

```
<input type="reset" value="Anular">
</form>
</body>
</html>
```

Este formulario, que incluye funcionalidades propias de HTML5, es visualmente muy distinto al anterior, con la introducción del cursor para informar las tallas o del contador para la cantidad.

Los campos **Nombre y apellidos** y **Dirección** aparecen completados por una sugerencia sombreada (atributo `placeholder`).

Algunos campos se han complementado con el atributo `required`, que los vuelve obligatorios. En el momento de realizar el envío, aparece una etiqueta informativa indicando que es preciso completar este campo.

Hemos dotado al código postal de una expresión regular (atributo `pattern`) que exige una codificación exclusivamente numérica. Aparece una etiqueta informativa si el formato no es válido.

El contador, para introducir la cantidad, permite limitar una cantidad máxima, 6 unidades en este caso, mediante el atributo `max`.

He aquí las diferencias respecto a HTML 4.0, que abren perspectivas prometedoras para las aplicaciones HTML del futuro.

## Capítulo 10: Multimedia

- A. Insertar un archivo de audio . . . . . 210
- B. Insertar un archivo de vídeo . . . . . 218

## A. Insertar un archivo de audio

### 1. Etiqueta <audio>

Hasta el presente, no existía un estándar para agregar sonido en una aplicación Web. Esta operación se realizaba mediante un plug-in, por ejemplo Flash, aunque no todos los navegadores disponían de los mismos plug-ins.

Html5 proporciona a día de hoy una etiqueta nueva para reproducir archivos de audio, independientemente de los plug-ins instalados en el puesto del usuario.

Insertar un archivo de audio puede realizarse de forma muy sencilla mediante la etiqueta:

```
<audio src="archivo_sonido"></audio>
```

Dado que la implementación de la etiqueta <audio> es todavía muy limitada en los navegadores existentes, no es inútil agregar un mensaje para aquellos navegadores que no puedan procesar esta etiqueta.

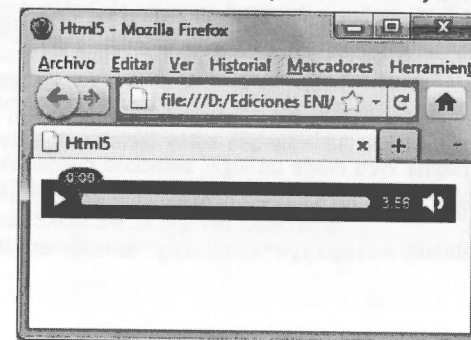
```
<audio src="archivo_sonido">
Su navegador no soporta la etiqueta audio.
</audio>
```

Veamos un primer ejemplo:

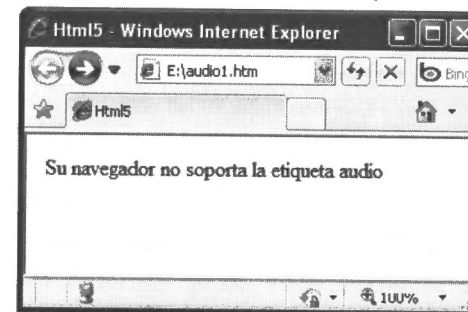
```
<!DOCTYPE html>
<html lang="es">
<head>
<title>Html5</title>
<meta charset="iso-8859-1">
</head>
<body>
<audio src="piano_ogg.ogg" controls>
Su navegador no soporta la etiqueta audio.
</audio>
</body>
</html>
```

La ruta del archivo de audio supone, en este caso, que el archivo se encuentra en la misma carpeta del servidor que el archivo Html que contiene la etiqueta audio.

El resultado en Firefox 3.6, que reconoce la etiqueta <audio> y el formato ogg:



El resultado en Internet Explorer 8, que no reconoce la etiqueta <audio>:



Los atributos de la etiqueta <audio> son:

#### src

Obligatorio. Define la ruta del archivo de sonido.

#### controls

Muestra los controles del reproductor de audio (véase la captura de pantalla). Incluye las funcionalidades de reproducción, pausa, avance y volumen.

```
<audio src="sonido.ogg" controls>
```

En ausencia del atributo controls, el navegador no mostrará los controles del reproductor de audio.

La sintaxis Xhtml para los atributos controls="controls" también es válida.

**autoplay**

Define la reproducción automática del archivo de sonido desde el momento en que esté disponible, tras la carga de la página.

```
<audio src="sonido.ogg" autoplay>
```

Es preciso recordar que esta lectura automática del archivo de sonido no siempre es bienvenida por parte de los usuarios. Aquellos que estén escuchando su propia música o que estén consultando su página Web desde un lugar silencioso pueden verse molestados. Por otro lado, esta reproducción automática será especialmente molesta para los usuarios de programas de síntesis vocal.

