



INSERTAR AUDIO EN LA WEB

Contenido

INSERTAR AUDIO EN LA WEB	1
Conceptos básicos del sonido digital	2
Frecuencia.	2
Tasa de muestreo (sample rate).	2
Resolución (bit resolution)	2
Velocidad de transmisión (bitrate).....	2
CBR/VBR	2
Códec.....	3
Decibelio.....	3
FORMATOS	3
El formato MP3	3
Ogg:	4
Real Audio:	4
Windows Media Audio	4
Wav o wave	5
vqf:.....	5
Free Lossless Audio Codec (FLAC)	5
Formato MIDI	6
CONVERSIÓN DE FORMATOS	6
Optimización de archivos de audio	7
INSERTANDO AUDIO EN UNA WEB	8
Audio de fondo (en HTML4)	8
Descargar/ejecutar audio por el usuario (en HTML4)	9
Insertar sonido (HTML5)	9
ACTIVIDAD	14
CONSIDERACIONES PARA EL USO DE AUDIO EN UN SITIO WEB	15



Conceptos básicos del sonido digital

Frecuencia.

Es el número de vibraciones por segundo que da origen al sonido analógico. El espectro de un sonido se caracteriza por su rango de frecuencias. Ésta se mide en Hertzios (Hz). El oído humano capta sólo aquellos sonidos comprendidos en el rango de frecuencias 20 Hz y 20.000 Hz.

Tasa de muestreo (sample rate).

Un audio digital es una secuencia de ceros y unos que se obtiene del muestreo de la señal analógica. La tasa de muestreo o sample rate define cada cuánto tiempo se tomará el valor de la señal analógica para generar el audio digital. Esta tasa se mide en Hertzios (Hz). Por ejemplo: 44100 Hz. nos indica que en un segundo se tomaron 44100 muestras de la señal analógica de audio para crear el audio digital correspondiente. Un audio tendrá más calidad cuanto mayor sea su tasa de muestreo. Algunas frecuencias estándares son 44100 Hz., 22050 Hz., y 11025 Hz.

Resolución (bit resolution)

Es el número de bits utilizados para almacenar cada muestra de la señal analógica. Una resolución de 8-bits proporciona 256 (2^8) niveles de amplitud, mientras que una resolución de 16-bits alcanza 65536 (2^{16}). Un audio digital tendrá más calidad cuanto mayor sea su resolución. Ejemplo: El audio de calidad CD suele ser un sonido de 44.100 Hz – 16 bits – estereo.

Velocidad de transmisión (bitrate)

El bitrate define la cantidad de espacio físico (en bits) que ocupa un segundo de duración de ese audio. Por ejemplo, 3 minutos de audio MP3 a 128kBit/sg, ocupa 2,81 Mb de espacio físico ($3\text{min} \times 60\text{ seg/min} \times 128\text{ kBit/seg} = 23040\text{ kBits} \rightarrow 23040\text{ kBits} \times 1024\text{ bits/Kbit} : 8\text{ bits/bytes} : 1024\text{ bytes/Kbytes} : 1024\text{ Kbytes/Mbytes} = 2,81\text{ MBytes}$ ó Mb). Por ejemplo en los audios en formato MP3 se suele trabajar con bitrates de 128 kbps (kilobits por segundo). El audio tendrá más calidad cuanto mayor sea su bitrate y el archivo que lo contiene tendrá mayor peso. Esta magnitud se utiliza sobre todo en el formato MP3 de audio más destinado a la descarga por Internet.

CBR/VBR

Constant/Variable Bitrate. CBR indica que el audio ha sido codificado manteniendo el bitrate constante a lo largo del clip de audio mientras que VBR varía entre un rango máximo y mínimo en función de la tasa de transferencia.



Códec.

Acrónimo de "codificación/decodificación". Un códec es un algoritmo especial que reduce el número de bytes que ocupa un archivo de audio. Los archivos codificados con un códec específico requieren el mismo códec para ser decodificados y reproducidos. El códec más utilizado en audio es el MP3.

Decibelio.

Unidad de medida del volumen o intensidad de un sonido. El silencio o ausencia de sonido se cuantifica como 0 dB y el umbral del dolor para el oído humano se sitúa en torno a los 130,140 dB.

