

I'm currently working on automating our deployments and specifically automatically generating configuration files for different targets.

I tried using Puppet for that (I mean, it's a **configuration** management system), but failed, because the kind of configuration files we're dealing with here is way too complex for it. I was basically searching for a framework, that renders multiple files from one directory into another directory, copying static files and building up dynamic files out of snippets.

Take this directory layout:

```
* /root [A]
|
*--* document.xml (XML-file with snippet-definitions) [A1]
*--* static.json (Static file) [A2]
|
*--* /Very unspecific [B]
|
*--* snippet1.xml [B1]
*--* snippet2.xml [B2]
|
*--* /More specific [C]
|
*--* snippet1.xml [C1]
|
*--* /Even more specific [D]
| |
| *--* snippet2.xml [D1]
|
*--* /Even more specific [E]
|
*--* snippet1.xml [E1]
*--* snippet2.xml [E2]
*--* static.json [E3]
```

So, when I fire up my generator, I give it a node and it generates my output directory.

For example, if I'd use node B, I get the static file A2 and the dynamic file A1 with snippets B1 and B2 included. If I'd use node C instead, I get the static file A2 and the dynamic file A1 with snippets **C1** and B2, because C1 is defined under C and there's no snippet2.xml there.

Do you get, where I'm going here?

However, I couldn't find a „template engine“ (or whatever you might call it), that can do things like that.

So, I wrote my own. Introducing **SOCKO!**, a „**Hierarchical file weaver**“, that does just that. It's a node module, that you can get through the **official NPM registry**. There's also already a **grunt-plugin** for you to use.

SOCKO! copies static files from the hierarchy and bundles dynamic files (called „sockets“, hence the name) with snippets (called „cartridges“). There are even three different inclusion tags for the socket-files available, so your IDE won't complain.

Check it out.