Code coverage: making sure your code is tested

Published November 28, 2006

Citation
Willighagen, E. (2006, November 28). Code coverage: making sure your code is tested. *Chem-bla-ics*. https://doi.org/10.59350/gtt31-tee03

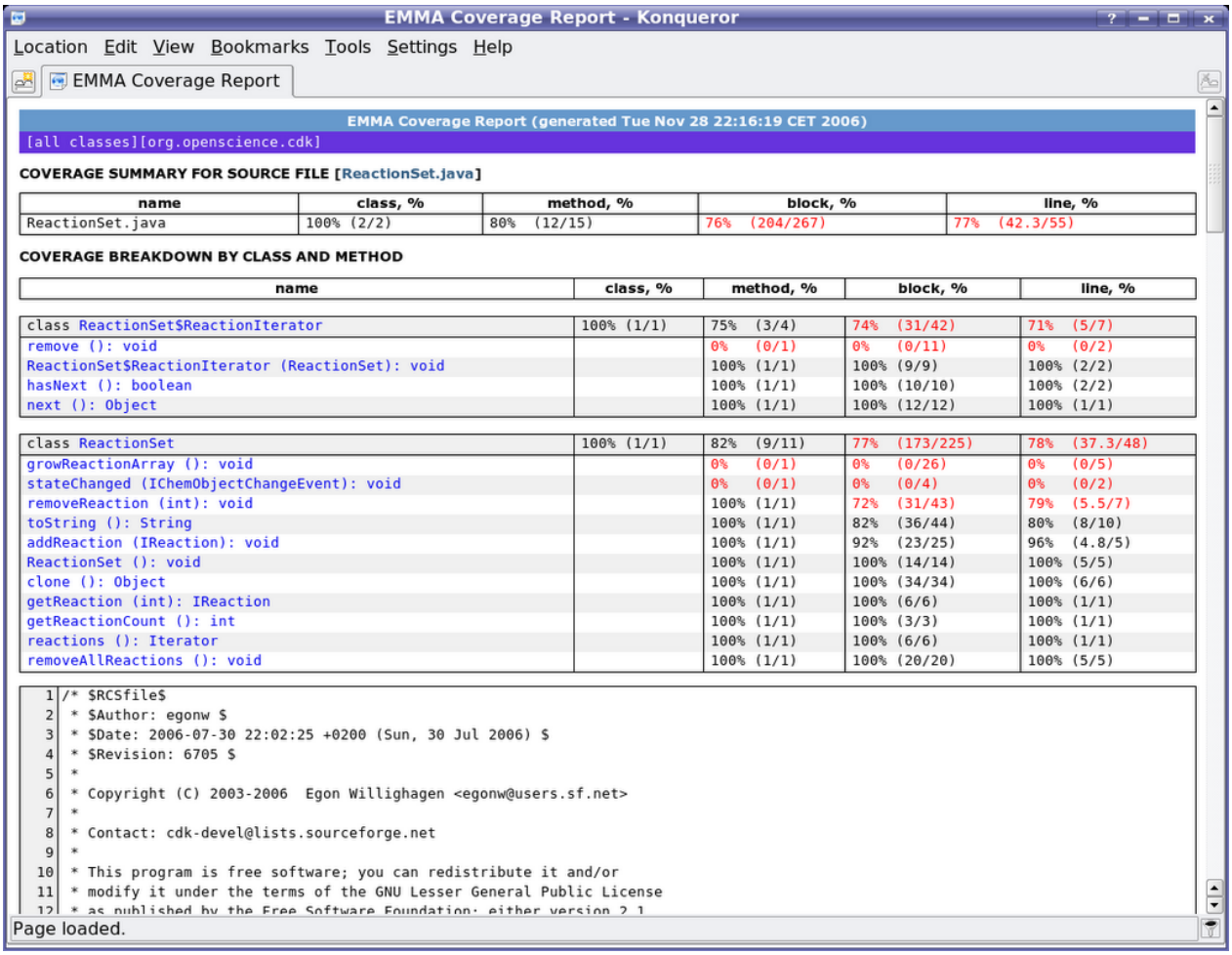
Keywords
Opensource, Cdk, Junit

Abstract
Recently I discussed JUnit testing from within Eclipse , and blogged at several occasions about it in other situations. I cannot stress enough how useful unit testing is: it adds this extra set of eyeballs to make bugs shallow. And it does that, indeed.

Recently I discussed JUnit testing from within Eclipse , and blogged at several occasions about it in other situations. I cannot stress enough how useful unit testing is: it adds this extra set of eyeballs to make bugs shallow. And it does that, indeed.