FORMATOS

Al hablar del audio en Internet, el formato **más conocido es mp3**. Éste es un formato de compresión de audio digital patentado que usa un algoritmo con **pérdida** para conseguir un **menor tamaño** de archivo. Es un formato de audio común usado para música tanto en ordenadores como en reproductores de audio portátil. Más concretamente, **mp3** fue desarrollado por el **Moving Pi.cture Experts Group** (MPEG) para formar parte del estándar MPEG-I y del posterior y más extendido MPEG-2. Un **mp3** puede comprimirse usando una mayor o menor **tasa de bits por segundo**, resultando directamente en su mayor o menor calidad de audio final, así como en el tamaño del archivo resultante.

La tecnología MP3 fue desarrollada principalmente por el instituto de investigación alemán Fraunhofer y por los laboratorios de AT&T Bell, que pasaron a formar parte de Lucent en 1996. Alcatel se hizo con Lucent el año pasado. Microsoft pagó 16 millones de dólares a Fraunhofer por las patentes relacionadas con el uso de MP3, pero Alcatel-Lucent argumenta que dos de las patentes fueron desarrolladas por los laboratorios Bell antes de que se unieran con Fraunhofer y, por lo tanto, no están incluidas en la licencia de Microsoft, algo que la multinacional niega.

Es importante destacar que, en lo que respecta al desarrollo web y al uso del audio para ser interpretado por los diferentes navegadores, **mp3** no es único y universal, es decir, hay más formatos iguales o mejores y no todos los navegadores lo soportan (por ejemplo, Firefox no interpreta **mp3** por sí solo). Por lo tanto, hay otras posibilidades de formatos que hay que conocer, sobre todo para, como luego veremos, permitir que la reproducción de audio sea lo más universal posible entre todos los navegadores actuales (al margen de sus batallas empresariales).

El formato MP3 (MPEG 1 Layer 3) fue creado por el Instituto Fraunhofer, y por su extraordinario grado de compresión y alta calidad está prácticamente monopolizando el mundo del audio digital.



- Es ideal para publicar audios en la web. Se puede escuchar desde la mayoría de reproductores.
- La transformación de WAV a MP3 o la publicación directa de una grabación en formato MP3 es un proceso fácil y al alcance de los principales editores de audio.
- Tiene un enorme nivel de compresión respecto al WAV. En igualdad del resto de condiciones reduciría el tamaño del archivo de un fragmento musical con un factor entre 1/10 y 1/12.
- Presentan una mínima pérdida de calidad.

Los siguientes formatos no son los únicos que existen, pero sí son los más conocidos en Internet, y los más usados en la distribución de audio en sitios web.

Ogg: es un formato válido para audio y vídeo, desarrollado por la Fundación *Xiph.org*. Su principal ventaja con respecto a *mp3* es que es un formato libre de patentes y abierto diseñado para dar un alto grado de eficiencia en el *streaming* y la compresión de archivos. De manera incorrecta, muchas veces los archivos *ogg* se les llama *Vorbis* ya que éste fue el primer códec desarrollado para leer este formato. Sin embargo, hay muchos otros códec además de Vorbis que interpretan *ogg*. Actualmente, Firefox y Chrome interpretan *ogg*.

- El formato OGG ha sido desarrollado por la Fundación Xiph.org.
- Es el formato más reciente y surgió como alternativa libre y de código abierto (a diferencia del formato MP3).
- Muestra un grado de compresión similar al MP3 pero según los expertos en música la calidad de reproducción es ligeramente superior.
- No todos los reproductores multimedia son capaces de leer por defecto este formato. En algunos casos es necesario instalar los códec o filtros oportunos.
- El formato OGG puede contener audio y vídeo.

Real Audio: sin duda, este formato es el rey del *streaming*. La gran mayoría de las páginas que usan *streaming* utilizan el formato Real Audio para difundir sus contenidos. RealNetworks es la compañía que ha creado Real Audio y, por supuesto, también ha creado un reproductor (*player* en inglés) para su formato, el RealPlayer (del que existe una versión gratuita y otra comercial).