La sintaxis Xhtml del atributo `autoplay="autoplay"` también es válida.

**loop**

Especifica que el archivo de sonido se reproduzca en bucle. De este modo, el sonido comienza de nuevo una vez termina.

```
<audio src="sonido.ogg" loop>
```

La sintaxis Xhtml del atributo `loop="loop"` también es válida.

**preload**

Indica al navegador que debe descargar el archivo de audio durante la carga de la página de modo que esté disponible para una reproducción inmediata una vez la solicite el usuario.

```
<audio src="sonido.ogg" preload>
```

Este atributo puede tomar los valores:

- `preload="none"`. Sin carga previa.
- `preload="metadata"`. Carga previa de los metadatos (metadata) asociados al archivo.
- `preload="auto"`. Carga previa automática.

Este atributo `preload` se ignorará si el atributo `autoplay` está activado.

🔗 Destacamos lo concisa que resulta la inserción de un archivo sonoro gracias a esta etiqueta `<audio>` y sus atributos de forma reducida.

**2. Formatos de archivo de audio**

La situación ideal hubiera sido que se hubiera adoptado un único formato de archivo de audio, y que se hubiera tratado de un formato libre.

**Formato ogg**

Esta ha sido la opción de Firefox, primer navegador en implementar la etiqueta `<audio>`. El formato ogg se ha mantenido.

Ogg es el nombre del principal proyecto de la fundación Xiph.org que tiene como objetivo proporcionar a la comunidad de usuarios formatos y codecs multimedia abiertos, libres y desprovistos de cualquier tipo de patente. La extensión `.ogg` es una de las extensiones posibles para los archivos de formato ogg. Por abuso del lenguaje, se denomina de forma corriente "archivo Ogg" de audio a un archivo de formato ogg que contenga datos de audio comprimidos con Vorbis, uno de los codecs del proyecto ogg (fuente Wikipedia).

Los archivos ogg suponen una alternativa libre al formato mp3, mucho más conocido y expandido.

El formato ogg está reconocido en Firefox 3.6+, Chrome 5+ y Opera 10.6+. No lo reconocen los navegadores Safari 5+ ni Internet Explorer 9.

**Formato mp3**

Es difícil olvidarse del emblemático formato mp3 para los archivos de sonido.

MPEG-1/2 Audio Layer 3, más conocido por su abreviatura MP3, es la especificación sonora del estándar MPEG-1/MPEG-2, de *Moving Picture Experts Group* (MPEG). Se trata de un algoritmo de compresión de audio capaz de reducir drásticamente la cantidad de datos necesaria para restituir el sonido, pero que, para el oyente, se asemeja a una reproducción del sonido original sin compresión, es decir con pérdida de calidad significativa pero aceptable para el oído humano (fuente Wikipedia).

MP3 se percibe a menudo por parte del usuario como una tecnología gratuita. Esto permite que pueda codificar y decodificar su música de forma totalmente legal siempre y cuando el archivo original le pertenezca o se trate de una copia de uso privado. No obstante, esta tecnología propietaria forma parte de una patente y una licencia comercial.

El formato mp3 lo reconocen Chrome 5+, Safari 5+ e Internet Explorer 9. No se reconoce en Firefox 3.6 y 4 ni en Opera 10.6.

**Formato acc**

Existen otros formatos con mejor rendimiento en términos de compresión que el formato ogg o el mp3. Se trata del formato acc.

El formato ACC (*Advanced Audio Coding*) es un algoritmo de compresión de audio con pérdida de datos pero que tiene mejor rendimiento en términos de compresión que el formato más antiguo MPEG-1/2 Audio Layer 3 (más conocido como mp3). Por este motivo ha sido escogido por algunas firmas como Apple o RealNetworks (fuente Wikipedia).

El formato acc es el formato de los archivos de audio soportados por Apple en su reproductor portátil iPod y en su software iTunes.

Se trata también de un formato propietario.

El formato acc lo reconocen Chrome 5+, Safari 5+ e Internet Explorer 9. No lo reconocen los navegadores Firefox 3.6 y 4 ni Opera 10.6.



## Capítulo 10

### Formato wav

Citamos el clásico formato wav que, por su ausencia de compresión, no parece uno de los más adaptados para competir en la red. En efecto, el tamaño de los archivos wav es muy (o demasiado) grande.

