
Eventos comunes y no comunes aceptados por el objeto Image. Trucos para utilizar otros eventos no aceptados.

El objeto *Image* admite solamente **3 eventos comunes** a los 3 navegadores: Internet Explorer, Netscape Navigator 4x y Netscape Navigator 6x. Estos son:

- **onAbort**, que se activa cuando se aborta la carga de una imagen en pantalla, por ejemplo porque el usuario ha pulsado la opción "detener" (stop) en la barra superior de iconos del navegador. Ejemplo de sintaxis:

```

```

- **onError**, que se dispara cuando por algún motivo externo no se ha podido realizar la carga de la imagen en pantalla, por ejemplo porque la ruta de la misma esté mal especificada. Ejemplo de sintaxis:

```

```

- **onLoad**, que se activa cuando se ha acabado de cargar la imagen y se presenta en pantalla. Ejemplo de sintaxis:

```

```

Pero además de estos eventos comunes, las imágenes soportan otra serie de eventos que, si bien no son soportados por Netscape 4x, sí lo son por Internet Explorer y por Netscape 6x. Por este motivo, sólo es conveniente su uso cuando van a ejecutar una acción que no sea fundamental ni para la presentación ni para la ejecución de código de la página. Esperemos a que pronto se estandarice el uso de Netscape 6x y se deje a un lado las versiones 4x, y entonces podremos aplicar estos eventos con compatibilidad total. Estos **eventos adicionales** son:

- **onClick**, que se activa cuando se hace click con el puntero del ratón sobre la imagen. Ejemplo de sintaxis:

```

```



ejemplo de evento onClick

- **onmouseover**, que se activa cuando se el puntero del ratón pasa sobre la imagen. Ejemplo de sintaxis:

```

```



ejemplo de evento onmouseover

- **onmouseout**, que se activa cuando el puntero del ratón, tras pasar sobre la imagen, se va de ella. Ejemplo de sintaxis:
-

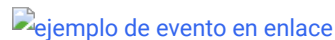
Compatibilizando eventos

Hemos visto que las imágenes sólo admiten como eventos estándares *onLoad*, *onError* y *onAbort*, pero la mayoría de las veces nosotros necesitamos elementos que soporten todo tipo de eventos, como *onClick*, *onmouseover*, etc., y en todo tipo de navegadores.

Podemos, con un poco de imaginación, utilizar otros elementos de JavaScript para conseguir esto. Así, sabemos que el objeto *Link* sí admite todo tipo de eventos, por lo que podemos usarlo como herramienta auxiliar para dotar a las imágenes de las funcionalidades que necesitamos.

La idea es muy simple: basta situar la imagen que deseamos que soporte los eventos dentro de un enlace simple de HTML, es decir, situando la etiqueta `<IMAGE>` dentro de la pareja de etiquetas `< A HREF.....>...`, y entonces establecer el evento necesario NO en la imagen, sino en el enlace. Como ejemplo práctico vamos a introducir una imagen que soporte de forma general el evento *onClick*:

```
<a href="#" onClick="alert('que evento más bonito');return false;"></a>
```



Nota lo siguiente:

1) si sitúas el cursor sobre la imagen verás que éste se transforma en la mano típica de los enlaces. Lógico, ya que hemos transformado la imagen en uno de ellos. Si deseas eliminar este efecto, basta con asignar al enlace un estilo para el cursor, dejando este como *default*.

```
<a href="#" ..... style="cursor:default;"><img src=.....></a>
```

pero ten en cuenta que esto sólo será válido para Internet Explorer y para Netscape 6x. Las versiones 4x de este último navegador no soportan el estilo en cursores, por lo que en ellos siempre aparecerá la mano.

2) Ojo con el atributo BORDER. Debes ponerlo siempre, igualándolo a cero, ya que si no se verá un recuadro azul alrededor de la imagen, el típico de todos los enlaces.

3) Hemos escrito *a href="#"* porque no vamos a llamar a ninguna página, y *return false* para anular el efecto del enlace en sí, ya que sólo queremos que se ejecute el evento, no la llamada del enlace.

Por lo demás este truco es totalmente compatible, y podemos desde el evento del enlace ejecutar código JavaScript o llamar a una función previamente definida.

Consultor, diseñador y desarrollador web en
ParaRedeBJS. Especialista en usabil...

Manual

Tratamiento de imágenes en Javascript

Propiedades del objeto Image

El constructor de objetos Image

¿Te ha resultado útil este artículo?

(0) 

 (0)

¿Alguna duda?

Pregunta y ayuda en la comunidad con tus respuestas en la [sección de FAQ](#)

HACER UNA PREGUNTA

Aprender

Manuales

Temas

Gestionar descargas

Canales



Comunidad

Preguntas y respuestas

Hacer una pregunta

Colecciones

Experiencia

[Home](#) [Datos legales](#) [Privacidad](#) [Política de cookies](#) [Contacto](#)