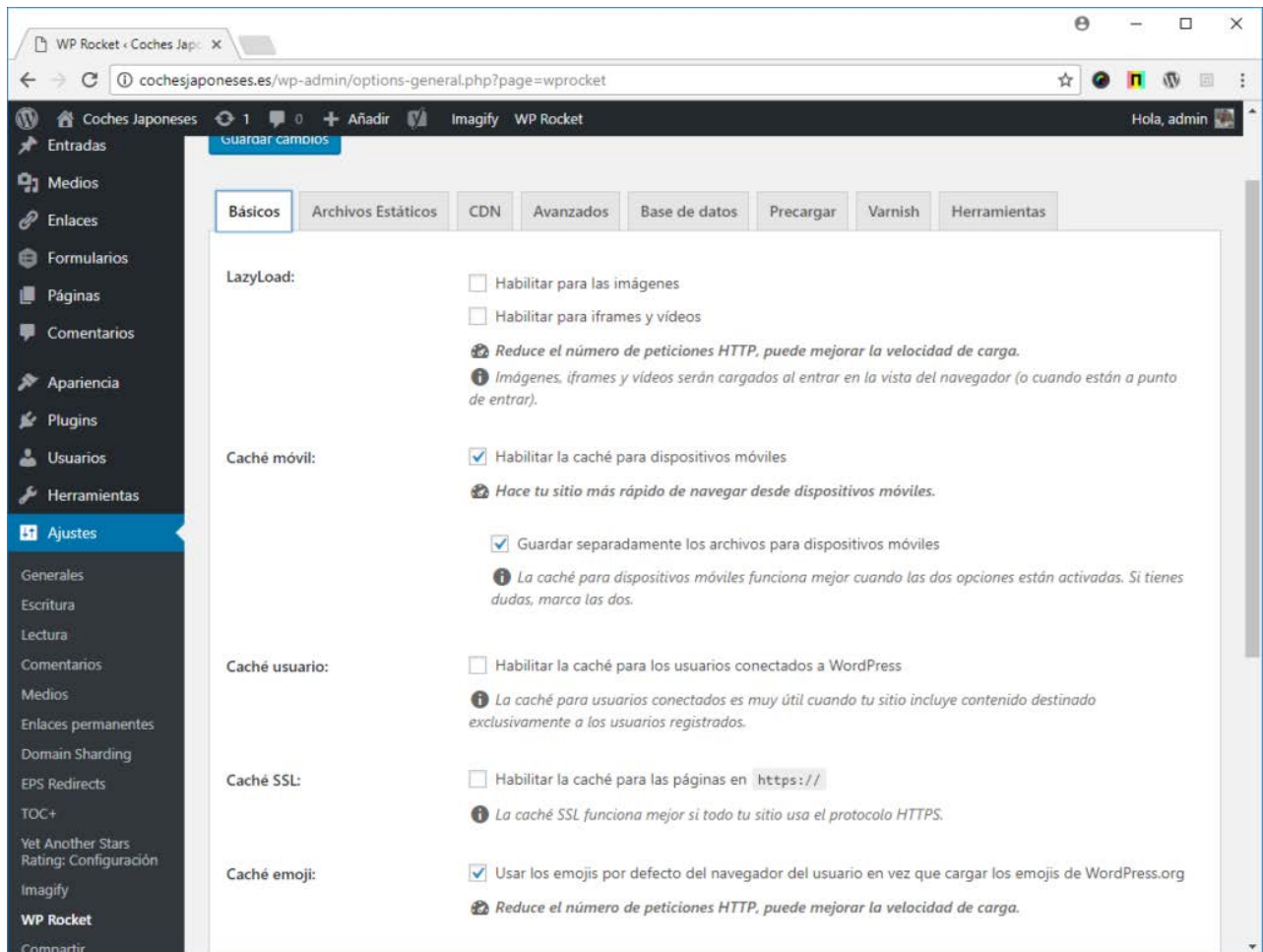


CONFIGURAR WP ROCKET PARA OPTIMIZAR WORDPRESS

Una vez que **instalamos y activamos WP Rocket** se añade a nuestro **dashboard de WordPress**, a la sección “Ajustes” una nueva sección de configuración llamada “**WP Rocket**”.



En cada una de las pestañas que puedes ver en la imagen anterior tenemos distintas **opciones de configuración**, aunque el **cache** (que es la actividad principal del plugin) se activa automáticamente al **activar WP Rocket**.

Antes de entrar en materia, te dejo este video que he grabado para que veas un poco por encima la **configuración de WP Rocket**:

CONFIGURACIÓN “BÁSICOS” DE WP ROCKET

La sección de “Básicos” es la base de todo, aquí se encuentran las opciones más generales de **WP Rocket** y algunas opciones variadas que no tienen hueco en otros lados.

Básicos Archivos Estáticos CDN Avanzados Base de datos Precargar Varnish Herramientas

LazyLoad:

- Habilitar para las imágenes
- Habilitar para iframes y vídeos
- Reduce el número de peticiones HTTP, puede mejorar la velocidad de carga.*
- Imágenes, iframes y vídeos serán cargados al entrar en la vista del navegador (o cuando están a punto de entrar).*

Caché móvil:

- Habilitar la caché para dispositivos móviles
- Hace tu sitio más rápido de navegar desde dispositivos móviles.*
- Guardar separadamente los archivos para dispositivos móviles
- La caché para dispositivos móviles funciona mejor cuando las dos opciones están activadas. Si tienes dudas, marca las dos.*

Caché usuario:

- Habilitar la caché para los usuarios conectados a WordPress
- La caché para usuarios conectados es muy útil cuando tu sitio incluye contenido destinado exclusivamente a los usuarios registrados.*

Caché SSL:

- Habilitar la caché para las páginas en
- La caché SSL funciona mejor si todo tu sitio usa el protocolo HTTPS.*

Caché emoji:

- Usar los emojis por defecto del navegador del usuario en vez que cargar los emojis de WordPress.org
- Reduce el número de peticiones HTTP, puede mejorar la velocidad de carga.*

Contenido incrustado:

- Desactivar la función WordPress para el contenido incrustado
- Evita que tu contenido sea incrustado en sitios de terceros, evita que tu incrustes contenido desde otros sitios (no aprobados) y suprime las peticiones JavaScript relacionadas a la [función WordPress para el contenido incrustado](#).*

Vida útil de la caché:

Especificar el tiempo tras del cual quieres vaciar la caché global (0 = ilimitado)

hora(s)

- La vida útil de la caché es el plazo temporal a partir del cual todos los archivos de caché serán eliminados. Activa la [Precarga](#) para que la caché se reconstruya automáticamente luego de la expiración de su vida útil.*
- Aconsejamos reducir la vida útil de la caché a 10 horas o menos en caso de que veas aparecer problemas con más frecuencia.*

Las opciones que puedes ver en la imagen anterior son estas:

- **Lazy Load:** WP Rocket permite activar e implementar lazy load para imágenes, iframes y videos. Por si no lo sabes, Lazy Load es una técnica que permite cargar ciertos elementos de la web mediante JavaScript cuando se hace scroll y el usuario los ve en pantalla. Esto ahorra peticiones en la carga de la página.
- **Cache móvil:** WP Rocket permite elegir si queremos tener activo el cache para dispositivos móviles o incluso tener un cache específico para la versión móvil, algo muy útil si utilizas plugins como WP Touch.
- **Cache usuario:** Podemos elegir si queremos ofrecerles páginas cacheadas a los usuarios autenticados en el sitio web o si queremos excluirlos, lo normal es excluirlos para evitar problemas al actualizar y publicar en el sitio web.
- **Cache SSL:** Podemos elegir si queremos servir o no servir páginas cacheadas a los usuarios que entran a nuestro sitio web a través de **HTTPS**, normalmente esta casilla se activa por defecto (sin posibilidad de desmarcarla) cuando se activa HTTPS.
- **Cache emoji:** Nos permite usar los emojis del ordenador / navegador en lugar de que se tengan que descargar los

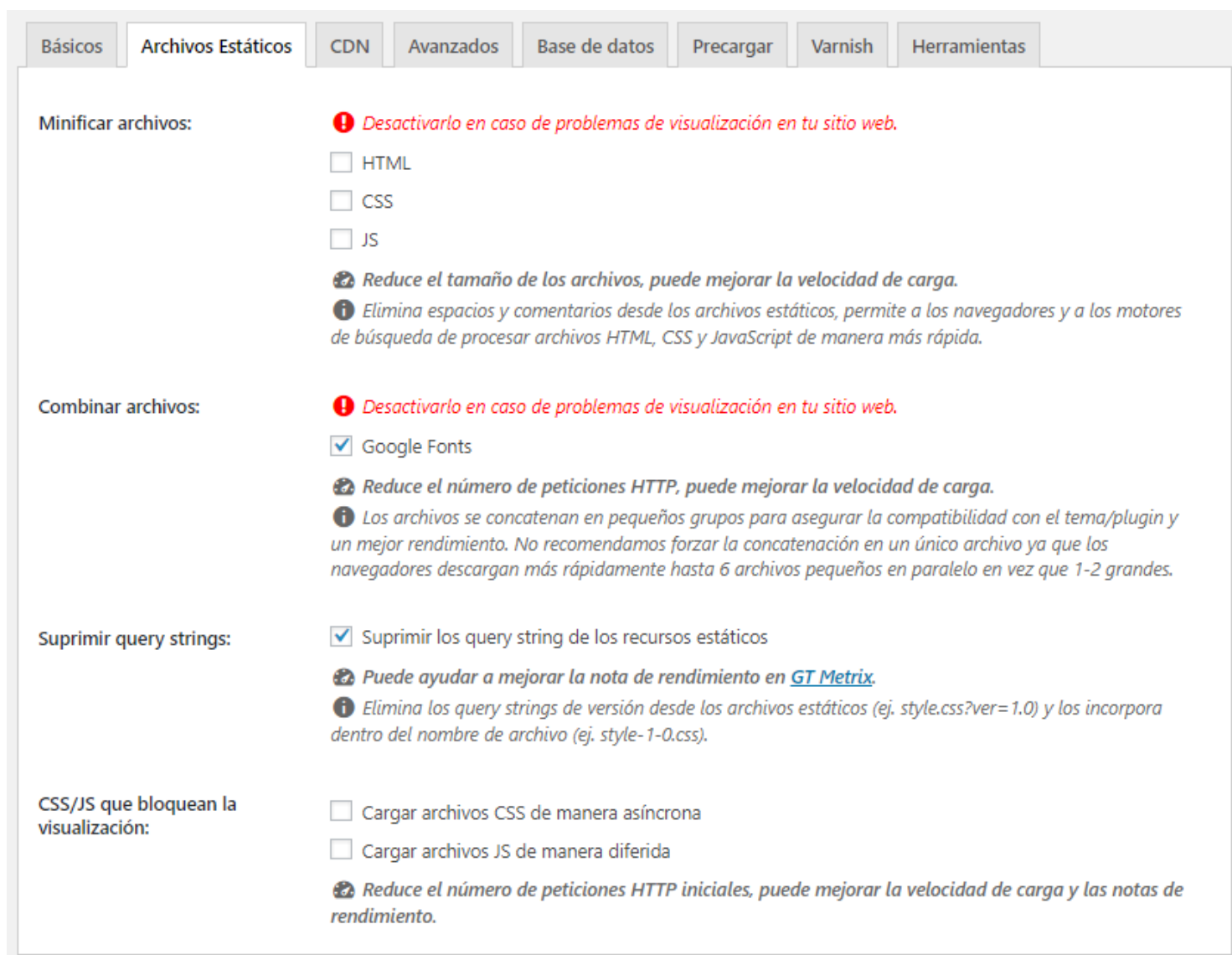
emoji desde los repositorios de WordPress.org, esto no es muy notable en cuanto a velocidad, pero puede ayudar en casos donde se utilicen muchos emojis como [foros u otros sitios web sociales creados con WordPress](#).

- **Contenido incrustado:** Esto nos permite activar o desactivar el contenido incrustado o insertado, es decir, hace que no se carguen los incrustados para que el sitio web se renderize mucho más rápido y no consuma tantos recursos en el navegador del visitante.
- **Vida útil de la cache:** Nos permite especificar el tiempo máximo de vida de las páginas cacheadas antes de que tengan que volver a cachearse. Si nuestro sitio web es un blog o una página estática podríamos poner “0” y confiar en que las políticas de vaciado de WP Rocket puedan vaciar el cache de forma efectiva en las actualizaciones o en las publicaciones de nuevo contenido.

En principio estas son todas las opciones que tiene **WP Rocket** en la pestaña “Básicos”, aunque como he dicho antes pueden aumentar en cualquier momento ya que es un plugin que se actualiza mucho y esta sección de “Básicos” da acogida a muchas opciones que no encajan en ningún otra parte de la **configuración de WP Rocket**.

CONFIGURACIÓN “ARCHIVOS ESTÁTICOS” DE WP ROCKET

En la sección de archivos estáticos se encuentran todas las opciones de configuración destinadas a la **minificación, optimización y combinado de archivos HTML, CSS y Javascript**, además de la optimización en la carga de **Google Fonts** desde los **servidores de Google**.



The screenshot shows the 'Archivos Estáticos' (Static Files) configuration tab in WP Rocket. It features a navigation bar with tabs: Básicos, Archivos Estáticos, CDN, Avanzados, Base de datos, Precargar, Varnish, and Herramientas. The main content area is divided into four sections:

- Minificar archivos:** Includes a warning icon and text: "Desactivarlo en caso de problemas de visualización en tu sitio web." Below are three unchecked checkboxes: HTML, CSS, and JS. A gear icon indicates a tip: "Reduce el tamaño de los archivos, puede mejorar la velocidad de carga." An information icon indicates another tip: "Elimina espacios y comentarios desde los archivos estáticos, permite a los navegadores y a los motores de búsqueda de procesar archivos HTML, CSS y JavaScript de manera más rápida."
- Combinar archivos:** Includes a warning icon and text: "Desactivarlo en caso de problemas de visualización en tu sitio web." Below is a checked checkbox for Google Fonts. A gear icon indicates a tip: "Reduce el número de peticiones HTTP, puede mejorar la velocidad de carga." An information icon indicates another tip: "Los archivos se concatenan en pequeños grupos para asegurar la compatibilidad con el tema/plugin y un mejor rendimiento. No recomendamos forzar la concatenación en un único archivo ya que los navegadores descargan más rápidamente hasta 6 archivos pequeños en paralelo en vez que 1-2 grandes."
- Suprimir query strings:** Includes a checked checkbox for "Suprimir los query string de los recursos estáticos". A gear icon indicates a tip: "Puede ayudar a mejorar la nota de rendimiento en [GT Metrix](#)." An information icon indicates another tip: "Elimina los query strings de versión desde los archivos estáticos (ej. style.css?ver=1.0) y los incorpora dentro del nombre de archivo (ej. style-1-0.css)."
- CSS/JS que bloquean la visualización:** Includes two unchecked checkboxes: "Cargar archivos CSS de manera asíncrona" and "Cargar archivos JS de manera diferida". A gear icon indicates a tip: "Reduce el número de peticiones HTTP iniciales, puede mejorar la velocidad de carga y las notas de rendimiento."

Las opciones que podemos encontrar en esta sección “Archivos estáticos” de la **configuración de WP Rocket** son

estos:

- **Minificar archivos:** Permite activar el sistema para minificar y reducir el peso de los archivos HTML, Javascript y CSS. WP Rocket utiliza un sistema de minificado propio que elimina los espacios, los saltos de línea y los comentarios del código HTML, CSS y JS.
- **Combinar archivos:** Permite combinar varios archivos de un mismo tipo en uno, en el caso de Google Fonts permite que se descarguen en un único archivo todas las fuentes de Google Fonts que utilice el sitio web y con esto se reduce la complejidad y las peticiones realizadas a servidores externos.
- **Suprimir query strings:** Esto es algo básico en WPO actualmente, lo que hace es [eliminar los query strings](#) de las URL de carga de los recursos estáticos tipo Javascript o CSS.
- **CSS/JS que bloquean la visualización:** Permite cargar el javascript y el CSS de manera asíncrona y diferida para que no interfieran en la carga general de la web. Estos métodos o técnicas suelen dar problemas con los themes y plugins más complejos, pero si los podemos solucionar, son muy beneficiosos para aprovechar mejor el [crawl Budget](#) de los principales buscadores.

La configuración de esta sección es complicada y es la que más problemas puede dar, dado la complejidad de los themes y plantillas actuales.

CONFIGURACIÓN “CDN” DE WP ROCKET

La sección “CDN” nos permite configurar el [Domain Sharding o Carga Paralela](#) para que podamos configurar cualquier servicio de **CDN** que cargue mediante subdominios.

Básicos Archivos Estáticos **CDN** Avanzados Base de datos Precargar Varnish Herramientas

Cloudflare Mostrar la pestaña de los ajustes Cloudflare

CDN: Habilitar el Content Delivery Network
i Todos los URLs de los archivos estáticos (CSS, JS, imágenes) serán reescritos por el CNAME(s) indicado debajo.

CNAME(S) del CDN:
 reservado para **x**
 reservado para **x**

CDN sin SSL: Desactivar la funcionalidad CDN para las páginas con HTTPS
i Si la cuenta con tu CDN no soporta el SSL, aquí puedes desactivar la reescritura de los URLs para las páginas HTTPS.

