

TE02UT04

Estudio comparativo de técnicas de carga de imágenes

Introducción	1
1.- Carga de imágenes sin aplicar técnicas	1
2.- Carga de imágenes usando color promedio	2
3.- Carga de imágenes usando baja resolución	2
4.- Uso de picture	3
5.- Uso de Preload - Prefetch	3
6.- Uso de image-set	4
7.- Conclusiones	4
ANEXO I. Tabla comparativa de técnicas de carga de imágenes	5

Introducción

Usando imagen original, formatos webp y png optimizados vamos a ver cómo mejorar la carga de las imágenes usando diferentes técnicas. Para ello, compararemos la velocidad de carga de las imágenes usando diferentes técnicas y las anotamos en la tabla del “Anexo I” para su posterior análisis en el apartado “Conclusión”.

1.- Carga de imágenes sin aplicar técnicas

- Crea un único html sencillo e inserta las imágenes: original, img.png, img.webp y img.avif según las dos líneas de abajo. (original.xxx será tu imagen original en el formato xxx que se haya creado).

```




```

- Observa la velocidad de carga de cada imagen, por ejemplo con la herramienta de Chrome “Chrome Dev” (F12”) en la ruta: Red / Img.
- Anota las velocidades en la tabla del Anexo I.

2.- Carga de imágenes usando color promedio

- Calcula la media de color para la imagen: original, png, webp y avif (Puedes usar las técnicas nombradas en las Presentaciones del tema u otras. En el ejemplo de las diapositivas era: #333).
- Crea un único html con las imágenes original, png, webp y avif. Repite el siguiente el código por cada formato (nota: en:img.xxx las "xxx" indica el formato).

```
<div style="background-color: #333">  
    
</div>
```

- Anota las velocidades en la tabla del Anexo I.

3.- Carga de imágenes usando baja resolución

- Por cada imagen: original, png, webp y avif. Crea una imagen de muy poca resolución.
- Crea un único html y usa el siguiente código (adaptado si es necesario) para cada una de las imágenes:
(nota: en:img.xxx las "xxx" indica el formato)

```

```

Y el javascript:

```
function init() {  
  var imgDefer = document.getElementsByTagName('img');  
  for (var i = 0; i < imgDefer.length; i++) {  
    if(imgDefer[i].getAttribute('data-src')) {  
      imgDefer[i].setAttribute('src',imgDefer[i].getAttribute('data-src'));  
    }  
  }  
}  
window.onload = init;
```

- Anota las velocidades en la tabla del Anexo I.

4.- Uso de picture

- A partir de las imágenes: original, png, webp y avif. Crea el resto de imágenes necesarias para ser usadas en el código <picture> dado:

Nota: por cada formato original, png, webp y avif tendrás que usar el código de abajo. Con lo cual por cada formato tendrás que crear una imagen de resolución: 468, 600 y 1024. Es decir, tendrás 16 ficheros:

Imagen original.xxx, original-468.xxx, original-600.xxx y original-1024.xxx
Imagen imgpng.png, imgpng-468.png, imgpng-600.png y imgpng-1024.png
Imagen imgwebp.webp, imgwebp-468.webp, imgwebp-600.webp y imgwebp-1024.webp
Imagen imgavif.avif, imgavif-468.avif, imgavif-600.avif y imgavif-1024.avif

El código a usar es:

```
<picture>
  <source srcset="image-468.jpg 2x, image-468@2x.jpg 3x" media="(min-width: 468px)">
  <source srcset="image-600.jpg 2x, image-600@2x.jpg 3x" media="(min-width: 600px)">
  <source srcset="image-1024.jpg 2x, image-1024@2x.jpg 3x" media="(min-width: 1024px)">
  
</picture>
```

En caso de no funcionar correctamente <picture> prueba a poner un <source> solo por <picture>.

- Crea un único html con dicho código.
- Anota las velocidades en la tabla del Anexo I para un dispositivo tipo monitor.
- Ahora modifica el tipo de dispositivo a móvil (Chrome F12 tiene la opción de cambiar de dispositivo) y recarga el html sin caché (Ctrl-F5 en Chrome).
- Anota las velocidades en la tabla del Anexo I para un dispositivo tipo móvil.