El formato wav lo reconocen Firefox 3.6+, Safari 5+ y Opera 10.6+. No se reconoce en Chrome 5+ ni en Internet Explorer 9.

### Tabla comparativa

	Chrome 5+	Firefox 3.6+	Opera 10.6+	Safari 5+	Explorer 9
ogg	sí	sí	sí	no	no
mp3	sí	no	no	sí	sí
acc	sí	no	no	sí	sí
wav	no	sí	sí	sí	no

Para concluir:

- La etiqueta `<audio>` ha sido implementada en los navegadores más recientes.
- Google Chrome 5+ es, con diferencia, el navegador más polivalente en lo referente a formatos de audio.

Html5 está todavía en un estado incipiente. Es posible que aparezca algún nuevo formato de audio. Del mismo modo, cualquier navegador puede incorporar cualquier tipo de archivo que no reconozca a día de hoy. El autor le aconseja consultar regularmente el apartado Html5 Audio Codecs del sitio Web [www.findmebyip.com/litmus/](http://www.findmebyip.com/litmus/) para consultar los futuros desarrollos y avances a este respecto.

### 3. Etiqueta `<source>`

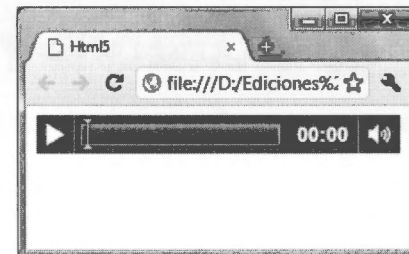
La etiqueta `<source>` nos va a permitir resolver la problemática de los distintos formatos de archivo.

La etiqueta `<source>` se utiliza para especificar varios tipos de archivos de audio. Cada navegador escogerá el formato que mejor le convenga o el que pueda reproducir.

#### Ejemplo

```
<!DOCTYPE html>
<html lang="es">
<head>
<title>Html5</title>
<meta charset="iso-8859-1">
</head>
<body>
<audio controls>
```

```
<source src="piano_ogg.ogg">
<source src="piano_mp3.mp3">
<source src="piano_acc.acc">
Su navegador no soporta la etiqueta audio.
</audio>
</body>
</html>
```



Los archivos de sonido están disponibles en el área de descargas.

Es importante comprender el funcionamiento de las distintas etiquetas `<source>`.

Supongamos que leemos esta página en el navegador Safari que, recordemos, no soporta el formato ogg. Safari ignorará la primera etiqueta `<source>` que redirige a un archivo de tipo ogg. Pasa a la segunda etiqueta `<source>` que propone un archivo mp3. Como el formato mp3 sí está soportado en Safari, será este recurso el que se usará para reproducir el archivo de sonido.

Por el contrario, Firefox, que soporta los archivos ogg, utilizará el primer recurso. Las demás etiquetas `<source>` serán ignoradas.

Las etiquetas `<source>` se leen en el orden de aparición en el código fuente. El navegador usará el primer recurso que soporte.

Las múltiples etiquetas `<source>` aseguran una perfecta reproducción del archivo de sonido en sus distintos formatos por parte de los navegadores. No obstante, el diseñador del sitio o de la aplicación Web tiene que codificar el mismo archivo de sonido en varios formatos para asegurar la compatibilidad entre los distintos formatos.

Los atributos de la etiqueta `<audio>` son:

#### src

Obligatorio. Define la ruta del archivo de sonido.

## type

Define el tipo MIME del contenido. Puede valer:

- type="audio/ogg".
- type="audio/mpeg".
- type="audio/acc".

También puede especificarse el codec utilizado. El atributo type queda: type="audio/ogg; codecs=vorbis".

Especificando el atributo type, acelerará el proceso de carga de la etiqueta <source> más adecuada para el navegador.

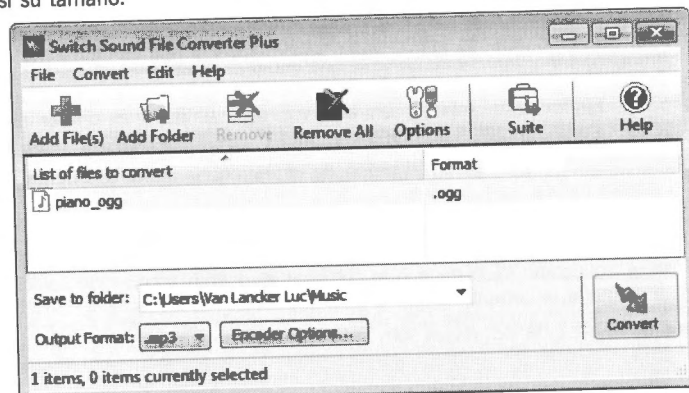
## 4. Programas de conversión de audio

Existen numerosos programas de conversión de audio, gratuitos o en versión de evaluación, en la red. Esbocemos a continuación un par de ellos.