Excluir archivos: *Especificar el URL(s) de los archivos que no se deberían enviar a través del CDN (uno por línea)*

i La porción de URL con el dominio será eliminada automáticamente.
Puedes usar los caracteres comodines **(.*)** para excluir todos los archivos del mismo tipo que se encuentran en una ruta específica.

En la sección “CDN” nos encontramos las siguientes opciones:

- **CloudFlare:** Esta opción simplemente activa la pestaña de configuración llamada “CloudFlare” donde se hacen los ajustes para la conexión con el API de CloudFlare y la posterior posibilidad de vaciar el cache de CloudFlare desde WP Rocket.
- **CDN:** Esta casilla permite habilitar la carga de elementos desde el CDN a través de subdominios o hosts que vamos a configurar en la opción CNAME(S) del CDN.
- **CNAME(S) del CDN:** Esta opción permite configurar los CNAME o subdominios que se utilizarán para cargar los elementos que se cargaran desde el CDN, además podemos seleccionar qué elementos se cargarán desde cada CNAME segmentando por imágenes, CSS y Javascript.
- **CDN sin SSL:** Permite desactivar el CDN para páginas que carguen con HTTPS, útil para tiendas online que tengan el contenido normal con HTTP pero la parte del carrito y los pasos previos a la pasarela de pago con HTTPS.
- **Excluir archivos:** Nos permite hacer un listado de archivos que no serán cargados desde el CDN, útil si algunos tipos de archivos muy concretos nos da algún problema. La herramienta de exclusión es bastante potente e incluso acepta la posibilidad de usar comodines.

La configuración de la sección “CDN” es sencilla, pero debemos de tener en cuenta que previamente tenemos que hacer toda la configuración previa en el [servicio CDN](#) que nosotros queramos.

Nosotros normalmente recomendamos [CDN77](#), [KeyCDN](#) o [Amazon CloudFront](#) como servicios de CDN Premium (la única opción gratuita que recomendamos y que sirva para proyectos más o menos serios es [CloudFlare](#)).

CONFIGURACIÓN “CLOUDFLARE” DE WP ROCKET

Esta pestaña solo aparece cuando activamos la opción de **CloudFlare** en la sección **CDN**, es decir, normalmente esta pestaña esta oculta.

Básicos	Archivos Estáticos	CDN	Avanzados	Base de datos	Precargar	Cloudflare	Varnish	Herramientas
Dirección email:	Ingresar la dirección email que usas para tu cuenta Cloudflare							
	<input type="text"/>							
Clave API global:	Ingresar la clave API global de tu cuenta Cloudflare							
	<input type="text"/>							
	i Recuperar tu clave API							
Dominio:	<input type="text" value="cochesjaponeses.es"/>							
Modo desarrollo:	<input type="text" value="Off"/>							
	i Activa temporalmente el modo desarrollador en tu sitio web. Esta opción se desmarcará automáticamente luego de 3 horas. Aprende más							
Ajustes óptimos:	<input type="text" value="Off"/>							
	i Mejora automáticamente tu configuración Cloudflare para aumentar la velocidad, la nota de rendimiento y la compatibilidad.							
Protocolo relativo:	<input type="text" value="Off"/>							
	i Debería ser usado exclusivamente a lado de la opción Flexible SSL de Cloudflare. Los URLs de los archivos estáticos (CSS, JS, imágenes) serán reescritos usando el protocolo relativo (//) en vez de <code>http://</code> o <code>https://</code>.							
Vaciar la caché Cloudflare:	<input type="button" value="Vaciar la caché Cloudflare"/>							
	i Vacía los recursos de tu sitio web copiados en la caché. Aprende más							

No voy a explicar esta sección al completo, ya que la mayoría de los campos se utilizan para conectarse al **API de CloudFlare**, además que el **módulo de CloudFlare** que tiene **WP Rocket** SOLO sirve para tareas básicas como vaciar cache o para mantener el modo desarrollador activo desde el plugin.

Por otro lado, mediante la opción “Ajustes óptimos” cargaremos en **CloudFlare** la configuración que recomienda **WP Rocket**, algo que personalmente no me gusta, pero también supongo que será una cuestión de gustos.

CONFIGURACIÓN “AVANZADOS” DE WP ROCKET

La sección de “Avanzados” de la **configuración de WP Rocket** nos permite moldear el comportamiento de **WP Rocket** en cuanto a la gestión de cache y a las exclusiones.

La mayoría de los campos que podemos encontrar en esta sección, como puedes ver a continuación, están dedicados a la gestión de exclusiones y purga, modificando las políticas de vaciado y cache predeterminadas de **WP Rocket**.

Básicos	Archivos Estáticos	CDN	Avanzados	Base de datos	Precargar	Varnish	Herramientas
---------	--------------------	-----	------------------	---------------	-----------	---------	--------------

Nunca poner en caché (URLs): *Especificar los URLs de las páginas o de las entradas que no deberían ser copiadas en la caché (una por línea)*

/members/(.*)

i La porción de URL con el dominio será eliminada automáticamente.
Puedes usar los caracteres comodines `(.*)` para identificar múltiples URLs con la misma ruta.

Nunca poner en caché (cookies): *Especificar el ID de los cookies que, una vez activados en el navegador del usuario, deberían evitar que una página se copie en la caché (un ID por línea)*

Nunca poner en caché (agentes de usuario): *Especificar los agentes de usuario que nunca deberían acceder a las páginas en caché (una cadena por línea)*

(.*)Mobile(.*)Safari(.*)

i Usar los caracteres comodines `(.*)` para [identificar partes del nombre del agente de usuario](#).

Purgar siempre (URLs): *Especificar los URLs que quieres purgar de la caché cada vez que actualices una entrada o una página (un URL por línea)*

i La porción de URL con el dominio será eliminada automáticamente.
Puedes usar los caracteres comodines `(.*)` para identificar múltiples URLs con la misma ruta.

Poner en caché los query strings: *Especificar los query strings que quieres poner en caché (uno por línea):*

i La caché para los query strings te permite de forzar la creación de una caché para uno parámetros GET específicos.

Estas son las opciones que tenemos disponibles en la sección “Avanzados”:

- **Nunca poner en cache (URLs):** Esta opción nos permite añadir un listado de URL del sitio web que nunca se cachearan, esto es útil para añadir paginas específicas como el carrito, alguna página de perfil o paginas muy dinámicas que muestren datos específicos dependiendo del visitante.
- **Nunca poner en cache (cookies):** Esta opción también nos permite añadir un listado, pero en este caso se trata de un listado de cookies que si son detectadas en el navegador del visitante no se mostrará una página cacheada ni se copiará esa página a cache.
- **Nunca poner en cache (agentes de usuario):** Esta opción es exactamente igual a las dos anteriores, pero en este caso no se añadirán a cache páginas que sean cargadas desde el user-agent o agente de usuario correspondiente al listado que nosotros especifiquemos.

- **Purgar siempre (URLs):** Esta opción nos permite modificar las políticas de vaciado de cache, ya que nos permite especificar un listado de URL del sitio web que serán vaciadas del cache guardado siempre que se publique o actualice contenido.
Normalmente WP Rocket vacía el cache de los loop de WordPress y de los archivos de WordPress cuando se actualiza o publica contenido, pero con esta opción podremos especificar nosotros otras páginas para que su cache sea actualizado.
- **Poner en cache los query strings:** Esta opción es útil para cachear páginas que se carguen o que carguen contenido específico mediante parámetros indicados por GET en la URL, ya que normalmente estas páginas no serán cacheadas.

Esta es otra de las secciones que más va cambiando en cada versión de **WP Rocket** o cada vez que añaden funcionalidades y características al plugin, por eso lo que he escrito en el momento de escribir este artículo, puede que no se corresponda con lo que te encuentres cuando leas el artículo.

CONFIGURACIÓN “BASE DE DATOS” DE WP ROCKET

La **herramienta de limpieza y optimización de base de datos de WP Rocket** es muy simple, es muy similar a [WP-Optimize](#) o cualquier otro limpiador.

Esta herramienta o módulo para **WP Rocket** ha aparecido hace algunas versiones y personalmente puedo decir que no lo he usado mucho dado que creo que hay opciones mejores, aunque supongo que la posibilidad de **programar limpiezas automáticas** ayuda bastante.

Básicos Archivos Estáticos CDN Avanzados **Base de datos** Precargar Varnish Herramientas

Limpieza de las entradas:

- Revisiones
81 revisiones en tu base de datos.
- Borradores automáticos
2 borradores en tu base de datos.
- Entradas descartadas
5 entradas descartadas en tu base de datos.

Limpieza de los comentarios:

- Comentarios spam
0 comentarios spam en tu base de datos.
- Comentarios descartados
0 comentarios descartados en tu base de datos.

Limpieza transients:

- Transients expirados
1521 transients expirados en tu base de datos.
- Todos los transients
3101 transients en tu base de datos.

Limpieza base de datos:

- Tablas optimizadas
0 tablas para optimizar en tu base de datos.

Limpieza automática:

- Programar limpieza automática

Ejecutar limpieza:

“Guardar y optimizar” guardará tus ajustes y ejecutará una optimización de la base de datos: “Guardar cambios” abajo guardará exclusivamente los ajustes.

No voy a explicar con pelos y señales cómo [realizar una limpieza con esta herramienta](#) ya que no quiero mojarme, hay que tener en cuenta que esto **BORRA cosas de la base de datos** y por lo tanto debemos ser totalmente conscientes de lo que estamos haciendo en cada momento, además de tener una buena **copia de seguridad** preparada por si hubiera algún problema de compatibilidad o similar que ocasionara una catástrofe.

Quizás el elemento que se almacena en la **base de datos de WordPress** y que más afecta al **WPO** son los **transients**, unos elementos que se guardan en la tabla de “options” y que cuando se acumulan muchos pueden **causar problemas y ralentizaciones generales**.

CONFIGURACIÓN “PRECARGAR” DE WP ROCKET

La parte de “Precargar” a mí me encanta, ya que realizando una buena configuración de la **precarga de cache** junto con un buen ajuste de las políticas de **vaciado de cache** podemos hacer que el **cache** funcione de una forma dinámica, automatizada y EFECTIVA.

Básicos Archivos Estáticos CDN Avanzados Base de datos **Precargar** Varnish Herramientas

Precargar la sitemap:

- Activar la función de precarga caché a través del sitemap
 - La precarga del sitemap se ejecuta automáticamente al caducar de la vida útil de la caché. Puedes ejecutarla también de manera automática desde el menu en la barra superior o desde la pestaña Herramientas.*
- Sitemap XML de Yoast SEO
 - Detectamos automáticamente la sitemap generada por el plugin Yoast SEO. Puedes marcar la opción para precargarla.*

Intervalo análisis sitemap: 250ms tiempo de espera entre el análisis de un URL y el siguiente.

- ¡Aumenta este valor si notas problemas de sobrecarga en tu servidor!*

Sitemaps para precargar: Especificar los sitemap(s) XML que quieres usar para la precarga (uno por línea)

http://cochesjaponeses.es/sitemap.xml

Robot de precarga:

- Manual
- Automático
- La precarga a través del robot debería usarse exclusivamente con servidores de alto rendimiento. Una vez activado, el robot se lanza automáticamente cada vez que añades o actualizas el contenido de tu sitio web. Puedes ejecutarlo también manualmente desde la barra del menu admin o desde la pestaña Herramientas.*
- ¡Desactivar estas opciones en caso de que notes problemas de sobrecarga en tu servidor!*

Precarga de peticiones DNS: Especificar los servidores externos que quieres precargar (sin http: , uno por línea)

//example.com

- La precarga DNS ayuda a cargar los archivos externos más rápidamente, sobretudo con redes móviles.*

Vamos a intentar desglosar una por una las opciones disponibles en la parte de “Precargar” de **WP Rocket**:

