chem-bla-ics

XHTML+RDFa: chemical examples

Egon Willighagen

Published August 10, 2010

Citation
Willighagen, E. (n.d.). In *chem-bla-ics*. chem-bla-ics. https://doi.org/10.59350/kkvt7-zpm64

Keywords
Html, Rdf, Rdfa

Steffen asked me if I could also provide a few examples on how to actually put RDF triples in the HTML, as the template I gave yesterday is a mere empty canvas to draw the triples on. There are actually various examples in my blog, which I will summarize here.

Before I start, I like to put some emphasize on the following RDFa pattern. An RDF resource that serves as subject is always mapped to a HTML element. This can be a div element, but also other elements, as we will see in the example.

A molecule SMILES

The oldest RDFa example in my blog is from

1. That was almost two years before the final Recommendation, and is not quite accurate anymore. But here's the correct version:

This example shows how to embed the SMILES string CCO semantically. This example shows that the outer most span element is used to define the subject of the RDF triple, using the @about attribute to define the URI of the resource: *#ethanol*. Note that this URI is relative to the URI of the HTML page in which it is embedded. Later we will see an example with a full URI.

But I don't want to hack HTML!

Yeah, fair point. Just make a point with your publisher when you submit a new paper. It is the duty of the publisher and your software vendor to do this right. In 2008 I wrote a small Ubiquity script to automagically convert an InChI into semantified HTML content. But I am not sure this script still works. If interesting, let me know, and I will revive the Firefox thingy.

And why would I want to do it anyway??

Because software can more easily understand what you mean. This is why Google is now pushing rich snippets. Chemical blogspace understands them too, allowing you to see blog posts about molecules on other webpages. With a simple bit of JavaScript you can link from your webpages, you can enrich your HTML sites with semantic chemistry yourself. Bioclipse also has no problem with extracting the RDF from HTML. Even Firefox can understand it. Really, there is no end to it.

Of course, why you should do this comes basically down to Molecular Chemometrics Principle #2, but I have not written that on up yet (see also McPrinciple #1).

Reporting problems with molecular representations

More recently, I reported about using RDFa in human readable log file for computations I am doing (see Scripts logs as HTML+RDFa: mix free text reporting with CSV). That code looks like:

chem-bla-ics

This example uses a *div* element to host the subject resource. Again, the resource URI is relative to the URI of the document, e.g. this one. We can also note a new attribute, *@typeof*, which is here used to define the rdf:type of the *#200234* resource.

This code snippet does not define the *um* namespace, which was done elsewhere in the HTML. Moreover, this code snippet does not actually reuse existing ontologies, which is highly recommended. The upcoming RDF symposium in Boston will tell you all about chemical ontologies in the RDF world (see this detailed program, which itself is HTML+RDFa!). But, if you would just overlook the ad hoc namespaces used, you might appreciate the nesting: besides the compound (*#200234*), a second resource is defined (#error0). In total, this example contains six triples.

Meanwhile, the output simply looks like:

CID 200234: Ti1

A molecule table

This third, and for now last, example shows several other features. This HTML snippet show a one entry molecule table, very much like those molecular spreadsheets in Excel, but than right here in your webbrowser. (Can you imagine what happens if we mash this up with JavaScript molecular viewers

First of all, the rdf.openmolecules.net project is used to construct an absolute URI for the molecule. The table then gives some properties of the molecule: its name (using Dublin Core, though perhaps rdfs:label is better), the boiling point (nicely encoded as *t0* in this 1947 paper), two cheminformatics descriptors, and the SMILES, using the same approach as the first example in this post.

The output of this table looks like:

n-Butane -0.5 10 1 CCCC

I will shortly blog about the source of the above code snippet, but you are invited to go ahead and checkout my GitHub activity (RSS).

Steffen, I think these examples should get you pretty far, but please let me know if you have further questions!