### Switch Audio Converter

Es un programa gratuito. Se puede descargar de la dirección [www.nch.com.au/switch/index.html](http://www.nch.com.au/switch/index.html). Está disponible para Windows y para Mac.

Switch permite convertir a los formatos ogg, mp3 y acc, y también a otros tipos de formato de audio (wma, flac, raw...). Además, es posible ajustar la calidad del archivo de salida y reducir así su tamaño.



## Audacity

Con Audacity, dispondrá de un editor de audio relativamente completo y de fácil manejo. Está disponible para Windows, Mac OS X, Linux y otros sistemas operativos.



Este programa le permitirá editar los archivos de audio para copiar/pegar, mezclar varias pistas, agregar efectos de fundido al inicio o al final, agregar efectos sonoros, corregir el ruido o el volumen, etc.

En lo que respecta a la conversión de formatos de audio, Audacity proporciona numerosos formatos de exportación, entre ellos los formatos ogg, mp3 y acc de la etiqueta <audio>.

## 5. Compatibilidad con navegadores antiguos

Es posible asegurar la compatibilidad hacia atrás con los navegadores más antiguos que no soportan la etiqueta <audio>, como por ejemplo Internet Explorer 8.

Hemos visto que es posible agregar un contenido alternativo en la etiqueta <audio>. Mejor que una simple frase indicando que el navegador no soporta esta etiqueta, es posible incluir las etiquetas <object> y <param> que se usan en la generación anterior de navegadores para leer archivos de tipo multimedia.

El código pierde simplicidad pero se asegura la compatibilidad.

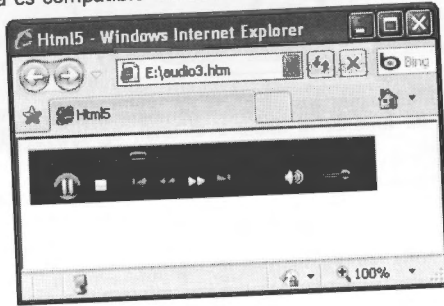
```
<!DOCTYPE html>
<html lang="es">
<head>
<title>Html5</title>
<meta charset="iso-8859-1">
</head>
<body>
<audio controls>
<source src="piano_ogg.ogg">
<source src="piano_mp3.mp3">
<source src="piano_acc.acc">
<object type="audio/mpeg" data="piano_mp3.mp3"
title="Archivo sonoro" width="280" height="45"
classid="CLSID:22d6f312-b0f6-11d0-94ab-0080c74c7e95">
<param name="src" value="piano_mp3.mp3" />
<param name="autoplay" value="false" />
<param name="loop" value="false" />
<param name="controller" value="true" />
</object>
```



## Capítulo 10

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

El archivo de sonido ya es compatible con Internet Explorer 8.



## B. Insertar un archivo de vídeo

### 1. Etiqueta <video>

Tras varios años, el vídeo ha invadido la Web. Basta, por ejemplo, con pensar en el éxito de YouTube. No obstante en Html 4.0 no existía un estándar para agregar vídeo en una aplicación Web. Esta operación se realizaba con la llamada a algún plugin: QuickTime, RealPlayer o Flash. Esto implicaba contar con el plugin adecuado en cada momento.

Html5 proporciona a día de hoy una etiqueta nueva para reproducir los archivos de vídeo, independientemente de los plugins instalados en el puesto del usuario.

Insertar un archivo de vídeo se realiza de forma muy sencilla mediante la etiqueta:

```
<video src="archivo_video"></video>
```

Dado que la implementación de esta etiqueta <video> es muy limitada a día de hoy en los navegadores existentes, no es descabellado prever un mensaje para los usuarios de aquellos navegadores que todavía no tienen en cuenta esta etiqueta.

