

# Nginx versus Apache, falacias de ayer y de hoy

## ? ¿"Apache para contenido dinámico y Nginx para estático"? Una aclaración técnica

Existe una generalización bastante extendida, que considero una falacia que muchas veces se usa para no avanzar en el uso de nginx:

“ Apache para contenido dinámico, Nginx para contenido estático.

A partir de ahí, muchas veces se concluye que:

- Si una aplicación tiene procesos dinámicos (como sincronización, API, backend), Apache es mejor.
- Si es solo para servir archivos estáticos, entonces Nginx.

Sin embargo, esta conclusión **no es técnicamente precisa**, y puede llevar a decisiones mal fundamentadas.

## ? ¿Qué hace realmente una aplicación web moderna?

Una aplicación típica puede incluir:

- Archivos estáticos: HTML, CSS, JS, imágenes
- Código backend dinámico (PHP, Python, Node.js...)
- APIs REST o GraphQL
- Procesos de subida, descarga y manipulación de archivos
- Tareas en segundo plano (cron, workers, colas)

Todo esto hace que la clasificación "dinámico vs estático" no sea suficiente para determinar qué servidor es más adecuado.

# ? Comparativa técnica entre Apache y Nginx

## Nginx

- ✗ Muy eficiente para servir archivos estáticos.
- ☐ Manejo asíncrono y event-driven: excelente bajo carga.
- ☐ Usa PHP-FPM para ejecutar código dinámico, separando claramente funciones.
- ☐ No soporta `.htaccess` (toda la configuración va en los archivos del servidor). Eso puede ser un handicap para algunos, pero para otros es justo lo contrario. Es **seguridad, estabilidad y eficacia**

## Apache

- ☐ Servidor muy compatible y ampliamente soportado.
- ☐ Soporta `.htaccess`, útil en entornos compartidos.
- ☐☐ Más pesado en escenarios de alta concurrencia (según el MPM usado).
- ☐ También puede usar PHP-FPM o `mod_php` según configuración.

# ? ¿Cuál es mejor?

Depende del caso:

- **Nginx** destaca por su rendimiento, menor consumo de recursos y eficiencia en entornos concurrentes.
- **Apache** es más flexible en cuanto a configuración por usuario, y puede ser más cómodo en ciertos entornos como hosting compartido y/o sysadmin, un poco vagones o acomodados.
- Ambos pueden manejar contenido estático y dinámico sin problemas cuando están bien configurados.

# ? Conclusión

Elegir entre Apache y Nginx **no debería basarse únicamente** en si una aplicación sirve "contenido estático" o "dinámico".

Lo correcto es considerar:

- Arquitectura de tu aplicación
- Escalabilidad esperada
- Modelo de despliegue
- Nivel de personalización y control que necesitas
- Consumo de recursos

Ambos servidores son válidos y potentes. La elección óptima no es cuestión de gustos, sino de contexto técnico. ☐

## Aviso

Esta documentación y su contenido, no implica que funcione en tu caso o determinados casos. También implica que tienes conocimientos sobre lo que trata, y que en cualquier caso tienes copias de seguridad. El contenido el contenido se entrega, tal y como está, sin que ello implique ningún obligación ni responsabilidad por parte de [Castris](#)

Si necesitas soporte profesional puedes contratar con Castris [soporte profesional](#).

---

Revision #1

Created 2025-05-29 09:55:25 UTC by Abkrim

Updated 2025-05-29 09:59:08 UTC by Abkrim