chem-bla-ics

Cleaner CDK Code #7: understand what the code is supposed to do

Egon Willighagen

Published August 5, 2010

Citation
Willighagen, E. (n.d.). In *chem-bla-ics*. chem-bla-ics. https://doi.org/10.59350/0a1n4-5q396

Keywords
Cdk

Copyright
Copyright © Egon Willighagen 2010. Distributed under the terms of the Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

It has been a while since I posted a blog post in this series (see below for a full list so far), but was fixing a problem for Nina (OpenTox), and found some code I did not understand. So, here's another useful tip (IMHO) for writing CDK code. Well, in particular, patching CDK code in this case.

Understand what the code is doing

Understanding what CDK code is doing is not always obvious. Theoretically, the mix of clean code, JavaDoc, and code comments explain enough, but sometimes it is not. Then, it is useful to see how code has evolved: who wrote the code; what was the commit message around that code (further documentation); what did the code look like before that change. This is where *git blame* comes in (see also how GitHub used this).

Now, to see the history of a source file, we do, for example:

```
git pickaxe -L 840 -- src/main/org/openscience/cdk/smiles/DeduceBondSystemTool.java
```

Which will output something like:

```
00000000 (Not Committed Yet 2010-08-05 16:35:29 +0100 840)      *
00000000 (Not Committed Yet 2010-08-05 16:35:29 +0100 841)      * @param interrupted true
00000000 (Not Committed Yet 2010-08-05 16:35:29 +0100 842)      */
00000000 (Not Committed Yet 2010-08-05 16:35:29 +0100 843)     @TestMethod("testInterrupt
7c5c872c (egonw             2008-02-22 19:13:45 +0000 844)      public void setInterrupte
7c5c872c (egonw             2008-02-22 19:13:45 +0000 845)              this.interrupted
7c5c872c (egonw             2008-02-22 19:13:45 +0000 846)      }
7c5c872c (egonw             2008-02-22 19:13:45 +0000 847)
00000000 (Not Committed Yet 2010-08-05 16:35:29 +0100 848)     @TestMethod("testInterrupt
7c5c872c (egonw             2008-02-22 19:13:45 +0000 849)      public boolean isInterrup
7c5c872c (egonw             2008-02-22 19:13:45 +0000 850)              return this.inter
7c5c872c (egonw             2008-02-22 19:13:45 +0000 851)      }
7c5c872c (egonw             2008-02-22 19:13:45 +0000 852)
7c5c872c (egonw             2008-02-22 19:13:45 +0000 853) }
```

The uncommited parts shows a patch I am working on. You see that these methods have last been touched by me. So, it is interesting to see why I added those, for which we look up the commit message (using he commit hash in the first column). We use `git show --shortstat 7c5c872c`:

```
commit 7c5c872c6be0e6a24a519eb227e7ed0b76ac37d6
Author: egonw
Date:   Fri Feb 22 19:13:45 2008 +0000

    Merged the branch egonw/maintest: sets up src/main and src/test for splitting main li
```

chem-bla-ics

```
    git-svn-id: https://cdk.svn.sourceforge.net/svnroot/cdk/trunk/cdk@10219 eb4e18e3-b210

 2340 files changed, 398392 insertions(+), 92370 deletions(-)
```

We now actually see an important commit in history. This was the commit where the source code was split up into the two folders we have right now.

Dealing with moved files

Normally, we could do the following to see what the code was like before this commit, by adding the revision hash and a ^ to indicate the last commit before that hash, with:

```
git pickaxe -L 840 7c5c872c^ -- src/main/org/openscience/cdk/smiles/DeduceBondSystemTool.
```

However, this will not work here, because the source code originally was elsewhere. In particular, the code used to be all in src/. So, the original location was src/main/org/openscience/cdk/smiles/DeduceBondSystemTool.java. Therefore, we must use instead:

```
git pickaxe -L 840 7c5c872c^ -- src/org/openscience/cdk/smiles/DeduceBondSystemTool.java
```

And then we can continue again.

Previous topics

- Cleaner CDK Code #6: set the CDKException's cause Exception
- Cleaner CDK Code #5: develop against interfaces
- Cleaner CDK Code #4: inheriting JavaDoc from super classes and interfaces
- Cleaner CDK Code #3: run the PMD tests
- Cleaner CDK Code #2: String.contains() and logger messages
- Cleaner CDK Code #1: List and the for-each loop