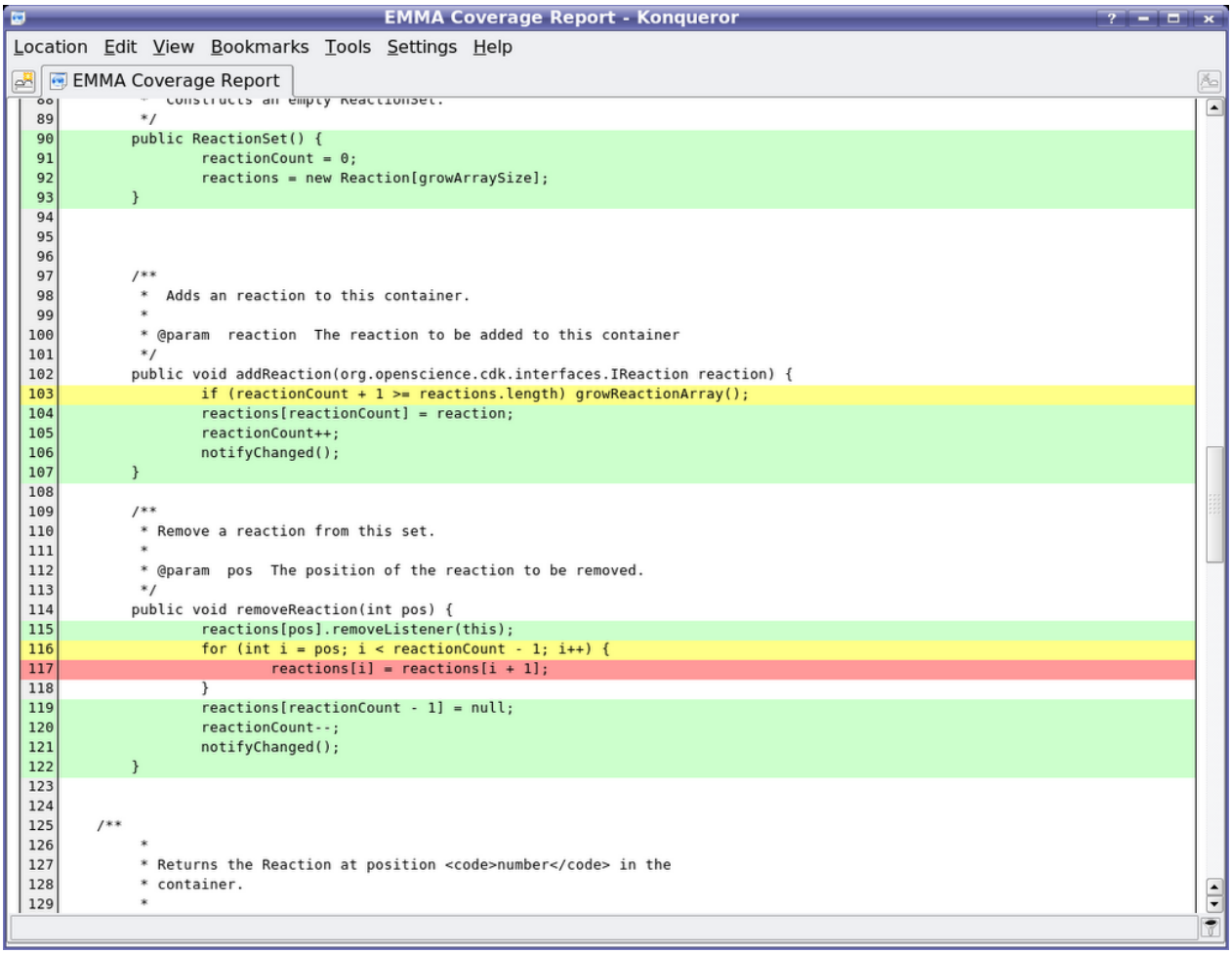
Ensuring that you actually test all the code you write, however, is not easy. A couple of years back I read an article about Hansel, which does code coverage checking, but never got it nicely working for the CDK project. Never looked at that lately, so no idea how the current release would work out. Hansel is an extension of JUnit, and requires hard coding class names, which conflicts with CDK's module setup.

Thomas Kuhn pointed me last week to Emma, which seems a nice tool. It does not require hacking our source, and generates cool HTML:



And even highlights the source code:

chem-bla-ics



BTW, I seem to be in good company: Classpath is using it too.

Below is the command I issued to generate the HTML output. Rajarshi, maybe this can be integrated into Nightly? Note that it only runs the tests for the data module:

```
ant dist-large dist-test-large
java -cp ~/tmp/emma-2.0.5312/lib/emma.jar emmarun \
  -cp develjar/junit.jar:dist/jar/cdk-svn-20061128.jar:dist/jar/cdk-test-
svn-20061128.jar \
  -r html -sp src junit.textui.TestRunner org.openscience.cdk.test.MdataTest
```