

---

## Migrating to Statamic

Abhishek Tiwari 

Citation: *A. Tiwari*, "Migrating to Statamic", Abhishek Tiwari, 2014.

[doi:10.59350/8gkjy-n8y18](https://doi.org/10.59350/8gkjy-n8y18)

Published on: February 24, 2014

Migration of this blog from Jekyll to Statamic is now complete. To my surprise migration process was very smooth and quick (it took less than 5 hours). As part of migration I also ported design of existing Jekyll website to Statamic. In addition, actual content migration was quite straightforward.

For those who are unfamiliar with Statamic, it is *a new kind of flat-file CMS built on PHP*. Just like other flat-file CMS systems such as Jekyll and Hyde, Statamic is 100% version controllable.



**Figure 1:** Statamic, a dynamic flat-file CMS built on PHP

## Static or Dynamic

In Statamic,

- Everything including content, cookies, users, groups , etc., are stored in flat files.
- YAML files are used extensively for configuration.
- A lightweight & responsive control panel makes publishing easier.
- Pages are rendered dynamically without using any database.
- One can easily mix static elements with dynamic.

## Migrating existing Jekyll blog

I can summarise the whole migration as two step process,

1. Porting the existing Jekyll theme to Statamic, and
2. Migrating existing content

Porting the existing Jekyll theme to Statamic requires restructuring and refactoring of Jekyll templates. This may include,

- Moving Jekyll `_includes` in Statamic `partials` folder.
- Refactoring the Jekyll `_layouts` into Statamic `layouts` and `templates`.
- Moving `_config.yml` settings in `theme.yml`.
- Changing template block syntax `()`.
- Changing the includes to partials helper syntax ( `include footer.html` to `theme: partial src="footer"`).
- Changing the conditional syntax from Jekyll to Statamic.
- Porting the for loop syntax from Jekyll to Statamic's tag pairs, list helpers and partial with `use_context`.
- Changing the Liquid tags and filters used in Jekyll template to variable modifiers provided by Statamic.
- Changing Jekyll normal Link CSS/ Script Source HTML tags to Statamic theme helpers.
- Creating templates with names `tags.html`, `categories.html` or `taxonomies.html` for tags and categories pages rendering.

Migrating existing content was actually quite easy, just needs copying the markdown and image files from Jekyll to Statamic content folder.

## First Impression

Although I am a big fan of Jekyll, on this occasion I decided to go with Statamic. This was mainly driven by ease of publishing using Statamic control panel. Statamic control panel provides ability to manage content anytime anywhere on any device. Now all I need is a browser with internet connection. With Jekyll I was limited by publishing workflow which requires more than a browser.

Again, as a static-site generator Jekyll has its own advantages mainly around performance, availability and maintenance. With Jekyll I had less things to worry as site was served by Amazon S3. For Statamic I am running a virtual server and I hoping I will spend more time in writing than managing server.

Having worked with variety of CMS/WCM/ECM systems, I must admit that Statamic is a breath of fresh air. Fieldtypes are interesting concept, and I love the way *Statamic promotes convention and configuration*. Using appropriate YAML files such `folder.yml`, `fields.yml`, `routes.yml`, `theme.yml` etc., you can configure and customise the behaviour of Statamic according to your needs.

**What's Next**

Firstly, a lot of optimisation is required to improve current page load time, which is bit skewed due to heavy static assets such as fonts and images. Secondly, some automation is required to auto-commit and push the content change on live server to GIT repo.

Happy blogging again.