11 Etiqueta <input>

Etiqueta de tipo contenido, los tipos soportados (valores del atributo *type*) son *text, radio, check, submit*. También debe tener en el atributo *name* el nombre y en *value* el valor. La pareja nombre-valor será enviada al servidor por el método GET. Un conjunto de etiquetas *input* de tipo radio que tengan el mismo nombre las hace excluyentes entre sí. Las etiquetas de navegación también deben estar incluidas (directamente en la etiqueta)

- **nav_index**
- **nav-right**

- **nav-left**
- **nav-down**
- **nav-up**
- **id** (opcional)

Ejemplo:

<input type="text" name="nombre" nav-up="1" nav-down="1" nav-left="1" nav-right="1" nav-index="0"/>

### G.2.12 Etiqueta 

Etiqueta que determina la estructura, opcionalmente tiene id. Los atributos que tiene y son heredados por las etiquetas de contenidos que alberga son:

- **top**
- **right**
- **background-color**
- **opacity**
- **width**
- **heigh**

## G.3 Documento CSS

El formato para el documento de hojas de estilos es basada en la recomendación de la W3C para hojas de estilo en cascada. Todos los componentes deben tener la siguiente forma:

```
nombre-etiqueta
{ (corchete apertura)
Atributo: (dos puntos) valor; (punto y coma)
Atributo2: (dos puntos) valor; (punto y coma)
.
.
.
}(corchete cierre)
```

Para especificar una etiqueta con un id:

```
nombre-etiqueta # (numeral) id
{ (corchete apertura)
Atributo: (dos puntos) valor; (punto y coma)
Atributo2: (dos puntos) valor; (punto y coma)
.
.
.
}(corchete cierre)
```

Características:

- No es soportado el uso del *id* de la etiqueta directamente.

- El valor y el formato dependen del tipo de atributo. Si el atributo es de color el valor tiene el formato: *rgb(a,b,c)*; donde a, b, c son enteros entre 0 y 256 que representa rojo verde y azul respectivamente.

- La posición o longitud en atributos como *right, left, width, heigh* van en pixeles de la forma *apx* (donde a es el numero de pixeles).

- Los valores que sean textos como el tamaño de fuente tienen la siguiente forma: **font-*size*: a**; donde a es un numero entero.

- Los valores de texto como tipo de fuente () o alineamiento van sin ningún texto o terminación adicional (atributo:texto;).

- El valor de la trasparencia es un número entre 0 y 1, tiene el siguiente formato ***opacity*: a**; donde a es un numero entre 0 y 1.

## G.4 Documento ECMAScript

El soporte de ECMAScript es demostrativo y no es guiado en el estándar para tener referencia de los objetos del documento. Aunque el navegador tiene acceso a los atributos del documento mediante DOM nivel 2. La etiqueta que incluya una función en el atributo *onfocus* u *onblour,* generará la ejecución de la función ECMAScript que tenga asociada. Al final de la función ECMAScript es posible solicitar la actualización del documento desplegado, la actualización es realizada al documento completo.

La obtención de la referencia al objeto dentro de una función, es hecha mediante la invocación del *singlenton* bean *BeanDOM.getInstance()*. De este objeto es posible la obtención del *Objeto de la Aplicación* con el método *getDOM()*. Asi, con la referencia es posible realizar cambios dentro del documento siguiendo los métodos DOM nivel 2. Para que el nuevo documento sea desplegado es necesario invocar al método *actualizar()*, éste envía de nuevo el *Objeto de la Aplicación* al módulo de Visualización para que lo despliegue.