
Rotten green tests in Java, Pharo and Python

Vincent Aranega^{*1}

¹CRIStAL – Univ. Lille, CNRS, Centrale Lille Institut – France

Résumé

Rotten Green Tests are tests that pass, but not because the assertions they contain are true: a rotten test passes because some or all of its assertions are not actually executed. The presence of a rotten green test is a test smell, and a bad one, because the existence of a test gives us false confidence that the code under test is valid, when in fact that code may not have been tested at all. This article reports on an empirical evaluation of the tests in a corpus of projects found in the wild. We selected approximately one hundred mature projects written in each of Java, Pharo, and Python. We looked for rotten green tests in each project, taking into account test helper methods, inherited helpers, and trait composition. Previous work has shown the presence of rotten green tests in Pharo projects; the results reported here show that they are also present in Java and Python projects, and that they fall into similar categories. Furthermore, we found code bugs that were hidden by rotten tests in Pharo and Python. We also discuss two test smells -*missed fail* and *missed skip* -that arise from the misuse of testing frameworks, and which we observed in tests written in all three languages.

^{*}Intervenant