Windows Media Audio (*wma*): es un formato desarrollado por Microsoft. Ofrece gran calidad de sonido en un tamaño reducido. Y como ventaja para las compañías y los músicos, el formato de Microsoft incluye un sistema de gestión de los derechos de autor para evitar la piratería y la creación de copias no autorizadas de las canciones. Por las características de calidad y tamaño *wma* resulta muy adecuado para la reproducción de archivos en *streaming*. El reproductor que Microsoft ha desarrollado para escuchar su formato de audio es el Windows Media Player que se encuentra disponible en su web. La desventaja de *wma* es que, al ser de



Microsoft, el navegador IExplorer lo soporta, pero no se puede asegurar en el resto.

Wav o wave (*WAVEform audio file format*): es un formato de audio que se suele utilizar sin compresión de datos. wav es propiedad de Microsoft y de IBM. Este formato es muy conocido en entornos Windows, pero no se suele usar en Internet por ocupar mucho espacio al ser usado sin compresión.

- El formato WAV (Wave Form Audio File) es un archivo que desarrolló originalmente Microsoft para guardar audio. Los archivos tienen extensión *.wav
- Es ideal para guardar audios originales a partir de los cuales se puede comprimir y guardar en distintos tamaños de muestreo para publicar en la web.
- Es un formato de excelente calidad de audio.
- Sin embargo produce archivos de un peso enorme. Una canción extraída de un CD (16 bytes, 44100 Hz y estéreo) puede ocupar entre 20 y 30 Mb.
- Compresión: Los archivos WAV se pueden guardar con distintos tipos de compresión. Las más utilizadas son la compresión PCM y la compresión ADPCM. No obstante incluso definiendo un sistema de compresión, con un audio de cierta duración se genera un archivo excesivamente pesado.
- El formato WAV se suele utilizar para fragmentos muy cortos (no superiores a 3-4 segundos), normalmente en calidad mono y con una compresión Microsoft ADPCM 4 bits.

vqf: el formato de audio VQF *ha* sido desarrollado por la compañía Yamaha y presenta algunas ventajas respecto al *mp3*: mayor calidad de sonido y archivos de menor tamaño (se puede comprimir hasta un 30% más que *mp3*). El reproductor más asociado a este archivo es el Yamaha *SoundVQ*, aunque hay más que lo soportan. El mayor inconveniente del *vqf* es que no está muy extendido (por lo que muchos navegadores no lo soportan), además, su reproducción necesita de más recursos que cualquiera de los formatos anteriores.

Free Lossless Audio Codec (FLAC) es un códec de audio que permite que el audio digital sea comprimido sin pérdidas de tal manera que el tamaño del archivo de audio se reduce sin que se pierda ningún tipo de información. El audio digital comprimido por el algoritmo de FLAC típicamente se puede reducir de 50 a 60% de su tamaño original,³ y se descomprime en una copia idéntica de los datos de audio originales. La desventaja es que el archivo ocupa mucho más espacio del que se obtendría al aplicar un algoritmo de compresión con pérdida.

FLAC es un formato abierto con licencia libre de regalías y una implementación de referencia la cual es software libre. FLAC cuenta con soporte para etiquetado de metadatos, inclusión de la portada del álbum, y la búsqueda rápida.



Formato MIDI

- El formato MIDI (Musical Instrument Digital Interface = Interface Digital para Instrumentos Digitales) en realidad no resulta de un proceso de digitalización de un sonido analógico. Un archivo de extensión *.mid almacena secuencias de dispositivos MIDI (sintetizadores) donde se recoge qué instrumento interviene, en qué forma lo hace y cuándo.
- Este formato es interpretado por los principales reproductores del mercado: Windows Media Player, QuickTime, etc.
- Los archivos MIDI se pueden editar y manipular mediante programas especiales y distintos de los empleados para editar formatos WAV, MP3, etc. El manejo de estos programas suele conllevar ciertos conocimientos musicales.
- Los archivos MIDI permiten audios de cierta duración con un reducido peso. Esto es debido a que no guardan el sonido sino la información o partitura necesaria para que el ordenador la componga y reproduzca a través de la tarjeta de sonido.
- Se suelen utilizar en sonidos de fondo de páginas HTML o para escuchar composiciones musicales de carácter instrumental.
- El formato MIDI no permite la riqueza de matices sonoros que otros formatos ni la grabación a partir de eventos sonoros analógicos.