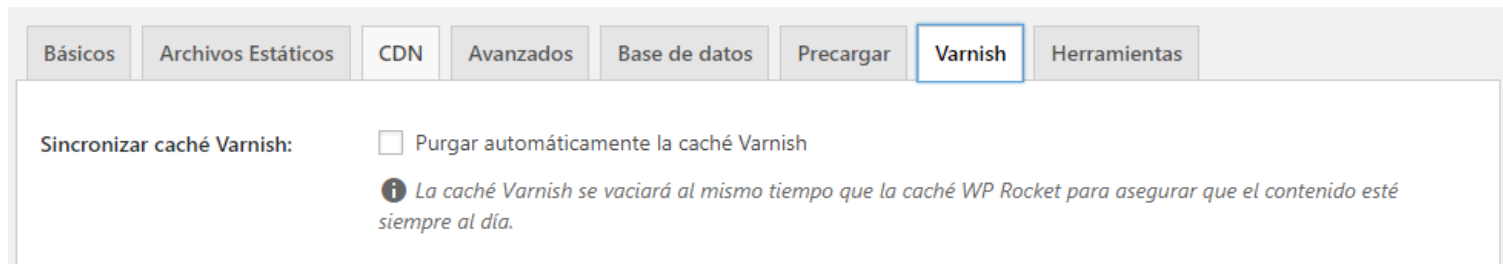
- **Precargar la sitemap:** La precarga de cache de WP Rocket se apoya en el sitemap para tener un listado de URL del sitio web para cargar en la cache, esto lo hace mucho más efectivo. Además es capaz de reconocer los principales [plugins de SEO](#) que crean [sitemaps.xml](#), como por ejemplo [Yoast SEO](#) y [All in SEO Pack](#).
- **Intervalo análisis sitemap:** Nos permite configurar el tiempo entre rastreos del bot que se encarga del precache, cuanto más bajo sea el tiempo mayor será el impacto en los recursos consumidos en el servidor o hosting que aloja la web.
- **Sitemaps para precargar:** En caso de que WP Rocket no sea capaz de reconocer el sitemap, siempre podemos añadir su URL directamente en este campo o añadir sitemaps complementarios que permitan a WP Rocket conocer la estructura del sitio web para cachearlo.
- **Robot de precargar:** Podemos elegir si queremos ejecutar el robot de precarga de forma automática o manual, si es manual podremos ejecutarla desde la barra de herramientas de WordPress y si es automatico se ejecutara cada cierto tiempo sin intervención del usuario y con los parámetros configurados en esta parte de la configuración.
- **Precarga de peticiones DNS:** Una de las cosas que diferencia a WP Rocket frente a otros plugins de este tipo es el DNS Prefetch, es decir, la precarga de peticiones DNS. Esto es útil para optimizar la carga de peticiones a hosts externos como por ejemplo Google, Facebook, Twitter u otros servicios que hagan muchas peticiones a distintos

servidores.

Como he dicho antes, a mí me encanta la funcionalidad de precarga que tiene **WP Rocket**, ya que me parece mucho mejor que la que trae [W3 Total Cache](#) o [WP Super Cache](#), además mucho más fácil de configurar desde una interfaz mucho más clara.

CONFIGURACIÓN “VARNISH” DE WP ROCKET

La sección de **configuración de “Varnish”** es la más pequeña de todas, ya que solo tiene una opción de configuración.

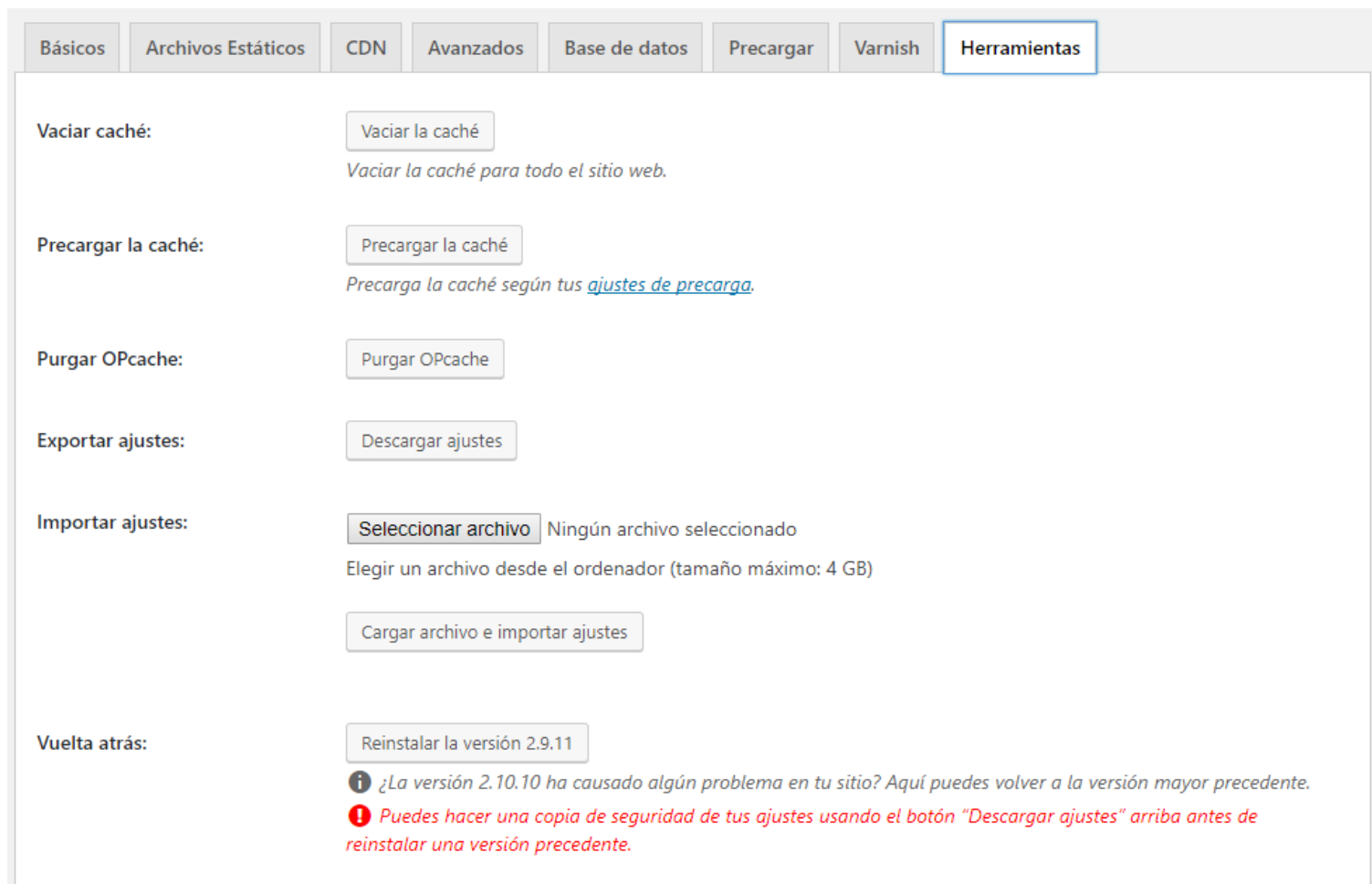


The screenshot shows the WP Rocket configuration interface with the 'Varnish' tab selected. The 'Sincronizar caché Varnish' section contains a checkbox for 'Purgar automáticamente la caché Varnish' which is unchecked. Below it is an information icon and a note: 'La caché Varnish se vaciará al mismo tiempo que la caché WP Rocket para asegurar que el contenido esté siempre al día.'

Esta opción de configuración nos permite **configurar el vaciado de cache de Varnish** (si lo tenemos instalado y funcionando en el servidor local) cuando realicemos el **vaciado de cache de WP Rocket** o cuando se vacíe mediante las políticas de **vaciado de cache**.

CONFIGURACIÓN “HERRAMIENTAS” DE WP ROCKET

La sección de **herramientas de WP Rocket** no es una sección de configuración, sino que es más una sección con ciertos accesos directos para realizar ciertas acciones.



The screenshot shows the WP Rocket configuration interface with the 'Herramientas' tab selected. It features several action buttons: 'Vaciar la caché' (with a note 'Vaciar la caché para todo el sitio web.'), 'Precargar la caché' (with a note 'Precarga la caché según tus ajustes de precarga.'), 'Purgar OPcache', 'Exportar ajustes' (with a 'Descargar ajustes' button), 'Importar ajustes' (with a 'Seleccionar archivo' button, a note 'Ningún archivo seleccionado', a note 'Elegir un archivo desde el ordenador (tamaño máximo: 4 GB)', and a 'Cargar archivo e importar ajustes' button), and 'Vuelta atrás' (with a 'Reinstalar la versión 2.9.11' button). At the bottom, there are two information icons: one with a note '¿La versión 2.10.10 ha causado algún problema en tu sitio? Aquí puedes volver a la versión mayor precedente.' and another with a red warning icon and a note 'Puedes hacer una copia de seguridad de tus ajustes usando el botón “Descargar ajustes” arriba antes de reinstalar una versión precedente.'

Estas son las opciones que podemos encontrar en la sección de **“Herramientas” de WP Rocket**:

- **Vaciar cache:** Permite vaciar la cache de WP Rocket y si tenemos configurado el vaciado de cache de Varnish también vaciara el cache de Varnish.
- **Precargar la cache:** Es el acceso directo para precargar cache de forma manual con los ajustes que hemos configurado en la sección “Precargar”.
- **Purgar OPCache:** Una opción que tiene WP Rocket desde hace algunas versiones es que permite purgar el OPCache de PHP en caso de tenerlo activado en el intérprete PHP del servidor.
- **Exportar ajustes:** Nos permite exportar la configuración de WP Rocket para importarlo en otra instalación de WordPress.
- **Importar ajustes:** Nos permite importar una configuración de WP Rocket previamente exportada en otra instalación de WordPress.
- **Vuelta atrás:** Nos permite volver a una versión anterior de WP Rocket en caso de que existan problemas de compatibilidad con la versión actual.

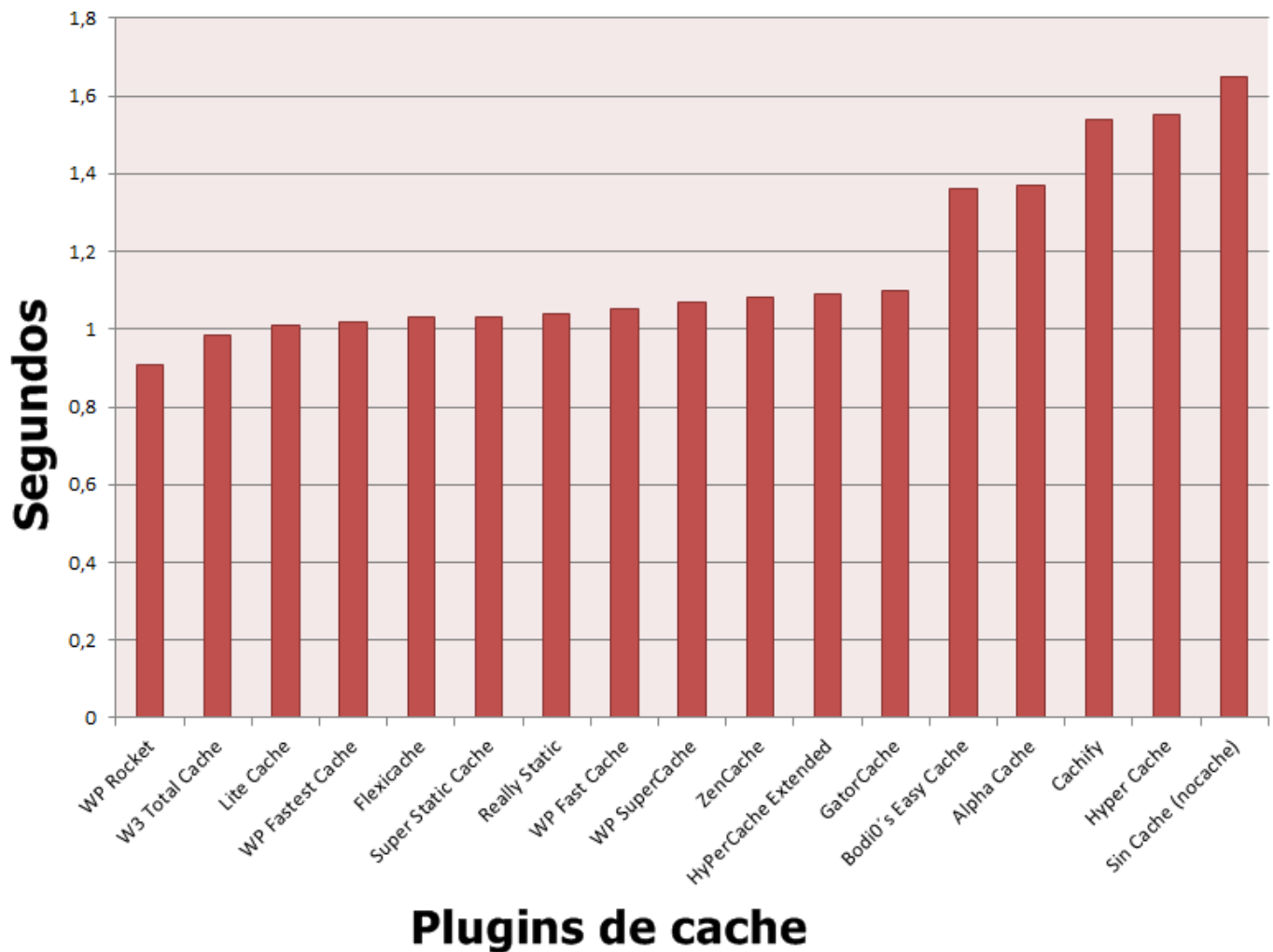
Esta es otra de las secciones que suele cambiar bastante cuando los **desarrolladores de WP Rocket** actualizan el plugin y se le van añadiendo funcionalidades.

WP ROCKET VS OTROS PLUGINS DE CACHE

Como he dicho antes y también en el video incrustado en este artículo, **WP Rocket** es uno de los **mejores plugins de cache para WordPress** en lo que a tiempos de respuesta y efectividad se refiere.

Sí que es cierto que no tienen las posibilidades de **cache distribuido**, **cache en RAM** y otras funcionalidades de **cache avanzado de base de datos** y objetos como tiene el **plugin W3 Total Cache**, pero en su segmento **WP Rocket** es un plugin bueno y eficaz.

A continuación te dejo una gráfica de un benchmark realizado por nosotros hace ya algún tiempo donde puedes ver con detalle los tiempos de respuesta de algunos **plugins de cache comparados** con los de **WP Rocket**:



De todas formas, aunque como ves en la gráfica **WP Rocket** es muy bueno, personalmente creo que cada plugin tiene sus ventajas y sus inconvenientes, depende mucho del proyecto y de las circunstancias de cada proyecto.

Nosotros normalmente recomendamos **W3 Total Cache** junto a **WP Rocket** y a **WP Fastest Cache**, depende del caso, del proyecto y de otras muchas variables.

MARCA BLANCA DE WP ROCKET

Una cosa que muy pocos saben de **WP Rocket** es que permite configurar el plugin en modo marca blanca, es decir, que no aparezca **WP Rocket** por ningún lado y que aparezca lo que nosotros queramos poner.

Para activar la marca blanca tendremos que añadir una línea al [archivo wp-config.php de WordPress](#), concretamente esta línea:

```
1 define ('WP_RWL', true);
```

Y una vez hagas eso, se añadirá una nueva pestaña llamada “Marca blanca” a la sección de **configuración de WP Rocket**:

Opciones básicas	Opciones avanzadas	CloudFlare	CDN	Varnish	Marca Blanca	Herramientas	Preguntas Frecuentes	Soporte
Nombre del plugin:	<input type="text" value="WP Rocket"/>							
URL del plugin:	<input type="text" value="http://www.wp-rocket.me"/>							
Descripción:	<input type="text" value="El mejor plugin de caching para WordPress."/>							
Autor:	<input type="text" value="WP Rocket"/>							
URL del autor:	<input type="text" value="http://www.wp-rocket.me"/>							
<input type="button" value="Restablecer los valores Marca Blanca por defecto"/>								
<i>Atención: Si cambias algo, las pestañas tutoriales + Preguntas Frecuentes + Soporte se ocultarán.</i>								

Ahora debemos rellenar todos los campos de esta sección con los datos que nosotros veamos oportunos.

Una vez los tengamos rellenados debemos eliminar la línea que hemos puesto en el **wp-config.php** o simplemente cambiar a "false" la función, para que desaparezca la pantalla "Marca blanca" de la configuración, con lo que dejaremos eso cambiado.

```
1 define ('WP_RWL', false);
```

Esta funcionalidad de **WP Rocket** es útil cuando queremos instalarle el plugin a un cliente nuestro pero no queremos que conozca **WP Rocket**, aunque le saca un poco de transparencia al asunto, puede ayudarnos en determinadas situaciones específicas.

DESACTIVAR EL CACHE DE WP ROCKET

Como he dicho al principio del artículo, justo al **activar WP Rocket** se activará el **cache de página de WP Rocket** para la [instalación de WordPress](#).

Desde la interfaz de **WP Rocket** no podemos **desactivar el cache de WP Rocket**, pero podemos hacerlo con un método oficial alternativo ofrecido por los desarrolladores de **WP Rocket**.