Finding differences between IChemObjects

Egon Willighagen ⬤

Published June 1, 2008

Citation

Keywords

Cdk

Abstract

CDK trunk is getting into shape, thanx to the many people who contribute to this, and special thanx to Miguel for cleaning up his code related to charge, resonance, and ionization potential calculations!

Copyright

chem-bla-ics

[CDK trunk](#) is getting into shape, thanx to the many people who contribute to this, and special thanx to Miguel for cleaning up his code related to charge, resonance, and ionization potential calculations!

At the moment, I am focusing at two issues:

1. QSAR descriptors that change the input (causing other descriptors to randomly fail)
2. Cloning of IChemObject (for which last week a rather serious bug was found)

Until some days ago, the CDK had one main method to introspect [IChemObject](#)s: their `toString()` results. However, finding the difference between `IChemObjects` using this approach is not trivial, particularly if there are several differences.

So, I started a new module called diff. If two objects are identical, it returns a zero-length `String`. If not, it lists the changes between the two classes, in a way much like that of the IChemObjects `toString()` methods.

For example, consider this bit of code:

```
IChemObject atom1 = new ChemObject();
IChemObject atom2 = new ChemObject();
atom2.setFlag(CDKConstants.ISAROMATIC, true);
String result = ChemObjectDiff.diff( atom1, atom2 );
```

The result value then looks like:

```
ChemObjectDiff(, flag5:F/T)
```

Now, output will likely change a bit over time. But at least, I now have a easier to use approach for debugging and writing unit tests. Don't be suprised to see `test-*` modules start depending on the new `diff` module.