

Checklist – Autoload & Transients en WordPress

Acelera WordPress limpiando carga oculta

Una guía completa y segura para desarrolladores y administradores de WordPress que buscan optimizar el rendimiento mediante la limpieza del autoload y transients. Este checklist te ayudará a medir, limpiar y prevenir el bloat de manera sistemática.

Por: Fabini | [@ilfabini](#)

Checklist gratuito • Guía práctica y accionable

Antes de empezar: Seguridad primero

Backup completo

Realiza siempre un backup completo que incluya tanto archivos como base de datos antes de cualquier intervención.

No hay marcha atrás sin esto.

- Base de datos completa
- Todos los archivos del sitio
- Configuraciones del servidor

Entorno de staging

Siempre que sea posible, ejecuta estas operaciones primero en un entorno de staging idéntico a producción.

- Copia exacta de producción
- Misma versión de PHP y MySQL
- Plugins y temas idénticos

Punto de restauración

Crea un punto de restauración específico justo antes de comenzar la limpieza para poder volver rápidamente si algo falla.

📌 **Nota ética importante:** Ejecuta este checklist solo si comprendes cada paso descrito. En caso de duda, solicita ayuda a un desarrollador experimentado. La optimización incorrecta puede afectar la funcionalidad del sitio.

Qué es el autoload (explicado en 60 segundos)

El autoload es un mecanismo de WordPress donde ciertas opciones de configuración se cargan automáticamente en **cada petición** al servidor, independientemente de si se necesitan o no en esa página específica.

Cuando una opción tiene `autoload='yes'` en la tabla `wp_options`, WordPress la carga inmediatamente al inicializar, consumiendo memoria y tiempo de procesamiento innecesarios si contiene datos grandes o innecesarios.

El problema surge cuando plugins mal desarrollados o desinstalados incorrectamente dejan opciones pesadas marcadas como autoload, creando un lastre invisible que ralentiza todo el sitio.



<800KB

Objetivo orientativo

Tamaño ideal del autoload

1-2MB

Aceptable

Depende del tipo de sitio

Medir: Consultas SQL para diagnosticar

Antes de limpiar nada, necesitas medir el estado actual del autoload. Estas consultas te proporcionarán datos precisos sobre el peso total y las opciones más problemáticas.

Consulta total del autoload

Obtén el peso total en MB de todas las opciones autoload:

```
SELECT
ROUND(SUM(LENGTH(option_value))/1024/1024,
2) AS autoload_MB
FROM wp_options
WHERE autoload='yes';
```

Top 20 opciones más pesadas

Identifica las opciones que más peso aportan:

```
SELECT option_name,
ROUND(LENGTH(option_value)/1024, 1) AS KB
FROM wp_options
WHERE autoload='yes'
ORDER BY LENGTH(option_value) DESC
LIMIT 20;
```

 **Importante:** Si tu instalación no usa el prefijo estándar `wp_`, sustituye todas las referencias a `wp_options` por tu prefijo personalizado (por ejemplo: `xyz_options`).

Ejecuta estas consultas desde phpMyAdmin, Adminer o cualquier cliente MySQL y anota los resultados. Estos números serán tu línea base para comparar mejoras después de la limpieza.

Limpieza: Proceso seguro paso a paso

La limpieza debe realizarse de forma gradual y controlada. Empezamos con las acciones más seguras y progresamos hacia las que requieren más atención.

01

Transients expirados (seguro)

Elimina automáticamente todos los transients que han caducado:

```
wp transient delete --expired
```

Esta operación es completamente segura ya que solo elimina datos temporales ya vencidos.

02

Todos los transients (con precaución)

Solo si comprendes las implicaciones, puedes eliminar todos los transients:

```
wp transient delete --all
```

Advertencia: Esto puede provocar regeneración de cachés y temporalmente ralentizar el sitio.

03

Identificar opciones huérfanas

Revisa la lista "Top 20" y verifica si las opciones grandes pertenecen a plugins ya desinstalados. Busca patrones como:

- Nombres que incluyan prefijos de plugins eliminados
- Opciones con tamaños sospechosamente grandes
- Configuraciones de servicios ya no utilizados

Documenta cada acción realizada para poder revertir cambios si es necesario. La paciencia en este proceso evita problemas mayores.

Eliminar opciones huérfanas con máxima precaución

Esta es la parte más delicada del proceso. Un error aquí puede romper funcionalidades del sitio, por lo que cada paso debe ejecutarse con extrema cautela y verificación previa.

1

1. Identificar con certeza

Antes de eliminar cualquier opción, verifica que realmente pertenezca a un plugin desinstalado:

- Busca documentación del plugin original
- Confirma que el plugin ya no está presente
- Verifica que no sea una opción del tema activo

2

2. Confirmar inexistencia del sistema

Asegúrate de que la funcionalidad asociada ya no existe en el sitio:

- El plugin fue completamente desinstalado
- No hay dependencias de otros plugins
- WordPress core no utiliza esa opción

3

3. Eliminar con SQL específico

Usa esta plantilla SQL, sustituyendo `[[option_name]]` por el nombre exacto:

```
DELETE FROM wp_options
WHERE option_name='[[option_name]]'
AND autoload='yes'
LIMIT 1;
```

El `LIMIT 1` es crucial para evitar eliminaciones masivas accidentales.

Advertencia crítica: Antes de eliminar, comprueba que no existan duplicados de la opción con diferentes valores de `autoload`. Si algo falla después de la eliminación, restaura inmediatamente desde el backup.

