chem-bla-ics

UML diagram of CDK module dependencies

Egon Willighagen

Published December 6, 2005

Citation
Willighagen, E. (n.d.). In *chem-bla-ics*. chem-bla-ics. https://doi.org/10.59350/a3r1n-72841

Keywords
Cdk, Uml

Copyright
Copyright © Egon Willighagen 2005. 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.

The code clean up after CDK's interfaces transition is in progress, and two CDK modules are now independent of the *data* module. After doing the *core* module , the standard was next, and I finished this yesterday. The dependencies in CVS now look like (click it to get a larger view):

IMAGE LOST

This UML diagram was made with Umbrello, and the source is in XMI in CVS.

I cannot stress enough the advantages of these changes:

1. the code is cleaner
2. module dependencies are cleaner
3. impossible to use methods outside the interface
4. the algorithms are independent of the data classes

The last advantage is really important: it allows alternative implementations of the data classes. For example, we could make debug data classes, which, unlike the normal classes, do all sorts of checks when using methods of these classes. For example, they can explicitely check that parameters are not null, of the right class, and generally make sense. This makes them, possibly, slower, but also more type save, and as such great for debugging and development sessions.

Another important application of making the CDK library independent of the data classes (and only depending on the interfaces), is that we can have data classes shared with other Java libraries, such as JOElib, Octet, CML (Jumbo 5.0 is out!), and even proprietary libraries. This approach is already used in the CDK-Taverna library, and I anticipate much wider use with the arrival of Bioclipse.