CONVERSIÓN DE FORMATOS

Un hándicap importante a la hora de trabajar con audio en páginas web es saber cómo implementa cada navegador los reproductores. No todos los navegadores tienen reproductores para todos los formatos, sino que hay preferencias y lo que un navegador puede reproducir otro no. Esta situación lo único que hace es complicar el trabajo del desarrollador, ya que éste debe intentar garantizar que los audios que inserte en un sitio web puedan ser reproducidos por todos los navegadores, y si eso no es posible, por aquellos más utilizados por el usuarios objetivo del portal.

La siguiente tabla muestra una comparativa de qué formatos son reproducidos por los 5 navegadores más destacados actualmente (bajo Windows). Esta lista no es definitiva, ya que la vertiginosa actualización de los navegadores hace que las prestaciones que ofrecen en una versión sean mejoradas o reducidas en las siguientes versiones.

FORMATO	EDGE	EXPLORER	FIREFOX	OPERA	CHROME	SAFARI
Ogg	NO	No	Si	Si	Si	No
Mp3	SI	Si	SI	SI	Si	Si
WAV	SI	No	Si	Si	Si	Si
FLAC	NO	NO	NO	NO	NO	NO
MIDI	NO	SI(Con bgsound)	NO	NO	NO	NO

Los valores de Tabla hay que tenerlos en cuenta a la hora de saber qué



formatos insertar como audio en un sitio web. Como se verá en los siguientes puntos, HTML5 ofrece una alternativa para insertar varios formatos de un audio dependiendo de cuál es el que soporta un determinado navegador. El desarrollador guarda el mismo audio en diferentes formatos, y en HTML5 se referencian todos ellos para que cada navegador elija el que puede reproducir.

Para poder tener un audio en varios formatos, es necesario usar herramientas de conversión. Estas herramientas suelen soportar una gran cantidad de formatos y suelen ser sensibles respecto a los parámetros de compresión en aquellos que lo soportan.

Herramientas de conversión hay muchas disponibles en Internet. Algunas gratuitas son:

Free Studio: es una potente herramienta para convertir todo tipo de archivos (vídeo, audio, imagen). Es fácil de usar y muy adecuada para un diseñador que trabaja con audio en la web. Solo versión para Windows.

Audacity: es una herramienta avanzada para grabar y editar. Es libre y de código abierto. Su funcionalidad es mucha, y entre ella está la de convertir archivos. Es muy adecuada si el diseñador desea crear sus propios sonidos. Para Linux y Windows.

También hay conversores *online*, servicios web a los que se le sube el archivo de audio, se selecciona el formato al que se quiere convertir y el resultado se envía por *email*. Un ejemplo de este tipo de herramientas es *Switchr*, sin embargo hay muchas más disponibles en Internet.

Enlaces de interés:

<http://www.dvdvideosoft.com/s/free.dvd.video-.software.htm>

<http://audacity.sourceforge.net>

<http://switchr.net>

Optimización de archivos de audio

Para optimizar el peso del archivo de audio será necesario utilizar un editor para reducir alguno o algunos de los siguientes parámetros:

1. **Tasa de muestreo.** Definir valores inferiores: 44100 Hz., 22050 Hz., 11025 Hz, etc.
2. **Resolución.** Establecer resoluciones más pequeñas: 32-bits, 16-bits, 8-bits, 4-bits, etc.
3. **Duración.** En ocasiones se puede utilizar un fragmento más corto que reproducido en bucle cubre el tiempo suficiente de acompañamiento musical. A éstos se les llama loops.
4. **Calidad estéreo/mono.** La reducción a calidad “mono” reduce considerablemente el peso del archivo. Por otro lado la calidad de reproducción “mono” para la mayoría de audios y de público es apenas perceptible.



5. **Formato.** Es preferible utilizar el formato MP3 u OGG en lugar del WAV por su potente factor de compresión y su aceptable calidad de audio.
6. **Factor de compresión.** El formato WAV admite distintos factores de compresión: PCM y ADPCM.

INSERTANDO AUDIO EN UNA WEB

Vistos los diferentes formatos de audio adecuados para Internet, cuáles son soportados por los navegadores más adecuados, y herramientas de conversión de formatos, en este punto se muestran diferentes maneras de introducir audio en una web con HTML.

