# Hosting WordPress :house:

Any mid-to-high performance WordPress host will work great. You may also have success on low performance WordPress hosts for smaller or medium sized sites, but a better server is recommended.

A great Gatsby/WordPress workflow is to use Local by Flywheel for hosting your local WordPress server and then pushing from there to Flywheel or WPEngine hosting for your live/staging WordPress servers.

## Host IP rate limiting

If you have a very large site (with a lot of images), you may need to ask your hosting provider to remove IP rate limiting for your IP addresses as this source plugin will try to fetch them as fast as it can to speed up your build times.

## Server timeouts & Over-resourced server errors

If your server becomes overwhelmed by the number of requests being made from Gatsby to WP during an uncached build, your build may fail with an error explaining that your server is overloaded. You can fix this one of 2 ways:

### Solution 1: Get a better server (for faster build times)

If you're getting errors about your server being overloaded, you can try renting and migrating to a server with more resources. This will cost more but will mean your builds are faster and more reliable. For many sites WPEngine will work well. If WPE doesn't work for you or you need more resources at a cheaper price you'll need to do some investigation to find a suitable hosting provider. Once you have a good server you may be able to speed up your build times by increasing the GraphQL request concurrency (data requests) and media file request concurrency (requests for images, videos, pdf's, etc) settings from their defaults.

### Solution 2: Tweak Gatsby concurrency settings to suit a less powerful server (for slower build times)

By lowering the plugin options for request concurrency (the number of requests made at one time) you can fix over-resourced server errors. There's a setting for GraphQL request concurrency (data requests) as well as for media file request concurrency (requests for images, videos, pdf's, etc). Some servers may require these settings to be set as low as 1, 5 or 10. Try a few different settings to determine how many concurrent requests your server can handle. The higher this number can be set, the faster your uncached builds will become.

# Up Next :point_right:

- :athletic_shoe: Themes, Starters, and Examples
- :medal_sports: Usage with popular WPGraphQL extensions
- :hammer_and_wrench: Debugging and troubleshooting
- :national_park: Community and Support
- :point_left: Back to README.md