slides

September 20, 2022

##

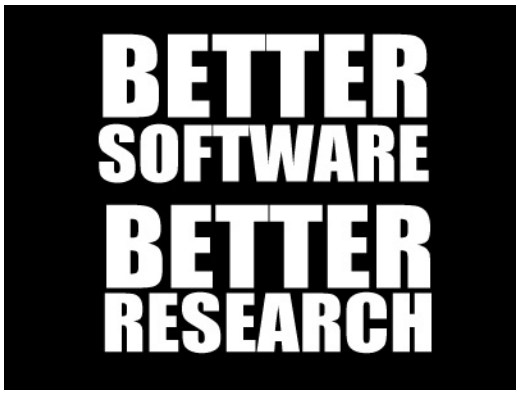
Workshop

#

Testing

##

Chris Edsall



- Perhaps one of the best ways to make software better is to test that it is actually doing what we think it it

1 Why

Currently: - Reproducability crisis - Retractions

Testing: - Gives collaborators confidence - Attracts more users => enhanced reputation - Finds bugs earlier in the SDLC ("shift left")

SCIENTIFIC PUBLISHING

A Scientist's Nightmare: Software Problem Leads to Five Retractions

Until recently, Geoffrey Chang's career was on a trajectory most young scientists only dream about. In 1999, at the age of 28, the protein crystallographer landed a faculty position at the prestigious Scripps Research Institute in San Diego, California. The next year, in a cer-

2001 *Science* paper, which described the structure of a protein called MsbA, isolated from the bacterium *Escherichia coli*. MsbA belongs to a huge and ancient family of molecules that use energy from adenosine triphosphate to transport molecules across cell membranes. These

LETTERS

edited by Etta Kavanagh

Retraction

WE WISH TO RETRACT OUR RESEARCH ARTICLE "STRUCTURE OF MsbA from *E. coli*: A homolog of the multidrug resistance ATP binding cassette (ABC) transporters" and both of our Reports "Structure of the ABC transporter MsbA in complex with ADP•vanadate and lipopolysaccharide" and "X-ray structure of the EmrE multidrug transporter in complex with a substrate" (*1*–*3*).

The recently reported structure of Sav1866 (*4*) indicated that our MsbA structures (*1*, *2*, *5*) were incorrect in both the hand of the structure and the topology. Thus, our biological interpretations based on these inverted models for MsbA are invalid.

An in-house data reduction program introduced a change in sign for anomalous differences. This program, which was not part of a conventional data processing package, converted the anomalous pairs (I+ and I−) to (F− and F+), thereby introducing a sign change. As the diffraction density for the connecting loop regions. Unfortunately, the use of the multicopy refinement procedure still allowed us to obtain reasonable refinement values for the wrong structures.

The Protein Data Bank (PDB) files 1JSQ, 1PF4, and 1Z2R for MsbA and 1S7B and 2F2M for EmrE have been moved to the archive of obsolete PDB entries. The MsbA and EmrE structures will be recalculated from the original data using the proper sign for the anomalous differences, and the new Cα coordinates and structure factors will be deposited.

We very sincerely regret the confusion that these papers have caused and, in particular, subsequent research efforts that were unproductive as a result of our original findings.

GEOFFREY CHANG, CHRISTOPHER B. ROTH, CHRISTOPHER L. REYES, OWEN PORNILLOS, YEN-JU CHEN, ANDY P. CHEN

Department of Molecular Biology, The Scripps Research Institute, La Jolla, CA 92037, USA.

Retraction: Measures of Clade Confidence Do Not Correlate with Accuracy of Phylogenetic Trees

Barry G Hall, Stephen J Salipante

Published: July 20, 2007 • https://doi.org/10.1371/journal.pcbi.0030158

In *PLoS Computational Biology,* volume 3, issue 3, doi:10.1371/journal.pcbi.0030051:

As a result of a bug in the Perl script used to compare estimated trees with true trees, the clade confidence measures were sometimes associated with the incorrect clades. The error was detected by the sharp eye of Professor Sarah P. Otto of the University of British Columbia. She noticed a discrepancy between the example tree in Figure 1B and the results reported for the gene *nuoK* in Table 1, and requested that she be sent all ten *nuoK* Bayesian trees. She painstakingly did a manual comparison of those trees with the true trees, concluded that for that dataset there was a strong correlation between clade confidence and the probability of a clade being true, and suggested the possibility of a bug in the Perl script. Dr. Otto put in considerable effort, and we want to acknowledge the generosity of that effort.

The major conclusion of our paper, as given in its title, is therefore invalid, and the paper must be retracted. It is important to stress that the responsibility for the necessity of retracting our paper is entirely mine (Barry Hall), and that my coauthor Stephen J. Salipante bears none of the responsibility. I wrote the Perl script and failed to check its accuracy sufficiently.