Audio de fondo (en HTML4)

Poner un sonido de fondo para una página web usando etiquetas HTML no es la opción más versátil. Existen dos etiquetas que, en teoría, lo permiten - <bgsound> y <embed>.

-<bgsound> es una alternativa aparentemente adecuada pero con el inconveniente de que es solo entendible por Internet Explorer. Ojo no funciona con Microsoft Edge, Evidentemente, esto la hace muy poco portable por lo que no se recomienda su uso, salvo en situaciones muy concretas. Un ejemplo de uso para ejecutar de fondo un sonido *audios / Beethoven.wma* (solo válida para IExplorer):

```
<bgsound src="audios/Beethoven.wma" loop="2"
volume="20" />
```

Lo más importante de esta etiqueta es que para indicar el fichero a reproducir hay que utilizar el atributo src. Otros atributos de interés son loop, que permite que la canción vuelva a empezar cada vez que acaba (se le puede indicar cuántas veces se repetirá, con INFINITE no hay límite); delay: es un valor para retrasar el inicio de la música (debe ser un número positivo en segundos); volumen, determina la intensidad del sonido de fondo, con valores entre -10.000 (el más débil) a 0 (el más alto) .

- -<embed> es una alternativa para insertar complementos (*plugins*) de audio y vídeo que es compatible para todos los navegadores. Sin embargo, no está incluida dentro del estándar W3C por lo que su implementación depende de los navegadores, y no todos la interpretan igual. Además, dependiendo del formato del archivo, así lo interpretarán los navegadores. Por ejemplo, en Google Chrome no funciona el reproductor por defecto cuando la extensión es *wma* (Microsoft) pero sí con *mp3*. Un ejemplo de código es el siguiente:

```
<embed src="audios/Aretha.mp3" height="0"
type="audio/mpeg" loop="false"
controller="false">
```

El atributo *src* sirve para especificar la ruta del archivo (como en <bgsound>». Los audios insertados a través de esta etiqueta se reproducen automáticamente al cargarse la página, y se reproducen solamente una



vez. Esto puede cambiarse a través de los atributos *autostart* y *loop*. El atributo *autostart* indica si el archivo se reproducirá automáticamente al cargarse la página, y puede tomar los valores *true* o *false* (aunque no funciona correctamente para todos los navegadores) El atributo *loop* indica si el archivo se reproducirá continuamente en un bucle, y también puede tomar los valores *true* o *false*. Los atributos *width* y *height* sirven para especificar el tamaño de la consola de control de sonido (o video). Estos atributos pueden tomar como valor un número, que indica el tamaño en píxeles. Si no se especifican estos atributos, la consola de control de vídeo se mostrará con el tamaño más adecuado al tamaño del sonido. *Controller* indica si aparece o no la consola de controles del reproductor. *Type* es un atributo importante, que declara el tipo de fichero de audio que estamos usando, con lo que el navegador web puede ejecutar el complemento o *plugin* adecuado para la reproducción del fichero, Puede ser *audio / midi*, *audio / wav* o *audio / mpeg*, entre otros.

En cualquier caso, porque depende del navegador y del *plugin* necesario para su inclusión, ninguna de estas dos alternativas ofrece una solución interoperable entre todos los navegadores, por tanto, es poco recomendable su uso.

Descargar/ejecutar audio por el usuario (en HTML4)

La manera más sencilla de incluir sonidos es dejando al usuario la decisión de escucharlos o no en local. En realidad es más una opción *descargar* que *reproducir*. Para ello se puede colocar el nombre del archivo de sonido en una atributo *href* de la etiqueta <a>. El siguiente código muestra un ejemplo.

```
<a href="audios/Beethoven.wma">Audio de Beethoven<a>
```

En este ejemplo, dependiendo de la versión del navegador que se esté usando, se le da al usuario la opción de descargar el archivo en local o se carga un complemento (*plugin*) que lo ejecuta. Pero todo necesita de descargarse en local y ejecutarlo con un reproductor (*player*).

Insertar sonido (HTML5)

En HTML5 la inclusión de sonidos en páginas web intenta paliar muchos de los inconvenientes vistos con las etiquetas <bgsound> o <embed>. De alguna manera, HTML5 ofrece una alternativa más sencilla y eficaz para que el desarrollador pueda insertar audio. Sin embargo, la realidad es que cada navegador implementa de HTML5 lo que le quiere, por lo que tampoco con esta opción se consigue una solución universal.

