chem-bla-ics

Finding the commit that causes the regressions...

Egon Willighagen

Published November 10, 2008

Citation

Willighagen, E. (2008). Finding the commit that causes the regressions... In *chem-bla-ics*. chem-bla-ics. https://doi.org/10.59350/pmme3-yzj05

Keywords

Cdk, Git

CDK 1.1.x releases are well in progress, but a recent commit broke a number of unit tests. Here comes git-bisect.

```
$ git checkout -b my-local-1.2 cdk1.2.x
$ git bisect start
$ git bisect bad
$ git bisect good 8219139e9236ab8036e9d08c13fcd0482d500c79
```

These lines indicate that the current version (HEAD) is broken, and that revision 8219139e9236ab8036e9d08c13fcd0482d500c79 was OK. Now, `git-bisect` does the proper thing, and starts in the middle, allowing me to run my tests, and issue a `git bisect bad` or `git bisect good` depending on whether my test fails or not. The test I am running is:

```
$ ant clean dist-all test-dist-all jarTestdata
$ ant -Dmodule=smarts test-module
$ git bisect [good|bad]
```

So, if I had to inspect 1024 commits, I'd found the bad commit in 10 times running this test suite. For the culprit I was after it was 6 times. The outcome was this commit, what I already suspected and emailed about to the cdk-devel mailing list:

```
[fa49ac603c36908f341b25d52a78435cdb8ca4d3] atomicNumber set as default (Integer) CDKConst
```