Third Party search engines in self-hosted WordPress

WordPress doesn't reaally have a search engine built into the CMS. This may cause unexpected results and it may skew a user's search results.

The first thing to look at is replacing WordPress built in search with something more robust and that works as a real search engine.

Elasticsearch

I've been looking at Elasticsearch for a while and I think it's time to see what it would take to make it work in production.

First we'll test Elasticsearch locally with a local installation of WordPress using the elasticpress plugin and an Installation of Elasticsearch using Docker.

Then we'll explore the lowest cost alternative for deploying Elasticsearch in production looking at different cloud providers before integrating it with the same plugin on our production site.

Elasticsearch: Testing and Development

Rather than install it locally and have to deal with all the files and configuration, I chose to install Elasticsearch as a Docker image (pick the self-managed option from the guide).

Once we have Elasticsearch running on Docker, we need to install a plugin. I chose elasticpress to test with.

> **Warning**
>
> Elasticpress has one big drawback. It only works with Elasticsearch 7.9 and older. The current version is not supported so it might or might not work (it has worked so far).

For testing the plugin, I'll concentrate on the configuration and weighing of terms for different post types.

Self hosted configuration is just adding the URL to the Elasticsearch server, since we're running everything on the same machine it's just pointing to the Docker Elasticsearch instance

Figure 1: Elastic Press Configuration Dialogue

Adding more weight to an item will mean it will have more presence during searches. For example, adding more weight to the content attribute will cause search matches on the post content to appear more prominently.

For my blog I want to prioritize the content over the other searchable fields so I've assigned it a higher weight than the other available fields.

I don't use pages on my blog so I'm OK with leaving all weights for pages equal. This may change when the site uses pages

Figure 2: Elasticpress weight configuration for WordPress posts and pages

I have two custom post types in my development blog. The we can search the books CPT by content or taxonomy. The glossary CPT doesn't have a taxonomy so we can only weight the content.

Figure 3:

Elasticpress
weight
configuration
for
WordPress
custom post
types

Elasticsearch: Production

I am happy with the results of the development site using Elasticsearch with WordPress but running the plugin in production means that you have to choose a cloud service provider to run the application from. Elastic, the company behind Elasticsearch, offers their own cloud service; you can also choose your favorite cloud provider to run the code from.

I chose to go with Elastic Search hosting and their 14-day free trial to make sure that everything works as intended.

I hit the first issue with the plugin in production when trying to configure it. It appears that the only two options available are to use their paid service ($16 per month on the cheapest tier) or run your own installation of Elasticsearch, there doesn't appear to be any other options.

It appears that for production sites the best option would be to use Elasticpress and the [Elastic.io](#) hosting service but the cost is USD $79 a month so we're back to trying to figure out how to configure the plugin (or any otger plugin) for an Elasticsearch instance hosted on a different server where we have more control (Elastic Cloud costs USD $0.36 per hour running on GCP)

So, as much as I like Elastic Search, it remains an interesting choice for development and, maybe, for larger scale sites than my blog, so let's look at another alternative.

Algolia

Algolia is a managed search service. Third party plugins enable WordPress to use Algolia as the site's search engine.

Unlike Elasticsearh, Algolia is configured mostly on the server. The only thing we configure on the WordPress plugin is the information about the server, you get the information from your Algolia server configuration.

Figure 4: Algolia Search WordPress plugin configuration

The main advantage (or disadvantage) that I see is that there's only one set of configuration tools to set up. However, I miss the weight settings in Elasticsearch and wish there was an equivalent functionality for Algolia

The other thing that I found interesting is the automatic highlight of the term you searched for.

For evaluation purposes, I chose Algolia's free plan. It allows for ten thousand queries per month, which I believe will be more than enough for the traffic I get for the blog. If it goes above that, I can always switch plans later :)