

# Web Performance

*la impaciencia de la gente, la ventaja de la tecnología*

# Hola, soy Javier

 [JavierCasares.com](https://JavierCasares.com)

 [Casares.blog](https://Casares.blog)

 [javier@casares.org](mailto:javier@casares.org)

 [@JavierCasares](https://twitter.com/JavierCasares)

[javiercasares.com/wpo](https://javiercasares.com/wpo)

[javiercasares.com/seo](https://javiercasares.com/seo)

[JavierCasares.com](https://JavierCasares.com) [@JavierCasares](https://twitter.com/JavierCasares) [javier@casares.org](mailto:javier@casares.org)



# WPO

# ¿Qué?

- Web
  - Poco a decir, ¿no? 😊
- Performance
  - Rendimiento: Es la palabra mágica.
  - Objetivo: Que vaya lo mejor posible.
- Optimization
  - ¿Really?
  - Lo mismo que “el SEO va de serie”, el WPO también.

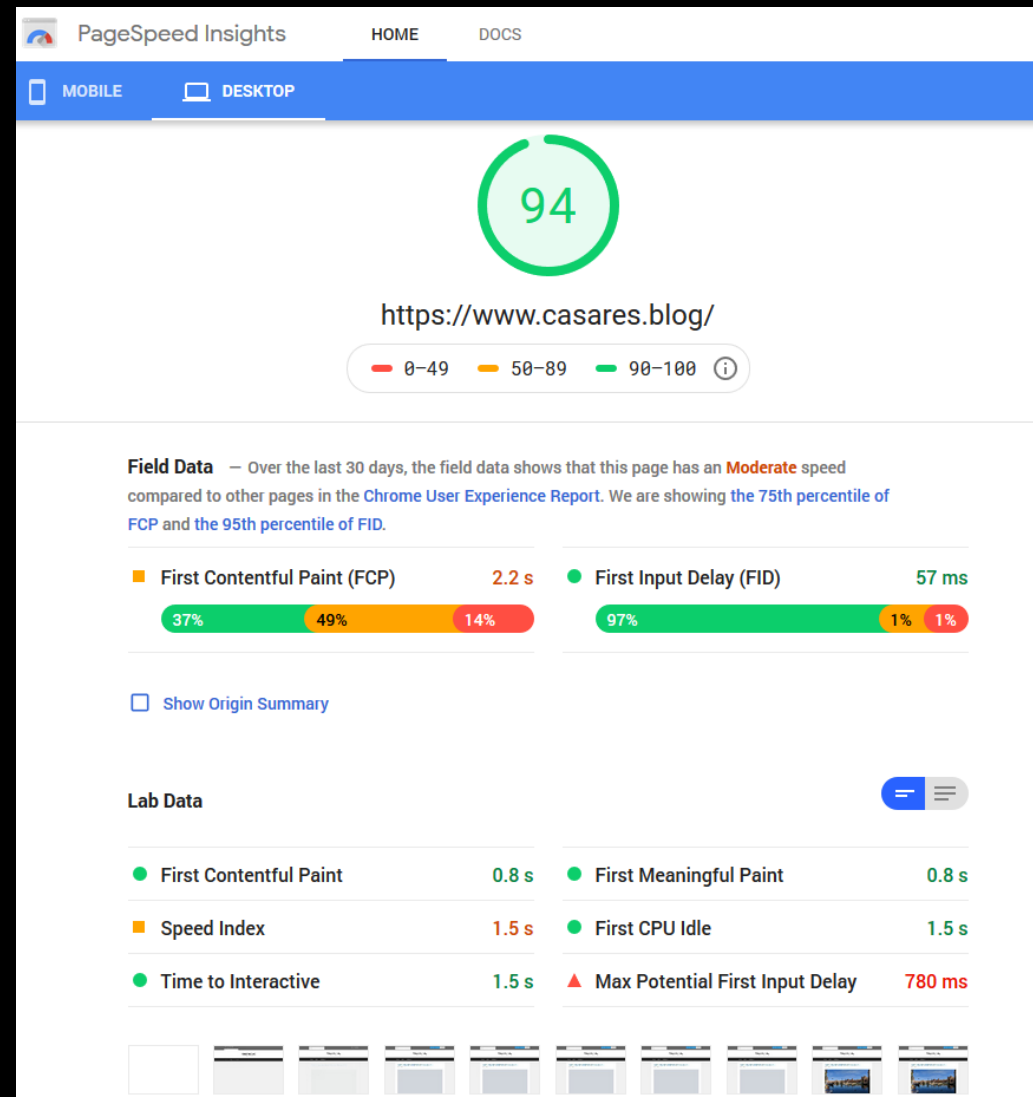
# ¿Cuándo?