La propuesta de HTML5 utiliza la etiqueta **<audio>**. Esta etiqueta es nueva en HTML5 y no existe para versiones de HTML4. Evidentemente, eso obliga a que los navegadores que la interpreten deban implementar HTML5 en general (y esta etiqueta en particular).

Los atributos de esta etiqueta son muy parecidos a los vistos para <embed>.



Autoplay especifica que el audio debería empezar automáticamente tan pronto como esté listo para reproducirse (el único valor de este atributo es *autoplay*). **Controls** indica que los controles de reproducción serán visibles (el único valor posible de este atributo es *controls*). **Loop** indica que el audio se repetirá automáticamente cuando termine (el único valor posible de este atributo es *loop*). **Preload**, especifica si el audio debería ser cargado cuando la página se carga o no, es decir, se empieza a cargar cuando el usuario lo quiera reproducir (el valor *auto* indica que se cargará todo el archivo cuando se cargue la página, el valor *meta* indica que solo se cargarán los metadatos del archivo y *none* que no habrá precarga). Por último, **src** especifica la URL de archivo de audio. Además de estos atributos, <audio> soporta también atributos globales de HTML5 y eventos.

Un ejemplo del uso de esta etiqueta para obtener un reproductor para un audio que se inicia solo, sería:

```
<audio src="audios/Aretha.mp3" controls="controls"
autoplay></audio>
```

Este ejemplo funciona correctamente en Explorer 9 y en Google Chrome 14. La apariencia de los reproductores es diferente, pero ambos reproducen el archivo *mp3* obedeciendo al atributo *autoplay*. Es importante destacar que el uso de HTML5 abre las puertas de navegadores de dispositivos móviles como iPad, iPhone y los basados en Android o Windows. El caso de iPad (que hasta la fecha) no interpreta archivos Flash, la etiqueta <audio> ofrece una alternativa muy elegante y sencilla para los desarrolladores.

Una mejora para hacer más versátil esta opción es usar la etiqueta <source> dentro de <audio>. De esta manera, el navegador intenta cargar la primera línea <source>. Si falla o no es soportada esa reproducción, entonces pasa a la siguiente. Por ejemplo, si Explorer, Safari y Google Chrome leen archivos *mp3* y Firefox lee archivos *ogg* (no soporta *mp3*), el siguiente código ofrece una solución para los cuatro navegadores. Cuando el navegador Firefox llegue al primer <source> y no pueda interpretar *mp3* entonces pasará al segundo con *ogg*. Sin embargo, Internet Explorer, Safari y Google Chrome se quedarán con el primer <source>. El siguiente ejemplo empieza definiendo el tipo de documento (<!DOCTYPE html>) que es: navegadores como Explorer lo necesitan para interpretar bien el contenido.

```
<!DOCTYPE html>
<html>
<head>
</head>
<body>
<audio controls="controls" preload="auto">
<source src="music.mp3" type="audio/mpeg"/>
<source src="music.ogg" type="audio/ogg"/>
El navegador no soporta el audio
</audio>
</body>
</html>
```

La Figura muestra el resultado en Google Chrome.



En resumen, este código funcionará en los navegadores más conocidos: IExplorer, Edge, Firefox, Google Chrome y Safari 5, pero no funciona en IExplorer8. Si se quitara la línea con el archivo *ogg* entonces Firefox 7 no lo soportaría ya que no lee archivos *mp3*, pero sí en el resto de navegadores antes enumerados. Por otro lado, si se quitara la línea con el archivo *mp3*, en Google Chrome 14 y Firefox 7 funcionaría (porque leen *.ogg*), pero en IExplorer 9 y Safari 5 no.

A modo de conclusión, el uso de la etiqueta `<audio>` hace que insertar audio en la web sea bastante interoperable entre diferentes navegadores (aquellos que interpretan HTML5). Sin embargo, un diseñador, debería ofrecer soluciones que también la soporten otros navegadores más antiguos. Es decir, en el ejemplo anterior ¿qué pasa si un usuario tiene IExplorer8? Si ese usuario no puede ver correctamente los elementos de su sitio web lo más seguro es que lo abandone defraudado.