5.- Uso de Preload - Prefetch

- En un único html carga imágenes: original, png, webp y avif en el Preload
- Anota las velocidades en la tabla del Anexo I
- En un único html carga imágenes: original, png, webp y avif en el Prefetch
- Anota las velocidades en la tabla del Anexo I

6.- Uso de image-set

- A partir de las imágenes generadas en el apartado “4.- Uso de picture” adapta el código siguiente por cada imagen y dispositivo: monitor y móvil.

```
.element {  
  background-image: image-set(  
    url(image.jpg) 1x,  
    url(image@2x.jpg) 2x,  
    url(image@3x.jpg) 3x  
  );  
}
```

```
.element {  
  background-image: image-set(  
    url(image.png) 1x,  
    url(image@2x.png) 2x,  
    url(image@3x.png) 3x  
  );  
}
```

- Crea un único html con el código generado..
- Anota las velocidades en la tabla del Anexo I para un dispositivo tipo monitor.
- Ahora modifica el tipo de dispositivo a móvil (Chrome F12 tiene la opción de cambiar de dispositivo) y recarga el html sin caché (Ctrl-F5 en Chrome).
- Anota las velocidades en la tabla del Anexo I para un dispositivo tipo móvil.

7.- Conclusiones

Tras completar la tabla del Anexo I responde a las siguientes preguntas:

- A. ¿Qué técnica es la que mejor resultados aporta?

Con diferencia, lo más efectivo es utilizar tamaños más bajos de imagen mediante picture. Pero esto supone a su vez perder calidad considerablemente.

- B. ¿Existe gran diferencia en las velocidades de carga de los formatos webp y avif en las diferentes técnicas?

Sí. El formato avif carga mucho más rápido que el resto sin importar la técnica utilizada

- C. Para el formato webp ¿qué técnica ofrece mejores velocidades?

Reducir el tamaño de la imagen.

D. Para el formato avif ¿qué técnica ofrece mejores velocidades?

Reducir el tamaño de la imagen.

En general no tengo claro que se obtiene de este análisis. Algunas técnicas suponen cargas sucesivas de diferentes elementos en la página y lo que comparamos al final es cuánto tardó en cargar una imagen que es el último elemento que permanece en la página. Tampoco tengo claro si lo que queremos es el tiempo transcurrido desde que se comienza a cargar o el tiempo de descarga.

En el apartado 4 no entiendo qué se pide con usar dispositivo móvil, ¿no va implícito en el tamaño de pantalla? Hasta donde yo sé no se puede indicar en la mediaquery que la pantalla pertenece a un móvil o a un monitor.

En el ejercicio 6 ninguna de las opciones que me ofrecía el navegador correspondía con la opción x1, por lo que no pude ver los tiempos de carga para 468px.

ANEXO I. Tabla comparativa de técnicas de carga de imágenes

Técnica	Dispositivo	Formato	Velocidad
1 Sin aplicar técnicas	Monitor	original	7.74 ms
		png	14.57 ms
		webp	2.45 ms
		avif	0.62 ms
2 Color promedio	Monitor	original	16.14 ms
		png	10.61 ms
		webp	1.76 ms
		avif	0.4 ms
3 Baja resolución	Monitor	original (baja resolución)	16.53 ms
		png (baja resolución)	33.87 ms
		webp (baja resolución)	22.56 ms
		avif (baja resolución)	10.26 ms
		original	4.68 ms
		png	15.74 ms
		webp	1.24 ms
		avif	0.35 ms
4 Uso de picture	Monitor	original-468	0.65 ms
		png-468	0.38 ms
		webp-468	0.30 ms
		avif-468	0.24 ms
		original-600	0.67 ms
		png-600	0.59 ms
		webp-600	0.46 ms
		avif-600	0.65 ms

		original-1024	1.61 ms
		png-1024	4.73 ms
		webp-1024	4.01
		avif-1024	3.86
	Móvil	original-468	
		png-468	
		webp-468	
		avif-468	
		original-600	
		png-600	
		webp-600	
		avif-600	
		original-1024	
		png-1024	
		webp-1024	
		avif-1024	
5 Uso de Preload - Prefetch	Monitor	original (preload)	7.41 ms
		png (preload)	14.24 ms
		webp (preload)	3.27 ms
		avif (preload)	0.71 ms
		original (prefetch)	10.38 ms
		png (prefetch)	20.53 ms
		webp (prefetch)	7.82 ms
		avif (prefetch)	5.82 ms
6 Uso de image-set	Monitor	original-468	
		png-468	
		webp-468	
		avif-468	
		original-600	0.48 ms

		png-600	0.48 ms
		webp-600	0.49 ms
		avif-600	1.26 ms
		original-1024	3.58 ms
		png-1024	4.62 ms
		webp-1024	1.08 ms
		avif-1024	0.63 ms
	Móvil	original-468	
		png-468	
		webp-468	
		avif-468	
		original-600	2.59 ms
		png-600	3.21 ms
		webp-600	3.18 ms
		avif-600	12.61 ms
		original-1024	14.49 ms
		png-1024	13.89 ms
		webp-1024	10.64 ms
		avif-1024	10.06 ms