- 2009-2010
- Microsoft comenzó a publicar de ello
- Google comenzó a decir que “la velocidad afecta al SEO”
- A diferencia del SEO, el WPO es muy abierto y público
- Los “creadores” de herramientas de WPO, publican TODO.

# ¿Cómo?

- YSlow
- PageSpeed
  
- WebPageTest.org
- Herramientas Waterfall de navegador

# Esto NO es una herramienta WPO



# Esto SÍ es una herramienta WPO

HOME **TEST RESULT** TEST HISTORY FORUMS DOCUMENTATION ABOUT

Web Page Performance Test for <https://www.casares.blog/> [Need help improving?](#)

From: La Rioja, Spain - Chrome - FIOS  
11/9/2019, 10:01:40 AM

**A** **A** **A** **A** **B** **X**

First Byte Time    Keep-alive Enabled    Compress Transfer    Compress Images    Cache static content    Effective use of CDN

**Summary** Details Performance Review Content Breakdown Domains Processing Breakdown Screenshot Image Analysis Request Map

Tester: localhost-82.223.50.181 [Raw page data](#) - [Raw object data](#)  
[Export HTTP Archive \(.har\)](#)  
[View Test Log](#)

[Re-run the test](#)

	Load Time	First Byte	Start Render	First Contentful Paint	Speed Index	Last Painted Hero	First CPU Idle	Document Complete			Fully Loaded			
								Time	Requests	Bytes In	Time	Requests	Bytes In	Cost
First View	2.390s	0.441s	0.800s	0.763s	1.247s	2.100s	> 2.112s	2.390s	38	1,409 KB	3.356s	49	1,570 KB	\$\$\$\$-
Repeat View	1.930s	0.397s	0.800s	0.792s	1.259s	2.100s	> 1.739s	1.930s	4	20 KB	3.036s	6	20 KB	

**Waterfall** **Screenshot**

First View (2.390s)  
[Timeline \(view\)](#)  
[Processing Breakdown](#)  
[Trace \(view\)](#)



# Lo más importante

- El WPO es como la Fórmula 1, todo depende de milisegundos.
- Hay que fasear:
  1. Arreglar “lo gordo”
  2. Ir al detalle, a conseguir reducir décimas de segundo.
- El objetivo es optimizar al máximo, pero que funcione.

# Comenzando por el principio

# Lo inicial en un proyecto Web

- Dominio
- DNS
- IP
- Conectividad

En resumen: el HOSTING.

Si tu hosting es una 🗑️, da igual el SEO, el WPO o lo que sea.

# ¿Qué mirar en un hosting?

- Si buscas WPO, nunca hosting compartido. Mínimo VPS.
- Discos SSD / NVMe de alto rendimiento.
- Capas de caché de datos (Redis, memcached...).
- Capas de caché de página (Varnish...).
- Conectividad de alta calidad (vamos, que Googlebot use el mismo proveedor que tú).
- Tener un sistema de DNS anycast.
- Usar HTTP/3 (sí, 3).

