chem-bla-ics

Test results for the CDK 1.0.x branch

Published December 19, 2007

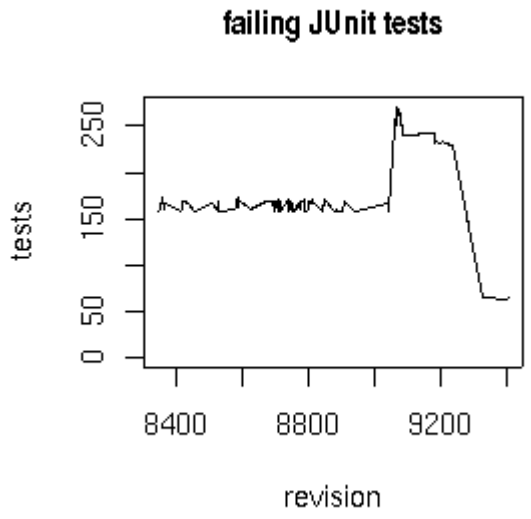
Citation
Willighagen, E. (2007, December 19). Test results for the CDK 1.0.x branch. *Chem-bla-ics*. https://doi.org/10.59350/ry980-qya21

Keywords
Cdk

The Chemistry Development Kit has never really been without any bugs, which is reflected in the number of failing JUnit tests. For trunk/ this is today 106 failing tests (live stats). The stable cdk-1.0.x/ branch, however, the number of failing tests is not much lower: 64 failing tests today (live stats).

Overall, only a low percentage of the tests fails (<2% for cdk-1.0.x/ and <3% for trunk/), and, more importantly, it is particular algorithms that are typically broken. For example, in the structgen module 8 tests fail, for both CDK versions. In the `cdk-1.0.x/` branch it is the valency checker code that causes quite a few fails, which I discussed in Atom typing in the CDK and which is the reason for the atom type perception refactoring in progress in trunk/ (see Evidence of Aromaticity). Not all code in trunk/ has yet been updated yet, and this causes quite a few failing tests for `trunk/` in the `reaction`, `qsarAtomic` and `qsarBond` modules.

Back to the `cdk-1.0.x/` branch. Previous CDK releases tended to have around 40 failing tests, so I was worried about the number of tests failing now. Maybe backported patches causes additional fails? To study that I had my machine run the JUnit tests for all revisions of the `cdk-1.0.x/` branch since the branch was made in commit 8343. The result looks like:



Indeed, it is a number of backports that cause the clear increase in bugs between commit 9044 and 9058. Nothing particular I can see, and worse, the intermediate revisions do not compile and do not have test restults:

```
104 9044 3731  84  73  979.709  0
105 9045    0   0   0    0.000  0
106 9046    0   0   0    0.000  0
107 9047    0   0   0    0.000  0
108 9048    0   0   0    0.000  0
109 9049    0   0   0    0.000  0
110 9050    0   0   0    0.000  0
111 9051    0   0   0    0.000  0
```

chem-bla-ics

```
112 9052    0   0   0    0.000  0
113 9053    0   0   0    0.000  0
114 9054    0   0   0    0.000  0
115 9055    0   0   0    0.000  0
116 9056    0   0   0    0.000  0
117 9057    0   0   0    0.000  0
118 9058 3740 104 146  989.566  0
```

I should have taken more care when merging in these patches, even though they are supposed to fix issues:

```
Merged r8697: Add a method to the query atom container creator which creates
an
  queryatomcontainer. This replaces each pseudoatom to an anyatom.
Merged r8699 and r8700: Added test file by Volker (see cdk-user) for the
shortest path problem;
  JUnit test provided by Volker Haehnke (haehnke - bioinformatik uni-frankfurt
de), somewhat
  rewritten.
Merged r8701: Renamed a variable to comply with http://en.wikipedia.org/wiki/
Dijkstra's_algorithm
Merged r8751: Bug fixes for bugs #1783367 'SmilesParser incorrectly assigns
double bonds' and
  #1783381 'SmilesParser uses Molecule instead of IMolecule'. Test case for
bug #1783367.
Merged r8754 and r8773: Fix and test case for bug #1783547 and #1783546 'Lost
aromaticity in
  SmilesParser with Biphenyl and Benzene'
Merged r8774: Add a MDL RXN reader which uses the MDLV2000Reader instead of
the MDLReader
Merged r8775, r8776, r8777: bug fixes for #150354 #1783774 #1778479 in the
SmilesParser,
  SmilesGenerator and MDLWriter/PseudoAtom.
Merged r8791: Code for v,mass atom two digits mass atom and exception
handeling
Merged r8800: Fixed reading of MDL molfiles with exactly 12 columns (==valid)
in the bond block
Merged r8802: Made a little more memory efficient by removing unnesscary
cloning operations
Merged r8803: Fixed it so that we make a deep copy of the input molecule
Merged r8809: Added code to work on a local copy of theinput molecule
Merged r8811: Updated Javadocs
Merged 8824 8821 8820 8819 8817 8816: Added code to properly work on a local
copy
```

chem-bla-ics

I'm quite sure it must be the deep-cloning fix ported from the commits 8800-8824. I already fixed a number of bugs in the IP calculation code which is still a good deal of the failing tests in the cdk-1.0.x/ branch (and affects trunk/ too), as can be seen by the drop in bugs just after the big increase:

```
r9079 | egonw | 2007-10-15 13:24:10 +0200 (Mon, 15 Oct 2007) | 1 line

Renamed container to localClone to clear up code. Fixed a bug where the
uncloned atoms was
searched in the cloned atomcontainer. More bugs like this are in the code.
Miguel is contacted
about this problem.
------------------------------------------------------------------------
r9082 | egonw | 2007-10-15 13:48:15 +0200 (Mon, 15 Oct 2007) | 1 line

Renamed container to localClone to clear up code. Fixed a bug where the
uncloned atoms was
searched in the cloned atomcontainer.
```

The big drop in number of fails is caused by the removal of the SMARTS code from the branch, which has been present since the start of the branch (see this page).

From this analysis I conclude that CDK 1.0.2 can soon be released. With the note that the ionization potential calculation is not safe to use.