

5 métodos para imágenes Responsives o adaptables

Ethan Marcott, el que acuñó el término Responsive Web Design, afirma en su artículo que el Fluid imagen es uno de los ingredientes de Responsive Web Design. El problema con la imagen fluida es que apesar que la imagen parece ser sensible y podría encajar muy bien en varios tamaños vista, el mismo tamaño de la imagen se descarga por los usuarios, independientemente del medio de la imagen en la cual se esta viendo.

Este es uno de los temas más discutidos entre los desarrolladores web. Y no hay mucho que hacer antes de RWD puede ser una solución adecuada para la construcción de un sitio web optimizado para móviles. Por el momento hay varias técnicas y herramientas que tratan de resolver este obstáculo. Echemos un vistazo a algunas de ellas ahora.

1. Adaptive Image

[Adaptive Image](#) es un pequeño script PHP que detecta el tamaño de la pantalla del usuario y ofrece la imagen correcta para ese tamaño de pantalla. No se requiere el cambio de su código en la etiqueta `<img>` actual, pero la aplicación es bastante intimidante para los usuarios menos conocedores de la tecnología.



2. Imagen Responsive con el uso de cookies

Keith Clark tiene su enfoque en el servicio de responsive imagen con Cookies. Técnicamente detecta el tamaño de la

pantalla usando JavaScript y tomar el tamaño de imagen adecuado usando PHP antes de que se cargue la imagen en la página.

Sin embargo, esta técnica tiene un montón de problemas y requieren un poco más de experimentación antes de que pueda convertirse en una solución práctica. Usted puede leer su explicación, así como obtener el código del artículo en el blog de Clark, [Imagen Responsive con el uso de cookies](#).

3. Sencha.io Src

[Sencha](#) es un marco de desarrollo móvil que nos permite construir una aplicación nativa con HTML, CSS y JavaScript. Si usted construye su aplicación con Sencha, puede utilizar su API, Sencha.io Src, para cambiar el tamaño de las imágenes de una manera inteligente basada en el tamaño de la pantalla del dispositivo del usuario. El API tiene un conjunto de funciones que permiten flexibilidad en la salida. Para más detalles técnicos, diríjase a esta [referencia](#).



4. HTML5 Picture

Este parece ser el futuro estándar de imagen sensible. Se propone un [nuevo elemento llamado <picture>](#) que nos permitirá establecer diferentes fuentes de imágenes, y también para referirse al punto de corte para los que se debe servir la fuente de la imagen, así:

```
[code lang=»html»]
<picture id="images">
<source media="(min-width: 45em)" srcset="large-1.jpg 1x,
large-2.jpg 2x">
<source media="(min-width: 18em)" srcset="med-1.jpg 1x,
med-2.jpg 2x">
</picture>
```

[/code]

Este elemento, sin embargo, está todavía en una fase de desarrollo. Todavía no es aplicable. Hasta entonces se puede utilizar un [Polyfill](#) llamada relleno de imagen.

Polyfill – relleno de imagen

[Picture Fill](#) es una pequeña biblioteca de JavaScript que se desarrolla por Scott Jehl. Imita el funcionamiento **<picture>** usando el elemento ****.

```
[code lang=»html»]
<span data-picture data-alt="A giant stone face at The Bayon
temple in Angkor Thom, Cambodia">
<span data-src="small.jpg"></span>
<span data-src="medium.jpg" data-media="(min-width:
400px)"></span>
</span>
[/code]
```

Plugins

Si usted está usando WordPress como su plataforma de publicación, estás de suerte, ya que hay varios plugins que sirven a la imagen de una manera similar al elemento **<picture>**.

- [PB Responsive Image](#)
- [WP Responsive Image](#)
- [Simple Responsive Image](#)
- [Picture Fill WP](#)

Si utiliza Drupal, hay un proyecto llamado [Picture](#) que funciona de la misma manera.

5. Punto focal

[Punto Focal](#) es un framework que permite a los desarrolladores de las imágenes «cultivar» y controlar su punto focal para el

diseño responsive. Esta técnica es puramente hecho con CSS; el autor simplemente añade las clases en el elemento que contiene la imagen. Cabe señalar sin embargo que el truco de esta técnica es el uso de desbordamiento: permite ocultar una parte de la imagen. Así que la imagen no es (técnicamente) cultivada(crop), lo que significa que terminamos de descargar el mismo tamaño de la imagen.

Como se ha dicho, todavía no existe una aplicación que establece un estándar para responder a este problema por completo. Estas técnicas e instrumentos de pesar antes mencionados como soluciones, tienen sus propios inconvenientes y ninguno puede salir como la solución final para todos los casos.

También se pregunta si la nueva propiedad propuesta, **<picture>** será una solución recomendada en el futuro. Así que, hasta entonces, parece que estamos atascados con la imagen fluida o responsive.