# No sé de qué me hablas

- Si quieres mejorar tu SEO y tu WPO, habla con un SysAdmin.
- El SysAdmin es el profesional que se dedica a optimizar sistemas informáticos, por lo que es tu mejor aliado para mejorar el WPO.
- No olvides que el WPO es tecnología, 100%.

**Vale, entonces ¿qué hago?**

# Lo primero

- Buen hosting.
- Buenas configuraciones:
  - Servidor web (nginx, LiteSpeed, Apache...)
  - Servidor SQL (MariaDB, Percona MySQL, MySQL...)
  - Programación (PHP...)
- Actualizar:
  - Usar últimas versiones de todo = rendimiento.

# Software

- Usa un buen software.
- Que esté bien programado y sea moderno.
- El tiempo de respuesta ha de ser bajo
  - Mezcla de hosting + software
  - Si uno de los dos falla, 🗑️.



# Software

- El resultado, “lo que se ve”, ha de estar optimizado.
- HTML5 de calidad y estándar.
- CSS común + específico
  - Un fichero común para el sitio, y código CSS específico por “plantilla”.
- JavaScript común + específico
  - Un fichero común para el sitio, y código JS específico por “plantilla”.

# Haciendo la web más rápida

- Evita las redirecciones:
  - Se puede hacer “la inicial”
  - Una vez establecido el *hostname*, no hagas más.
- Activa la compresión de contenidos:
  - Deflate (RFC 1951)
  - GZip (RFC 1952)
  - Brotli (RFC 7932)

# Haciendo la web más rápida

- Configurar la caché:
  - Definir el “Cache-Control”
  - Definir los “ETag”
  - Decidir la política de tiempos de caché
  - Caché pública vs. Privada
  - Usar URL consistentes
  - Invalidar la caché
  - Crear otras capas de caché
    - Base de Datos
    - Página
    - CDN “sólo de estáticos”

# Haciendo la web más rápida

- Reducir recursos:
  - “Minify” de HTML, CSS y JavaScript
- Optimizar imágenes:
  - Usar el formato adecuado (JPG, GIF, PNG, WEBP...)
    - Usa PNG mejor que GIF (si no es animado)
    - Elimina el “alpha channel” de PNG
    - Calidad 85% en JPG
    - Chroma sampling de 4:2:0 en JPG
    - Si JPG > 10KB usar formato progresivo
  - Usar los tamaños adecuados (media-queries)

# Haciendo la web más rápida

- Optimizar la entrega de recursos:
  - Servir primero los CSS genéricos básicos (link rel="stylesheet")
  - Servir asíncrono el resto de CSS.
  - No usar CSS inline (<p style=...>).
- Elimina JavaScript Render-Blocking
  - Carga el JavaScript de forma diferida o asíncrona

# Interpreta los datos

# Lee y piensa

*Esto ya es un consejo vital*

- Cuando revises los datos de las distintas herramientas:
  - Has de saber el origen de los datos
  - Has de saber qué significan los datos
  - Has de saber para qué sirven esos datos
  - Has de saber cómo afectan esos datos
  - Has de saber cómo optimizar gracias a los datos

# Piensa

Y deja esto en manos de profesionales

- El TTFB (Time To First Byte) es “alto”:
  - El hosting es una 🗑️. Cambia de hosting.
  - El hosting está mal configurado (servidor web, base de datos...)
  - El software es una 🗑️. Cambia de software.
  - El software está mal configurado (hay un plugin que rompe todo...)
- No está el Keep-alive:
  - Actívalo (y ya está)



