

CiteULike CiTO Use Case #1: Wordles

Egon Willighagen 

Published October 31, 2010

Citation

Willighagen, E. (2010). CiteULike CiTO Use Case #1: Wordles. In *chem-bla-ics*. chem-bla-ics. <https://doi.org/10.59350/npbqm-gfa49>

Keywords

Cito, Citeulike, Cdk, Wordle

Abstract

Last month I reported a few things I missed in CiteULike. One of them was support for CiTO (see doi:10.1186/2041-1480-1-S1-S6), a great Citation Typing Ontology.

Copyright

Copyright © Egon Willighagen 2010. Distributed under the terms of the [Creative Commons Attribution 4.0 International License](#), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

chem-bla-ics

Last month I reported a [few things I missed](#) in CiteULike. One of them was support for CiTO (see doi:[10.1186/2041-1480-1-S1-S6](https://doi.org/10.1186/2041-1480-1-S1-S6)), a great Citation Typing Ontology.

I promised the CiTO author, [David](#), my use cases, but have been horribly busy in the past few weeks with my new position, wrapping up my past position, and thinking on my position after Cambridge. But finally, here it is. Based on source code I [wrote and released earlier](#), the first use case I represent is the [Wordle](#) one, which I [showed with manual work in February](#).

Now that all the data is semantically marked up in CiteULike, I can easily extract all paper titles (or whatever is available in CiteULike) for all papers that cite the first [CDK](#) paper (doi:[10.1021/ci025584y](https://doi.org/10.1021/ci025584y)). Using the JSON interface, I have [this Groovy script](#) to extract all titles:

```
import groovyx.net.http.HTTPBuilder
import groovyx.net.http.Method
import static groovyx.net.http.ContentType.JSON

culUrl = "http://www.citeulike.org/";

citotags = [
    "cito--cites",
    "cito--usesMethodIn",
    "cito--discusses",
    "cito--extends"
    // there are more, but these are all
    // I use right now
]

papers = [
    "1073448",
    "423382"
]

http = new HTTPBuilder(culUrl)

papers.each { paper ->
    println "# Processing $paper..."
    citotags.each { tag ->
        citation = "$tag--$paper".toLowerCase()
        http.request(Method.valueOf("GET"), JSON) {
            uri.path = "/json/user/egonw/tag/$citation"

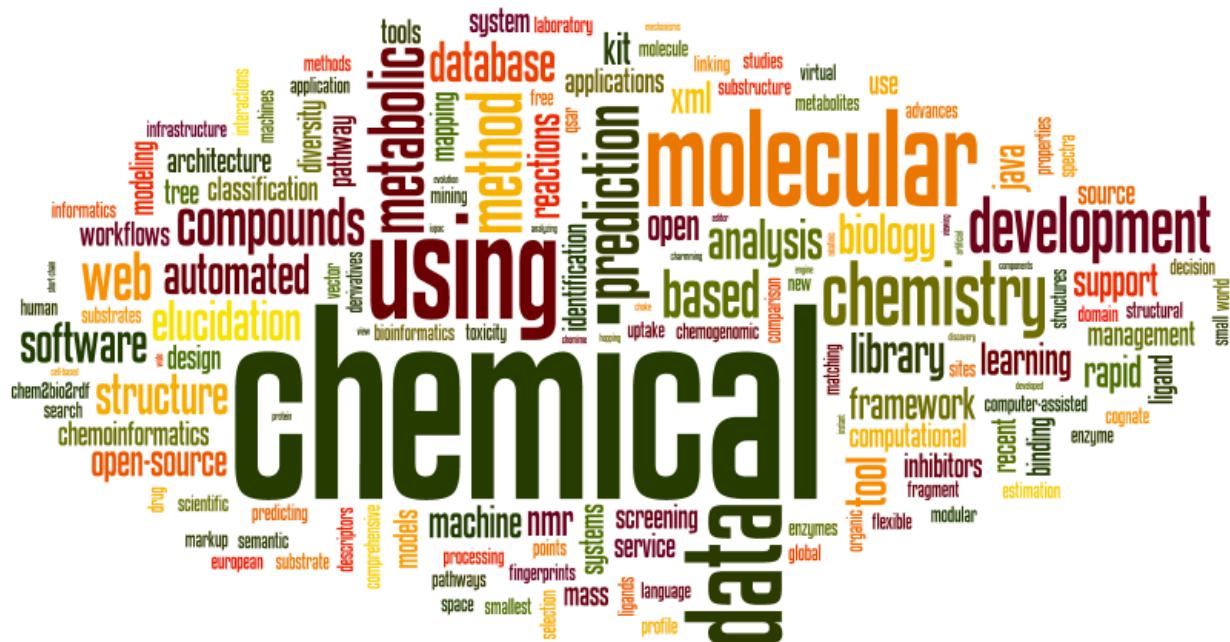
            response.success = { resp, json ->
                json.each { article ->
                    tripleCount = 0;
```

chem-bla-ics

```
        article.tags.each { artTag ->
            if (artTag.startsWith(tag)) tripleCount++
        }
        if (tripleCount > 0) {
            title = article.title
            title = title.replaceAll("\\{", "")
            title = title.replaceAll("\\}", "")
            println "$title"
        }
    }
}
}
}
```

The output is two blocks which I can easily copy/paste into Wordle. Now, I think I heard one can actually download the java code, so I am tempted to integrate it later, but for now copy/paste will do fine, after the data handling is mostly automated: with a few lines extra I can make such visualizations for any paper I annotated in CiteULike with CiTO.

The CDK I paper



The CDK II paper



Interesting differences... more statistics will soon follow. See [Further statistics on the papers citing the CDK](#) for the kind of analyses I have in mind.