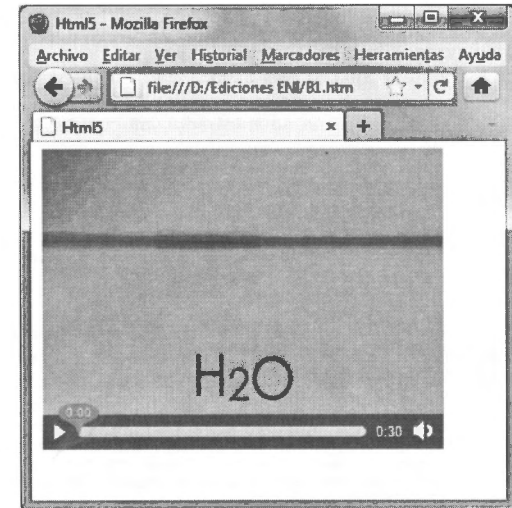
```
<video src="archivo_video">
Su navegador no soporta la etiqueta video.
</video>
```

Veamos un primer ejemplo:

```
<!DOCTYPE html>
<html lang="es">
<head>
<title>Html5</title>
<meta charset="iso-8859-1">
</head>
<body>
<video src="video_ogv.ogv" controls>
Su navegador no soporta la etiqueta video.
</video>
</body>
</html>
```

La ruta del archivo de vídeo supone, en este caso, que el archivo se encuentra en la misma carpeta del servidor que el archivo Html que contiene la etiqueta de vídeo.

El resultado en Firefox 3.6, que sí reconoce la etiqueta <video> y el formato ogv.



Los atributos de la etiqueta <video> son:

**src**

Obligatorio. Define la ruta del archivo de vídeo.

## width

Determina la anchura del vídeo.

## height

Determina la altura del vídeo.

Igual que con la etiqueta `<img>`, usted puede definir de forma explícita las dimensiones del vídeo. En otro caso, el elemento se muestra por defecto con la altura y la anchura propias del vídeo. Si especifica una dimensión, pero no la otra, el navegador ajustará automáticamente el tamaño de la dimensión que no se haya especificado para preservar la proporción de aspecto del vídeo.

## poster

El atributo `poster` permite especificar una imagen que el navegador usará mientras se está descargando el vídeo o hasta que el usuario inicie su reproducción. Si no se especifica este atributo, se inserta la primera imagen del vídeo en su lugar.

```
<video src="video_ogv.ogv" poster="imagen.png">
```

## controls

Muestra los controles del reproductor del vídeo (véase la captura de pantalla). Se incluyen las funciones de reproducción, pausa, avance y volumen.

```
<video src="video_ogv.ogv" controls>
```

En ausencia del atributo `controls`, no se mostrarán los controles del reproductor en el navegador.

La sintaxis Xhtml del atributo `controls="controls"` también es válida.

## autoplay

Define la lectura automática del archivo de vídeo una vez esté disponible, tras la carga de la página.

```
<video src="video_ogv.ogv" autoplay>
```

Recordemos que esta reproducción automática no siempre es apreciada por los usuarios, especialmente si la página contiene algo de texto.

La sintaxis Xhtml del atributo `autoplay="autoplay"` también es válida.

## loop

Especifica que el archivo de vídeo se reproducirá en bucle. De este modo, el vídeo se reproduce nuevamente una vez ha llegado a su fin.

```
<video src="video_ogv.ogv" loop>
```

La sintaxis Xhtml del atributo `loop="loop"` también es válida.

## preload

Indica al navegador que debe descargar el archivo de vídeo durante la carga de la página de modo que esté disponible para una reproducción inmediata una vez lo solicite el usuario.

```
<video src="video_ogv.ogv" preload>
```

Este atributo puede tomar los valores:

- `preload="none"`. No existe carga previa.
- `preload="metadata"`. Carga previa de los metadatos asociados al archivo.
- `preload="auto"`. Carga previa automática.

Este atributo `preload` se ignora si está presente el atributo `autoplay`.

## 2. Formatos de archivo de vídeo

Igual que para los archivos de audio, existen varios formatos de archivos de vídeo.

### Formato ogv

El formato ogv para los archivos de vídeo es el equivalente a la extensión ogg para los archivos de audio.

Los archivos de vídeo ogv utilizan los codecs Theora para el vídeo y Vorbis para el audio.

Recordemos que los formatos ogg y ogv son formatos libres.

Este formato está soportado en Firefox 3.6 desde la implementación de la etiqueta de vídeo (junio de 2009). Opera 10.5+ y Chrome 3+ también lo soportan.

### Formato H.264

El navegador Safari para Mac y PC utiliza QuickTime, que reconoce numerosos formatos de vídeo, aunque no reconoce los formatos ogg y los codecs Theora y Vorbis. Trabaja con el codec H.264.