# Piensa

Y deja esto en manos de profesionales

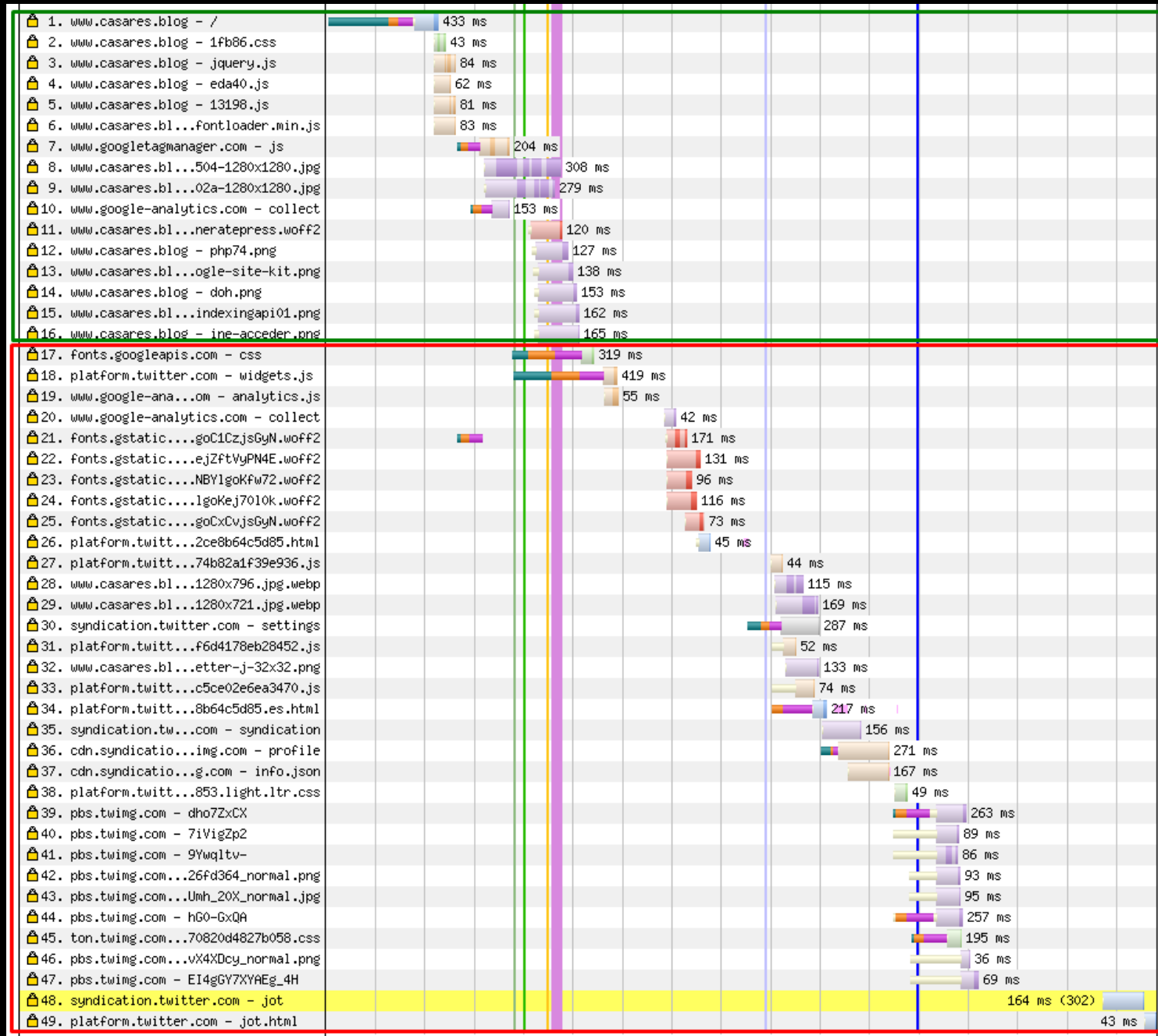
- El Compress Transfer no es correcto:
  - Actívalos (todos a la vez, que se puede)
    - Brotli (mejor)
    - GZip (habitual)
    - Deflate (peor)
- Las imágenes no están optimizadas
  - Los diseñadores a diseñar (y optimizar)
  - Las imágenes se han de optimizar ANTES de subirlas a la web
  - Las herramientas de optimización son una 🙄. Hazlo TÚ.

