

## Visualising wikis

I've been doing some website updates recently, as part of the CrisisCommons work. My father taught me to always clean and examine something carefully before you take it apart, and this works as well for code and sites as it does for cars and houses, so I've been carefully analysing each site against a set of intended (and frequency-weighted) user journeys. And what would be a really nice thing to have would be a tool that generated a semantic network of a wikisite so I could trace its hub nodes and get an easy visual representation of how much each node and link is used (colour-coding seemed obvious here).

Now I remember the small worlds (everything is just 6 steps from everything else if you know which 6 steps to take) and semantic network theories from uni, and I've knocked up a few labelled graphs myself in my time, and I know there's some great graph generation freeware out there, so I thought "this has got to be a standard item in the open source community, surely".

Er. No. But there are some good things out there already.

\* Aharef's visualising websites as graphs. (example). Verry close to what I'm looking for, but runs off weblinks and doesn't tag the hub nodes.

\* [Flexplorer](#). Great tool for mapping all the websites you're pointing at. Not so great for mapping just the one wikisite. Does do good labels.

\* [Wikmindmap](#). Does wikis. Does labels. Doesn't do the whole of a wiki. More of an explorer's torch (sees ashort way but very well) than an explorer's map (sees everything but in less detail).

\* [Powermapper](#) - too literal. You get a (non-graphical) representation of the site contents rather than a summary that you can infer metainformation from.

So close, but no cigars. There is however a plan. Aharef has published his/her/their mapping code, so I'm going to see if I can change it to pull out the [[]] tags, and add a name to every heavily-linked node. It's a plan. We'll see how the reality pans out. But right now, I'm having a lovely time playing with the [Processing](#) visualisation tool and working out whether hacking Flexplorer is a better option. And it seems like [I'm not the only person thinking about doing this](#).