H.264 es un codec de compresión de vídeo digital y de vídeo en alta resolución según la norma MPEG-4, desarrollada por el VCRG (*Video Coding Experts Group*) en asociación con el MPEG (*Moving Picture Experts Group*), también conocido con el nombre de AVC (*Advanced Video Coding*). El formato H.264 también se conoce con la nomenclatura MPEG-4 AVC o incluso MPEG-4 Part 10.

Este formato, promocionado por Apple, tuvo gran éxito, especialmente después de que YouTube lo adoptara.

No obstante el formato H.264 no es un formato libre ni gratuito. Se trata de un formato de vídeo propietario que forma parte de acuerdos comerciales y royalties.

Ante las protestas por parte de los defensores de una Web libre, el consorcio MPEG-LA ha liberado el formato H.264 para aquellos sitios Web y plataformas que ofrezcan gratuitamente sus contenidos de vídeo, hasta 2011. Este acuerdo se renovó en su momento hasta 2016. Finalmente, y tras la presión ejercida por un nuevo formato libre, WebM, promocionado



por Google, el consorcio ha decidido ofrecer el formato libre de royalties siempre y cuando el acceso al vídeo sea gratuito para el usuario final.

Es un buen ejemplo de la guerra de formatos de vídeo que se está librando en el presente...

El formato H.264 está soportado por Chrome 5+, Safari 5+ e Internet Explorer 9. No está soportado en Firefox ni Opera 10.6.

### Formato WebM

Google adquirió el codec de vídeo VP8 tras la compra de la empresa On2 Technologies.

Para contrarrestar el ascenso fulgurante del formato H.264, Google publicó en mayo de 2010 el código fuente del codec VP8, libre de licencia. Está asociado a Mozilla y Opera y proporciona un formato de vídeo abierto, WebM, que utiliza VP8 para el flujo de vídeo y Vorbis para el flujo de audio.

El formato WebM está soportado por Firefox 4+, Chrome 6+ y Opera 10.6+. No está soportado en Safari 5+ ni en Internet Explorer 9.

### Tabla comparativa

	Chrome		Firefox		Opera 10.6+	Safari 5+	Explorer 9
	5	6	3.6	4			
ogv	sí	sí	sí	sí	sí	no	no
H.264	sí	sí	no	no	no	sí	sí
WebM	no	sí	no	sí	sí	no	no

Para concluir:

- La etiqueta `<video>` está implementada en los navegadores más recientes.
- Google Chrome 6+ es, con diferencia, el navegador más polivalente en lo que a formatos de vídeo se refiere.

Html5 está todavía en un estado incipiente. Es posible que aparezca algún nuevo formato de vídeo. Del mismo modo, cualquier navegador puede incorporar cualquier tipo de archivo que no reconozca a día de hoy. El autor le aconseja consultar regularmente el apartado Html5 Video Codecs del sitio Web [www.findmebyip.com/litmus/](http://www.findmebyip.com/litmus/) para consultar los futuros desarrollos y avances a este respecto.

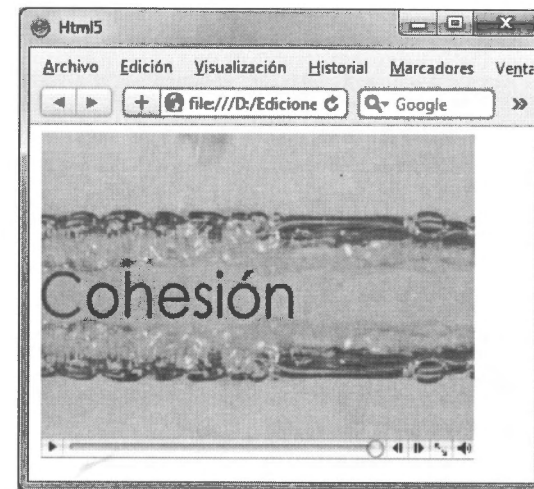
### 3. Etiqueta `<source>`

La etiqueta `<source>` nos va a permitir resolver la problemática de los distintos formatos de archivo.

La etiqueta `<source>` se utiliza para especificar varios tipos de archivos de vídeo. Cada navegador escogerá el formato que mejor le convenga o el que pueda reproducir.

### Ejemplo

```
<!DOCTYPE html>
<html lang="es">
<head>
<title>Html5</title>
<meta charset="iso-8859-1">
</head>
<body>
<video controls>
<source src="video_ogv.ogv">
<source src="video_mp4.mp4">
<source src="video_webm.webm">
Su navegador no soporta la etiqueta video.
</video>
</body>
</html>
```



Los archivos están disponibles en el área de descargas.

Es importante comprender el funcionamiento de las distintas etiquetas `<source>`.