# Piensa

Y deja esto en manos de profesionales

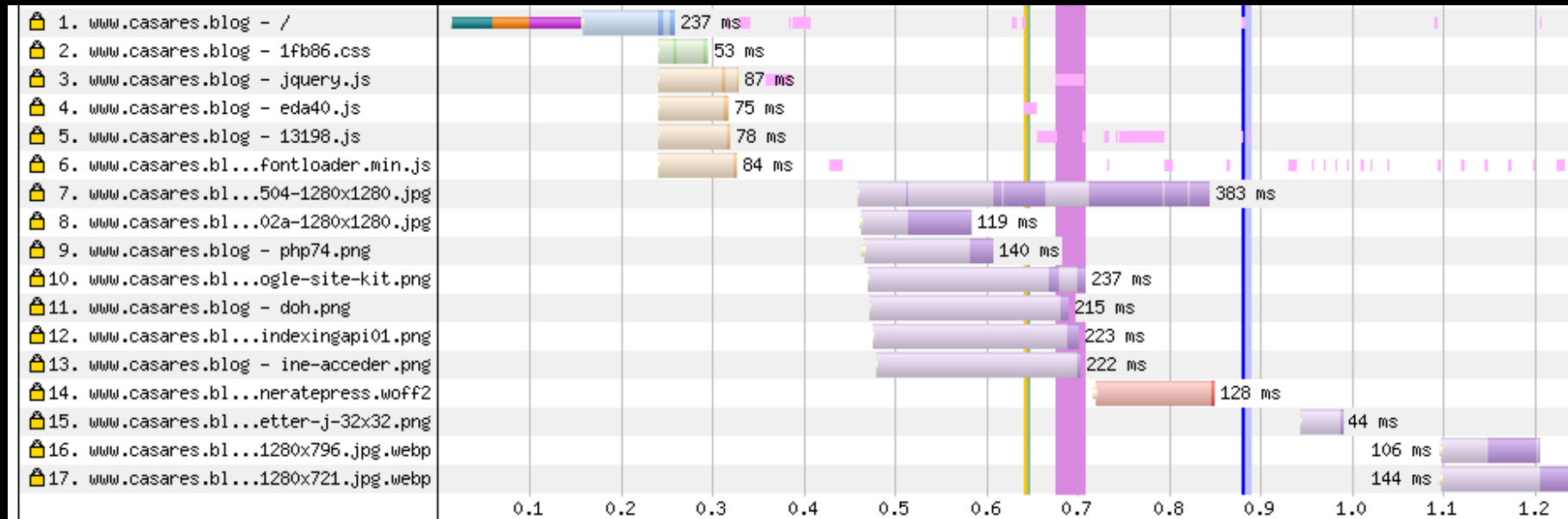
- No usas una CDN:
  - Es que lo más probable es que no te haga falta.
- Hay scripts que van mal:
  - ¿Qué culpa tengo yo se que Google Analytics, Facebook, Twitter y otros sean la lacra de Internet?
  - No uses herramientas de terceros y harás un mundo mejor sin promocionar empresas que viven de tus datos personales.
  - ¿Sabes que se pueden optimizar esos scripts? (Bueno, una parte)

