chem-bla-ics

Atom typing in the CDK

Egon Willighagen

Published July 1, 2007

Citation
Willighagen, E. (2007, July 1). Atom typing in the CDK. *Chem-bla-ics*. https://doi.org/10.59350/wm7aq-eqz10

Keywords
Cdk

Atom typing is one of principal activities in chemoinformatics. Atom types provide additional information that cannot be derived from the connection table that is being used, or may define what force fields terms should be used. This makes perception of atom types very important.

The CDK has a few places where atom types are perceived. The HydrogenAdder and ValencyChecker are two examples. Getting the perception wrong, makes it impossible to correctly add hydrogens (of course, hydrogen should always be explicit!) For a long time, these perception algorithms have been embedded in the classes that used them, but efforts have been undertaken to refactor the algorithms into separate classes. These can be found in the package cdk/atomtype/.

Different applications, different scheme

Now, the CDK can be a bit confusing with respect to the HydrogenAdder and IValencyChecker. Originally, the CDK had only one atom type list, the StructGen Atom Types. This list was used by the deterministic structure generator (and still is), and only defined atom types for neutral atoms, and does not know anything about hybridization states.

The first bug reports dropped in when people applied the HydrogenAdder to charged molecules. However, as said, charged atoms were not defined and the algorithm failed, not silently, just gave the wrong answer. Therefore, the Valency Atom Types list was setup, which does include charged atoms. Everyone happy again.

Later, bugs were reported about the SMILES parser, which comes with additional problems: bond orders are not explicit, and have to be deduced from the connectivity; atom type perception is the only way to decide how many bonds an atom should have, and with what bond order. However, SMILES defines hybridization states, and the CDK did not have an atom type list with hybridization information. So, while the Valency Atom Types list was extended from the StructGen Atom Type List, a new list was created extending from the Valency Atom Type list: the Hybridization Atom Types list.

Since then, applications asked for other atom type lists, such as the MM2, MMFF94, PDB, and Sybyl atom types. The first two are used for the force field code in the CDK, while the latter two are used for the respective IChemObjectReaders.

JUnit testing the perceivers

Not all applications actually already make use of the new atom type perception classes in cdk.atomtype. It is wished that these well tested before the replace code in the classes that use those atom types. Therefore, Rajarshi and me have been working on JUnit test suites. The latest step in this process was that I transformed the test classes to extend a new JUnit4-based AbstractAtomTypeTest class. New in this class is that it report which atom types in the atom type list have been tested, and the test will fail if not all atom types are tested. The StructGen

chem-bla-ics

Atom Types list is mostly covered now, but for all other lists tests still have to be written (monitor the progress on CDK Nightly).

For the MOL2 atom type list, there is no Java implementation of the IAtomTypeMatcher, but we have Fortran code that can be ported (provided by Martin Ott). Anyone interested?