Supongamos que leemos esta página en el navegador Safari que, recordemos, no soporta el formato ogv. Safari ignorará la primera etiqueta `<source>` que remite a un archivo de tipo ogv. Pasa a la segunda etiqueta `<source>` que propone un archivo mp4. Como el formato mp4 sí está soportado en Safari, será este recurso el que se usará para reproducir el archivo de vídeo.

Por el contrario, Firefox, que soporta los archivos ogv, utilizará el primer recurso. Las demás etiquetas `<source>` serán ignoradas.

Las etiquetas `<source>` se leen en el orden de aparición en el código fuente. El navegador usará el primer recurso que soporte.

Las múltiples etiquetas `<source>` aseguran una perfecta reproducción del archivo de vídeo en sus distintos formatos por parte de los navegadores. No obstante, el diseñador del sitio o de la aplicación Web tiene que codificar el mismo archivo de vídeo en varios formatos para asegurar la compatibilidad entre los distintos formatos.

Los atributos de la etiqueta `<source>` son:

**src**

Obligatorio. Define la ruta del archivo de sonido.

**type**

Define el tipo MIME del contenido. Puede valer:

- `type="video/ogg".`
- `type="video/mp4".`
- `type="video/webm".`

También puede especificarse el codec utilizado. El atributo `type` queda:

- `type='video/ogg; codecs="theora, vorbis"'.`
- `type='video/webm; codecs="vp8, vorbis"'.`
- `type='video/mp4; codecs="avc1.42E01E, mp4a.40.2"'.`

Especificando el atributo `type` acelerará el proceso de carga de la etiqueta `<source>` más adecuada para el navegador.

#### 4. Programas de conversión de vídeo

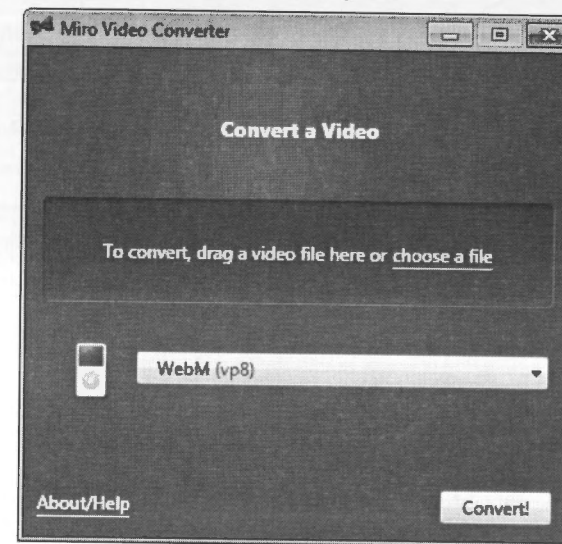
Existen numerosos programas de conversión de vídeo, gratuitos o en versión de evaluación, en la red. Esbocemos a continuación algunos de ellos.

##### Miro Video Converter

Miro Video Converter es un software libre para Windows y Mac. Puede descargarse de la dirección: [www.mirovideoconverter.com/](http://www.mirovideoconverter.com/)

Este programa convierte cualquier vídeo para habilitar su lectura en iPhone, Android, PSP, etc. Simplemente deslice y suelte el archivo que quiere transformar sobre la interfaz del programa y a continuación seleccione el formato de salida deseado (MP4, Ogg Theora,

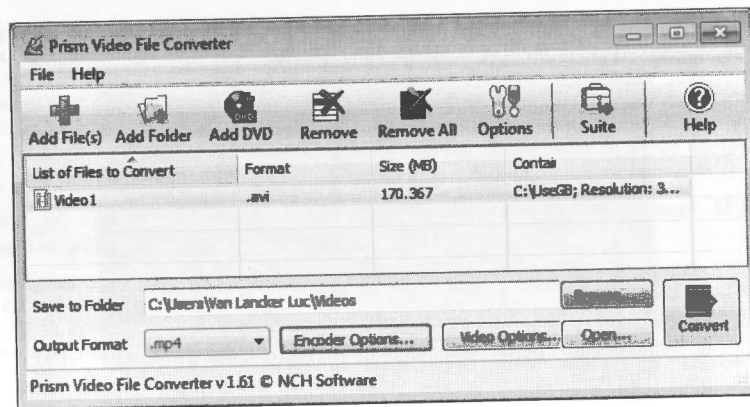
WebM, etc.) o bien el dispositivo de destino: Android, PSP, iPhone, iPad, etc. Por último, haga clic en **Convert!** para realizar la conversión, ¡y listo!