- Hay 49 peticiones
- 17 son mías
- Load Time: 2,39 s



- Hay 17 peticiones
- 17 son mías
- Load Time: 0,88 s

**Entre 2 y 3 veces  
más rápido**



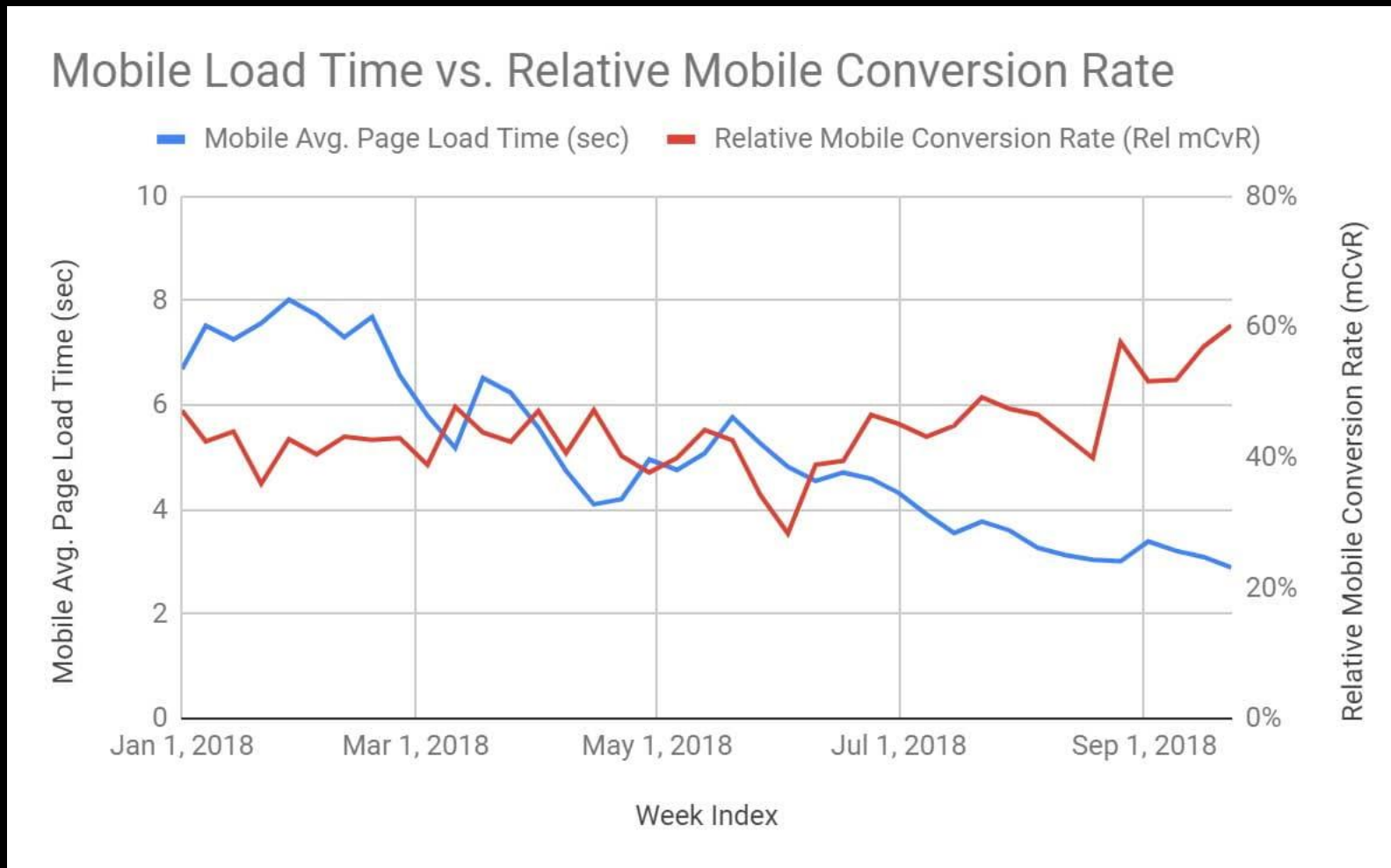
# Resumiendo

# Resumiendo

1. Invierte en alojamiento de calidad
2. Configura y optimiza tu hosting de calidad
3. Usa software de calidad ¿has probado WordPress?
4. Optimiza el código de tu sitio
5. Elimina la 🗑️ de terceros que no necesites.
6. SEO ≠ WPO no suelen ser la misma persona humana
7. Disfruta de la velocidad de tu sitio optimizado.

# El valor de la velocidad

The value of speed <https://web.dev/value-of-speed/>



# Gracias