

CDK - The Documentation

Egon Willighagen 

Published April 5, 2009

Citation

Willighagen, E. (2009). CDK - The Documentation. In *chem-bla-ics*. chem-bla-ics. <https://doi.org/10.59350/rzwcr-mq411>

Keywords

Cdk, Cdkbook

Copyright

Copyright © Egon Willighagen 2009. Distributed under the terms of the [Creative Commons Attribution 4.0 International License](https://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

In preparation for the [CDK workshop](#) later this month, I am writing up my material for my kick-off presentation of the workshop. So, I better make it good. Using LaTeX at least overcomes my laziness which always made Word documents look stupid. Even default LaTeX looks good:

edges between two atoms being the nodes.

4.1 IAtom

The CDK interface `IAtom` is the underlying data structure. It is fairly easy:

```
IAtom atom = new Atom("C");
```

Or, alternatively, using the *Elements* class:

```
IAtom atom = new Atom(Elements.CARBON);
```

Clearly, any such documentation becomes quickly outdated, in particular when source code fragments are involved. Yes, [CDK 1.2](#) is API stable, but only for the core classes. Moreover, I hope that the documentation will survive CDK 1.4 or 2.0 or whatever the next stable version is.

Therefore, I need to source code fragments compilable. R has the magnificent [Sweave](#), and I wanted for a long time something similar. While I do not have something that powerful yet, at least my current set up allows me have code that both compiles and embeds in the LaTeX. The system allows me to write both Java application code as [BeanShell](#) scripts. No clue yet what I will use in the workshop, maybe even both. Like Sweave, it even saves output, and I can include that in the LaTeX source too. The code fragments can either go in as a verbatim section, or as a listing, depending on what I find more appropriate.