chem-bla-ics

CDK Module dependencies #2

Egon Willighagen

Published March 23, 2008

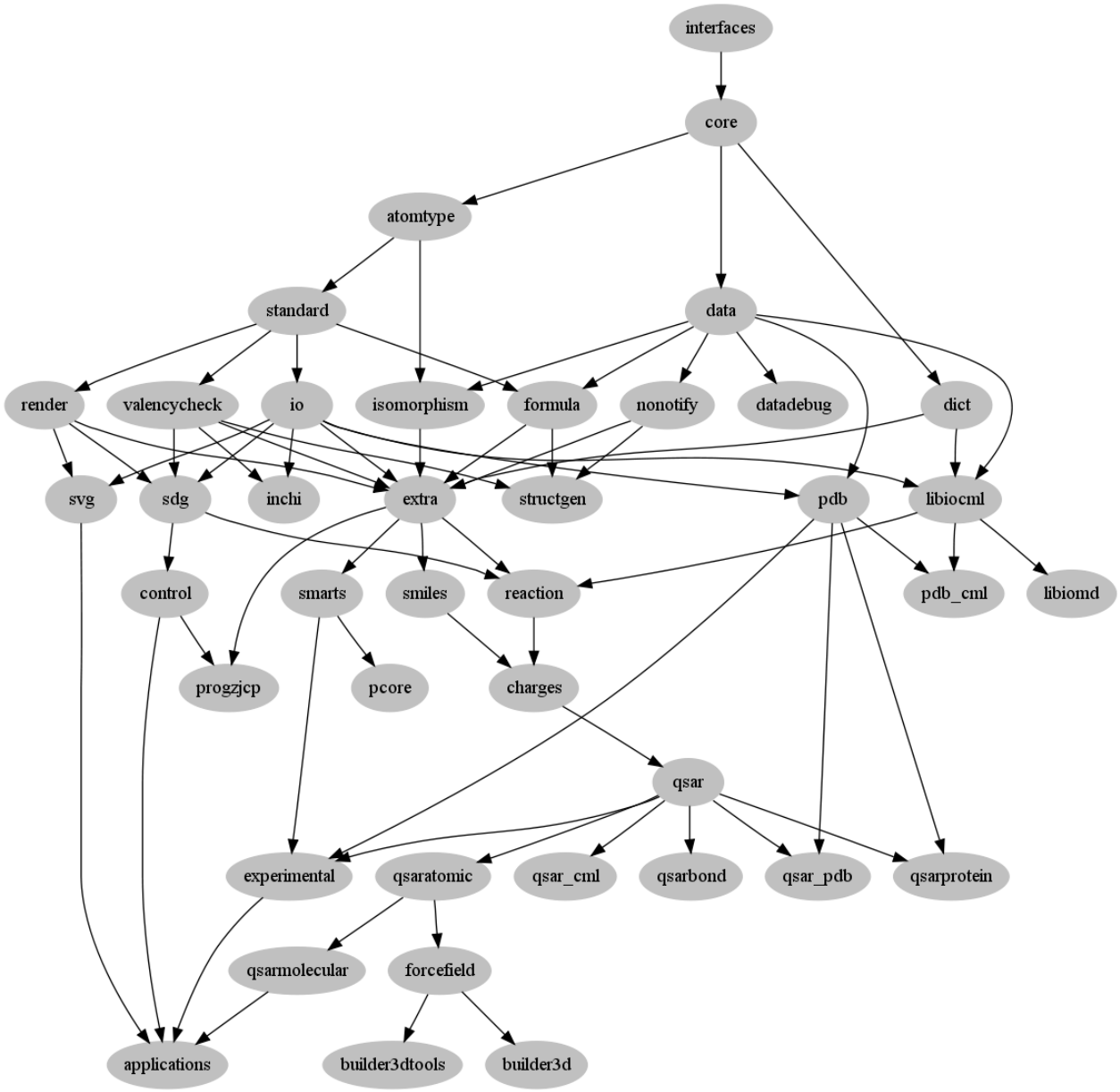
Citation
Willighagen, E. (2008). CDK Module dependencies #2. In *chem-bla-ics*. chem-bla-ics. https://doi.org/10.59350/e7qrk-ypg78

Keywords
Cdk

Abstract
A bit over 2 years ago I published a UML diagram showing the dependencies between CDK modules . Since then I lot of new modules have been defined, added or factored out from the extra module (click to zoom):

chem-bla-ics

A bit over 2 years ago I published a UML diagram showing the dependencies between CDK modules . Since then I lot of new modules have been defined, added or factored out from the extra module (click to zoom):



These kind of diagrams help us maintain the library, and apply some design goals, as explained in the first post on this.

If one compares the two diagrams, one sees that fewer code depends on the `data` module, but it is also clear that still a lot of them do. Another issue that had not properly addressed yet, is that a lot of modules still depend on the `extra` module, which aggregates everything that had not been assigned elsewhere.

Parallelism

This diagram also helped me use the [Ant task](http://ant.apache.org/manual/CoreTasks/parallel.html) to allow compiling CDK modules in parallel, instead of sequentially. Multicore machines can take advantage of that, and reduce the overall computation time. Full parallelism is not possible, and it is well visualized by the above diagram that there basically 12 sequential compilation steps in which one or more modules can be compiled. Further clean up of the module dependencies, will reduce this number, and further reduce the computation time on multicore machines.

Now, graph analysis could pinpoint the most troublesome nodes, but it would not surprise me that extra would be amongst them. But the following items are worth looking at too:

- why does `qsar` have to depend on `charges`?
- why does `sdg` (the 2D layout code) depends on `io` code?
- can `isomorphism` and `formula` be made independent of `data`?
- why does `reaction` depend on `sdg`?
- why does `forcefield` depend on `qsaratomic`?

Some of these issues are rather practical, but it is these kind of analyses that help us clean up the CDK library.