Es difícil hacer más sencillo un programa de conversión de vídeo. El único inconveniente es que no es posible configurar con precisión los parámetros de conversión.

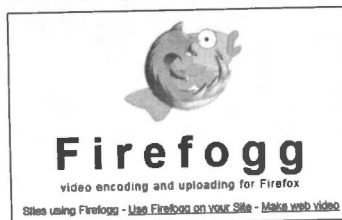
##### Prism Video Converter

Prism Video Converter es un conversor de vídeo gratuito que soporta la mayoría de formatos de vídeo. Le permitirá codificar sus propios archivos AVI, DivX, MPG, VOB, WMV, 3GP... según los formatos compatibles con PSP, iPod, teléfonos móviles, lectores portátiles, etc. Su uso es muy sencillo, y además permite configurar parámetros de conversión tales como la tasa de compresión y la resolución de los archivos resultantes, o incluso el acceso directo mediante el menú contextual.

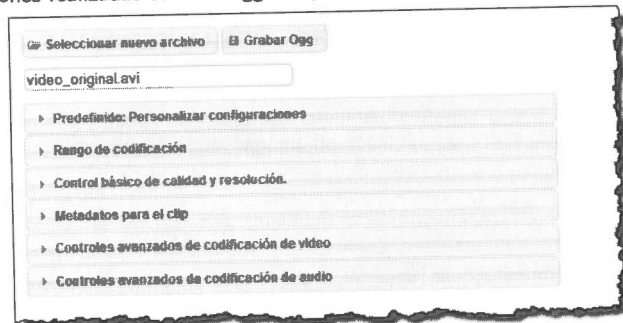


### Firefogg

Firefogg es una extensión de Firefox que convierte automáticamente los vídeos a un formato abierto (Ogv y WebM). Se puede descargar de la dirección <http://firefogg.org/>.



Las conversiones realizadas con Firefogg son perfectamente configurables.

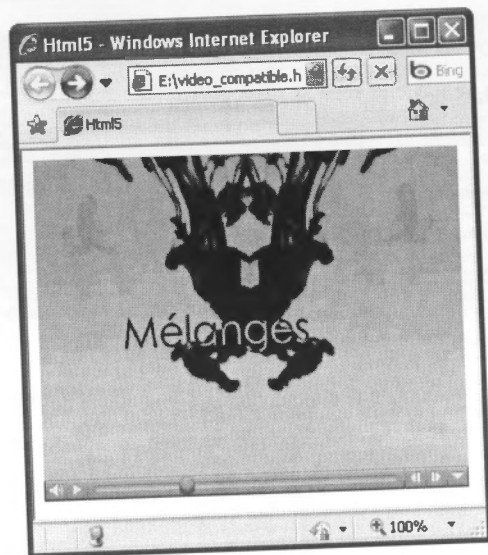


Como Firefogg es una extensión de Firefox, el usuario puede estar seguro de que tiene a su disposición los progresos más recientes en lo que respecta a formatos de vídeo soportados.

### 5. Compatibilidad con navegadores antiguos

```
<!DOCTYPE html>
<html lang="es">
<head>
<title>Html5</title>
<meta charset="iso-8859-1">
</head>
<body>
<video controls>
<source src="video_ogv.ogv">
<source src="video_mp4.mp4">
<source src="video_webm.webm">
<object classid="clsid:02BF25D5-8C17-4B23-BC80-D3488ABDDC6B"
codebase="http://www.apple.com/qtactivex/qtplugin.cab" width="550"
height="320"standby="Cargando...">
<param name="src" value="video_mp4.mp4">
<param name="type" value="application/x-mplayer2">
<param name="autoplay" value="true">
<param name="quality" value="high">
<p>No se puede mostrar el vídeo. Descargue el siguiente <a
href="http://www.apple.com/quicktime/download/">plug-in</a> e
instálelo o descargue el <a href="video_mp4.mp4">archivo
de vídeo</a> directamente.</p>
</object>
</video>
</body>
</html>
```





## Capítulo 11: Diseño 2D

A. Introducción . . . . .	230
B. Etiqueta <canvas> . . . . .	230
C. Llamada al API de diseño . . . . .	232
D. Diseñar un rectángulo . . . . .	234
E. Agregar color . . . . .	237
F. Trazar una línea recta . . . . .	238
G. Diseño de formas geométricas. . . . .	244
H. Importar imágenes . . . . .	248
I. Agregar texto . . . . .	251
J. Agregar sombreado . . . . .	253
K. Transformaciones . . . . .	254