2 What

3 Science Test Cases

- e.g. “Does this formulation of the climatge model, when spun up and run for 20 model years produce the quasi- biannial oscilartion in the right place and time?”
- important, but not the focus of this workshop

4 Integration tests

- Tests the application end to end
- Can be time consuming

5 Unit Tests

- Test the smallest sensible unit, e.g. a **single function**
- Usually much simpler, constrained test cases
- Easier to reason about the test cases
- (Ideally) **runs quickly**
- Can be used for Test Driven Development
  - When you write the test before you write the code that implements it

  - You know when to stop coding!
- Unit testing implies the code is sufficiently modular

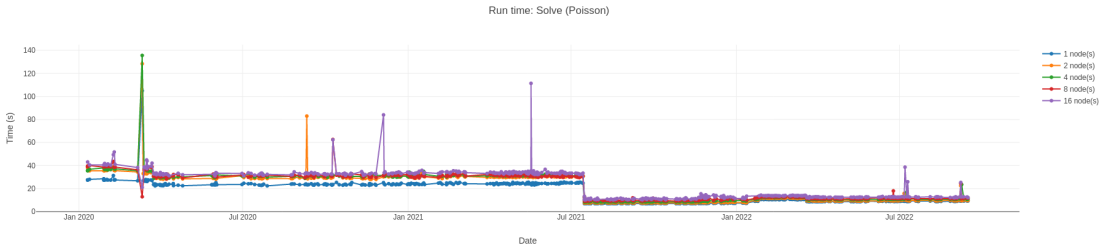
6 What makes a good test case?

- one simple known good example
- maximum and minimum examples
- edge / corner cases
  - test for off by one errors
  - complex boundary calculations
  - diagonal cases (stencil code)

7 Regression Tests

- make sure things *still* work the way they should
- detect
  - regressions in **correctness** (common to reintroduce an old bug when fixing a new one)
  - regressions in **perfromance**

e.g. https://fenics.github.io/performance-test-results/



8 Other types

8.1 Fuzz Testing

- feed the software under test **random input** or **syntacticaly correct but erroneous** data
- mostly used in security

8.2 Black Box

- "Golden outputs"

8.3 Property Based Testing

- Origianlly `QuickCheck` from Haskell
- automates the generation of test cases
- finds reduced cases on failure
- Python framework is called `hypothesis`

> "Program testing can be used to show the presence of bugs, but never to show their absence!"

Edsger W. Dijkstra

9 Coverage

- What fraction of the lines of code are run when running a test suite
- Diminishing returns when aiminig for 100%
- Python framework is calle `coverage`

10 Test Frameworks

- You can use bare asserts
- frameworks provide "affordances"
  - convenience functions like `assert_almost_equal`
  - grouping tests
  - running subsets of tests
  - providing "fixtures"
    - eg setting up and tearing down a populated database

11 Common Test Frameworks

11.1 Python

- pytest
- (`Unittest` is builtin, but don't use it)

11.2 C++

- Catch2
- Google Test

11.3 Fortran

- Currently don't have a recomendation, There are a number on https://fortran-lang.org/en/packages/programming/

11.4 Julia

- `Test` module

12 Pytest

13 Exercise 0 - Install Pytest

13.1 You might already have it!

```
pytest -v
```

13.2 pip / venv

Linux / mac

```
cd python
python3 -m venv pytestenv
source pytestenv/bin/activate
pip install -r requirements.txt
```

13.3 conda

Linux / mac

```
conda-env create --name pytestenv -f requiremnets.txt
```

14 Exercise 1 - Run Pytest

```
pytest
```

15 Exercise 2 - Add a test

Check that dividing any particular integer by 1 gives the same integer.

Aside: Checking thins in the general sense for all integers is a property based test for which you would use hypothesis

16 Expected failures

- Sometimes you want to make sure the code processes exceptions properly
- Normally this would exit the interpreter
- pytest can check in side a context

```
with pytest.raises(YourException):
    thing_that_will_hopefully_fail()
```

17 Exercise 3 - Add an expected fail test

Check that divide function rases the `ZeroDivisionError` exception when the divisor is zero.

18 Fixtures

- a fixture sets up a resource that will be used between multiple tests
- e.g.:
  - set up and tear down a database
  - download a large file (only once)
- decorate the setup function with `@pytest.fixture`
- specify that you want a fixture by adding it's name to the test function call

19 Exercise 4 - Add a test with a fixture

20 Exercise 5 - Install Catch2

- Catch2 can be used as a header only library, so it is sufficient to download the `.hpp` file and add it to your project
- Boost Licence

```
make fetch
make
./TestCase --success
```

21 Exercise 6 - Make the test pass

Hint: look at the commented out line

22 Hints and Tips

- `pytest --pdb` will drop you in to the python debugger on a failed test
- Whenever you fix a bug add a test
- Configure your IDE to run the tests
- Run the fast tests in your git commit hooks
- Run tests before pushing

[ ]: