



Gene Gotimer

Test Your Unit Tests: Mutation Testing



UNIT TESTING

What are unit tests?



Unit tests show that the code does what the developer intended it to do.

Unit tests

- must be automated
- should be independent
- have no external dependencies
- usually test individual methods or smaller
- become our safety net for introducing changes

CODE COVERAGE

What is code coverage?



Code coverage tools measure how much of the code was executed when the tests were run.

Covered code *might* be tested.

Uncovered code absolutely *not tested*.

80% code coverage means:

- At least 20% untested
- Could be 100% untested

CODE COVERAGE

Example



```
@Test
public void testCreateAckReport() throws IOException {
    final Ack ack = createTestAck();

    final byte[] pdfBytes = service.createAckReport(ack);

    assertNotNull(pdfBytes);
    assertIsPdf(pdfBytes);
}

private static void assertIsPdf(final byte[] bytes) {
    final byte[] header = Arrays.copyOf(bytes, 1024);
    final String headerStr = new String(header, StandardCharsets.UTF_8);

    assertTrue(headerStr.contains("%PDF"));
}
```

CODE COVERAGE

Example



```
for (Method method : object.getClass().getMethods()) {
    if (method.getName().startsWith("set")) {
        if (method.getParameterTypes().length > 0) {
            Class paramClass = method.getParameterTypes()[0];
            if (paramClass.getName().equals("java.lang.String")) {
                method.invoke(object, "JUNIT TEST");
            } else if (paramClass.getName().equals("java.lang.Object")) {
                method.invoke(object, new Object());
            } else if (paramClass.getName().equals("java.util.Date")) {
                method.invoke(object, new java.util.Date());
            }
        }
    }
}
```

CODE COVERAGE

Example



```
public void testAddTaxCreditSuccess() {  
    try {  
        Member person = generateTestMember();  
        String spouseId = person.findSpouseId();  
        assertNotNull(service.addTaxCredit(person, spouseId));  
    } catch (Throwable ex) {  
        assertNotNull(ex);  
    }  
}
```

What is mutation testing?




```
public int foo(int i) {
    i++; → i--;
    return i;
}
```

```
public String bar(String s) {
    if (s == → != null) {
        // do something
    }
}
```



Example PIT Report



```
38     String check(final Play human, final Play computer) {
39         final StringBuilder result = new StringBuilder();
40 1     if (computer == human) {
41 1         draws++;
42         result.append("Draw.\n");
43     } else {
44         String beats = human.beats(computer);
45 1         if (null == beats) {
46 1             losses++;
47             beats = computer.beats(human);
48             result.append(String.format("%s %s %s. You lose.%n", computer, beats,
49         } else {
50 1             wins++;
51             result.append(String.format("%s %s %s. You win.%n", human, beats, com
52         }
53     }
54 1     return result.toString();
55 }
```

Light green shows line coverage, dark green shows mutation coverage.

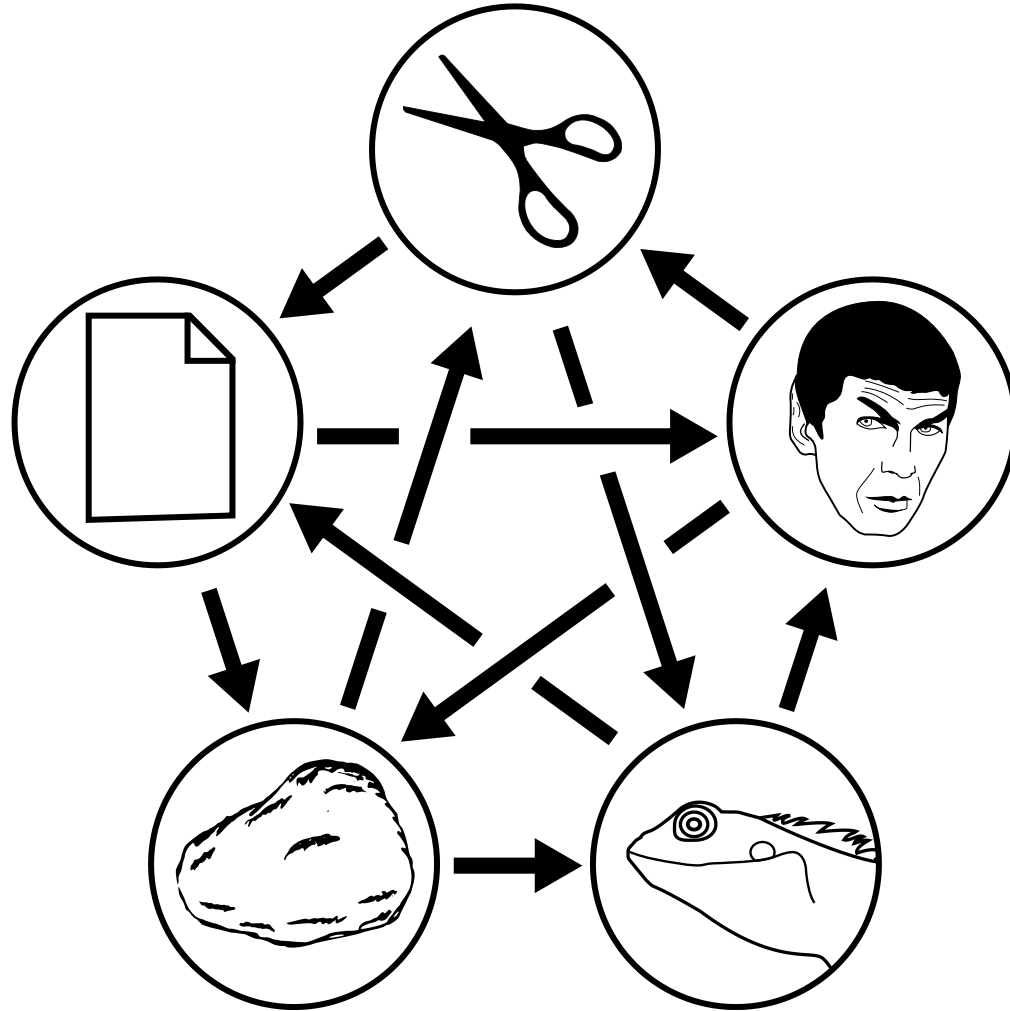
Light pink show lack of line coverage, dark pink shows lack of mutation coverage.



Demo

<https://github.com/OtherDevOpsGene/roshambo>

Rock
Paper
Scissors
Lizard
Spock



DEMO

Rock
Paper
Scissors
Lizard
Spock



The Big Bang Theory ©2013, Warner Bros. TV
<https://youtu.be/iSHPVCBsnLw?t=38>



Demo

<https://github.com/OtherDevOpsGene/roshambo>



Wrap-up

WRAP-UP

Key takeaways



- Use mutation testing to make sure your unit tests are actually testing what you need to be tested.
- Code coverage tools only show how much code is absolutely not tested.
- With new code, mutation testing can show you edge conditions you haven't considered.
- With legacy code, mutation testing can help you really understand the code you are changing.

WRAP-UP

Tools

JUnit JUnit <https://junit.org>JaCoCo <https://www.eclemma.org/jacoco>PIT <https://pitest.org>NUnit <https://nunit.org>AltCover <https://github.com/SteveGilham/altcover>Stryker.NET <https://stryker-mutator.io>

WRAP-UP

Tools for Other Languages

C/C++: Mull <https://github.com/mull-project/mull>

Go: go-mutesting <https://github.com/zimmski/go-mutesting>

JavaScript: StrykerJS <https://stryker-mutator.io>

PHP: Infection <https://infection.github.io>

Python: Mutmut <https://github.com/boxed/mutmut>

Python: MutPy <https://github.com/mutpy/mutpy>

Ruby: Mutant <https://github.com/mbj/mutant>

Scala: Stryker4s <https://stryker-mutator.io>



Questions?