Una alternativa incluida dentro de HTML4 para insertar objetos no soportados por el navegador es el uso de la etiqueta `<object>`. Combinando `<object>` con `<embed>` se puede, por ejemplo, insertar un reproductor (Flash) para aquellos navegadores que no soporten el código mostrado en el ejemplo anterior ~en nuestro caso solo era IExplorer8).

Esto ofrece una solución más interoperable, aunque tampoco es universal.

El siguiente código muestra un ejemplo de uso de `<object>` con `<audio>`. Si ninguna de las opciones ofrecidas en `<source>` es soportada por el navegador entonces intentará embeber el reproductor *player_mp3_mini.swf* (éste no es el único reproductor Flash, hay muchos más disponibles en Internet).

```
<!DOCTYPE html>
<html>
<head>
</head>
<body>
<audio controls="controls" preload="auto" >
<source src="Aretha.mp3" type="audio/mpeg" />
<source src="Aretha.ogg" type="audio/ogg" />
<object type="application/x-shockwave-flash"
data="player_mp3_mini.swf" width="200" height="20">
<param name="movie" value="player_mp3_mini.swf" />
<param name="bgcolor" value="#085c68" />
<param name="FlashVars" value="mp3=Aretha.mp3" />
<embed href="player_mp3_mini.swf" bgcolor="#085c68"
width="200"
height="20" name="movie" align=""
type="application/x-shockwave-flash"
flashvars="mp3=Aretha.mp3">
</embed>
```



```
</object>
</audio>
</body>
</html>
```

Para que este código funcione es necesario que el archivo *player_mp3_mini.swf* esté disponible en la misma carpeta (local o en web) que la página HTML. Este archivo es el que carga `<object>` cuando ninguna de las otras opciones son soportadas. `<object>` tiene varios parámetros que definen el tamaño del elemento que quiere embeber, en este caso el reproductor Flash. Sin embargo, es con la etiqueta `<embed>` con la que realmente se está incluyendo el archivo *mp3* que se quiere cargar en ese reproductor. De alguna manera, con este código se pretende insertar un audio *mp3* con la etiqueta `<embed>` vista anteriormente, pero añadiendo adrede el reproductor, y no suponer que el navegador lo tiene.

La etiqueta `<object>` es muy conocida en HTML4 por lo que no es objetivo de este manual tratarla en profundidad. Sin embargo, si es interesante resaltar que los valores de los atributos coincidentes entre `<object>` y `<embed>` deben tener los mismos valores para garantizar un funcionamiento adecuado.

A la etiqueta `<audio>` también se le puede aplicar estilos CSS. Por ejemplo, el siguiente código define una estilo para hacer que una etiqueta audio que use la clase *audio-fondo* aparezca con una color de fondo amarillo y una tamaño determinado.

```
<style>
.audio-fondo {
width: 160px;
height: 36px;
background: #FFFF00;
}
</style>
```

A modo de conclusión de los visto, aunque no todos los navegadores actuales soportan HTML5 con la madurez deseada, y no todos los usuarios usan los navegadores debidamente actualizados para garantizar que soportan HTML5, la etiqueta audio ofrece muchas ventajas que aconsejan su uso futuro a la hora de insertar audio en sitios web. HTML5 permite que los navegadores ya no necesiten complementos (*plugins*), del estilo Adobe Flash, Microsoft Silverlight, etc. para reproducir audio y eso le facilita al desarrollador no tener que preocuparse si un navegador reproduce o no un audio determinado.

Los dispositivos móviles actuales integran navegadores que soportan HTML5 por lo que es cuestión de poco tiempo que todos los usuarios tengan navegadores HTML5 y no sea necesario buscar alternativas con `<object>` o *JavaScript* para detectar que audio puede reproducir un complemento determinado de un navegador concreto. Aunque será inevitable que en la etiqueta `<audio>` se den opciones variadas respecto al tipo de archivo (*mp3*, *ogg*).

Por último, como opción de diseño avanzada, se puede usar *JavaScript* y



HTML5 para personalizar el reproductor.