Re-medir y validar resultados

Después de la limpieza, es fundamental medir de nuevo para cuantificar las mejoras obtenidas y asegurar que no se han introducido problemas.

Repetir mediciones SQL

Ejecuta nuevamente las consultas de la fase de medición:

- Consulta de autoload total en MB
- Top 20 opciones más pesadas
- Compara con los valores iniciales

Medir rendimiento web

Abre la homepage y una página interna representativa del sitio:

- Time to First Byte (TTFB)
- Largest Contentful Paint (LCP)
- Usa Lighthouse o GTmetrix

Una reducción significativa del autoload debería traducirse en mejoras medibles de TTFB. Si no observas mejoras, revisa si existen otros cuellos de botella como plugins pesados, temas mal optimizados o problemas de hosting.

Documenta estos resultados para futuras auditorías y para demostrar el valor de las optimizaciones realizadas.

Métrica	Antes	Después	Mejora
Autoload MB	[[anotar]]	[[anotar]]	[[calcular]]
TTFB (ms)	[[anotar]]	[[anotar]]	[[calcular]]
LCP (ms)	[[anotar]]	[[anotar]]	[[calcular]]

Prevención: Mantener el sitio limpio a largo plazo

La limpieza puntual es útil, pero la prevención sistemática es lo que mantiene un WordPress verdaderamente optimizado. Estas prácticas te ayudarán a evitar que el problema se repita.



Selección cuidadosa de plugins

Prioriza plugins que sean ligeros y que implementen correctamente la limpieza al desinstalar. Revisa las valoraciones y busca evidencias de que el desarrollador se preocupa por el rendimiento.

- Lee reseñas sobre rendimiento
- Verifica que tenga hook de desinstalación
- Evita plugins abandonados o mal mantenidos



Buenas prácticas de desarrollo

Si desarrollas funcionalidades personalizadas, evita guardar datos enormes en una sola opción con autoload activado. Usa transients o custom tables cuando sea apropiado.

- Datos grandes sin autoload
- Usar transients para cachés temporales
- Custom tables para datasets complejos



Object cache persistente

Considera implementar Redis o Memcached para reducir la dependencia del autoload y mejorar el rendimiento general de consultas repetitivas.

- Redis para sitios de medio-alto tráfico
- Memcached como alternativa ligera
- Configuración en hosting gestionado



Auditoría mensual

Programa recordatorios mensuales para ejecutar las consultas de medición y mantener el autoload controlado. La prevención constante es más eficaz que las limpiezas esporádicas.

- Recordatorio en calendario
- Registro histórico de mediciones
- Alertas automáticas si es posible

Hoja de trabajo imprimible

Esta sección proporciona tablas prácticas que puedes completar durante el proceso de optimización. Imprime esta página para tener un registro físico de tu trabajo.

Top opciones identificadas

Nombre de opción	Tamaño (KB)	Plugin origen	Acción
[[completar]]	[[completar]]	[[completar]]	✕ Mantener ✕ Eliminar
[[completar]]	[[completar]]	[[completar]]	✕ Mantener ✕ Eliminar
[[completar]]	[[completar]]	[[completar]]	✕ Mantener ✕ Eliminar
[[completar]]	[[completar]]	[[completar]]	✕ Mantener ✕ Eliminar

Checklist de acciones

- ✕ Backup completo realizado
- ✕ Entorno staging preparado
- ✕ Medición inicial completada
- ✕ Transients expirados eliminados
- ✕ Opciones huérfanas identificadas
- ✕ Opciones problemáticas eliminadas
- ✕ Medición final realizada
- ✕ Rendimiento web validado

Próximo recordatorio

Fecha programada: [[completar]]

Responsable: [[completar]]

Notas especiales:

[[espacio para anotaciones]]

Errores comunes y soluciones + Conclusión

1

Borrar sin backup

Problema: Se eliminaron opciones importantes sin respaldo.

Solución: Restaurar inmediatamente desde el último backup completo. Por esto insistimos tanto en la preparación inicial.

2

Prefijo de tabla incorrecto

Problema: Las consultas SQL no devuelven resultados.

Solución: Verificar el prefijo real de las tablas y sustituir `wp_` por el prefijo correcto en todas las consultas.

3

Eliminar opciones activas

Problema: Se rompió funcionalidad del sitio tras eliminar opciones.

Solución: Verificar siempre que el plugin o tema asociado realmente no existe antes de eliminar cualquier opción.

4

No medir resultados

Problema: No se puede cuantificar la mejora obtenida.

Solución: Registrar siempre métricas antes y después para demostrar el valor del trabajo realizado.

Mide → Limpia → Prevé

Este proceso de optimización del autoload y transients debe convertirse en parte de tu rutina de mantenimiento regular de WordPress. Los sitios más rápidos no son casualidad, sino el resultado de atención constante a los detalles técnicos que realmente impactan en el rendimiento.

Recuerda que cada sitio es único y puede requerir ajustes específicos a este checklist general. La clave está en comprender los principios subyacentes y aplicarlos con criterio técnico sólido.

Más plantillas en fabini.one

Etiqueta [@ilfabini](https://twitter.com/ilfabini) si te sirvió

Este documento se proporciona "tal como está" sin garantías. El uso de estas instrucciones es bajo tu propia responsabilidad. Siempre mantén backups actualizados y considera consultar con profesionales para implementaciones en sitios críticos.