El siguiente código utiliza botones para reproducir, en vez de un *player*. Esta opción da mucho juego en el diseño del reproductor al poder cambiar los botones por imágenes acordes con el estilo del sitio.

```
<!DOCTYPE html>
<html>
<head>
</head>
<body>
<audio id="player" src="Aretha.mp3"></audio>
<div>
<button onclick="document.getElementById('player').play()" ">Play</button>
<button onclick="document.getElementById('player').pause()" ">Pausa</button>
<button onclick="document.getElementById('player').volume+=0.1">+ Volumen</button>
<button onclick="document.getElementById('player').volume-=0.1">- Volumen</button>
</div>
</body>
</html>
```

Ojo es `getElementById` (l mayúscula no ele)

En JavaScript existen librerías, como *audio.js*

<http://kolber.github.com/audiojs/>, que permite reproducir audio en todos los navegadores como con la etiqueta <audio> aunque no soporten HTML5.

También tenemos la posibilidad de implementar reproductores de audio de librerías de jQuery, como por ejemplo: <http://audiocogs.org/codecs/alc/>



ACTIVIDAD

Modifica el esqueleto de tu proyecto para incluir en la parte inferior derecha de la cabecera un reproductor cuyos botones sean imágenes estáticas. Las funciones deben ser *play*, *stop*, subir volumen y bajar volumen y deben funcionar en Chrome 14 o superior (*mp3*) y en Firefox 7 o superior (*ogg*). Cuando el usuario presione una de las imágenes se ejecutará la función deseada. La clave de esta actividad son dos:

- Colocar con CSS las imágenes (controles) en la posición deseada sin modificar las capas existentes.
- Para estar prevenidos de un posible cambio en el tamaño del navegador, la caja con los controles no deberá ser absoluta.
- Asociar a las imágenes la funcionalidad para reproducir.

SOLUCIÓN

Las líneas que se deben incluir al ejemplo de la Actividad son las siguientes:

CSS

```
div.reproductor /*Se define el reproductor
a la derecha*/
margin-left:500px;
margin-bottom:inherit;
clear:left;
position:realtive;
```

HTML

```
<div class="reproductor">
<audio id="player">
<source src="Aretha.mp3" type="audio/mpeg" />
<source src="Aretha.ogg" type="audio/ogg" />
</audio>
</img>
</img>
 </img>
 </img>
</div>
```



CONSIDERACIONES PARA EL USO DE AUDIO EN UN SITIO WEB

En la sección anterior se han mostrado diferentes alternativas para insertar audio con HTML, con la idea de hacer sitios web más funcionales y "atractivos". El adjetivo "atractivo" aparece entre comillas para indicar que su semántica es subjetiva y que no siempre una página web con audio es más atractiva que otra que no lo tenga. El audio tiene varias ventajas que cualquier usuario de Internet conoce: pueden hacer sitios web más accesibles, más sorprendentes, explican cosas que de otra manera no se puede, etc.

En general, el audio en sitios web se puede usar de dos maneras:

Como contenido y por tanto tiene un lugar en un sitio web: Por ejemplo, un sitio web de una cadena de radio necesita insertar sonido para que los usuarios pueda acceder a programas de días anteriores.

Como elemento decorativo, por ejemplo, música de fondo.

El primer uso es inevitable y está asociado con el objetivo del sitio web, por lo que debe implementarse para que pueda ser interpretado por la mayor cantidad de navegadores.

Sin embargo, el segundo uso es meramente ornamental, no imprescindible, con el que se puede conseguir un efecto positivo (por ejemplo, llamar la atención del usuario) o negativo, contrario al que se espera de él: Un ejemplo clásico del audio como elemento decorativo negativo es cuando se usa como fondo. Actualmente su uso está "pasado de moda", los diseños actuales no suelen usar música de fondo, ya que en muchos casos son una molestia para los usuarios (por ejemplo, la música aparece cuando el volumen de los altavoces está en los más alto). En cualquier caso, si aparece sonido de fondo, siempre debe existir la posibilidad de que el usuario lo apague (y/o inicie) cuando quiera, por lo que deberían aparecer los controles del reproductor cuando se muestre.

En conclusión, los audios son archivos con un tamaño considerable. Por lo tanto, su uso se debe restringir para cuando es imprescindible, y siempre optimizando su reproducción (jugando, por ejemplo, con *preload* de la etiqueta <audio>).