COM4506/6506: Testing and Verification in Safety Critical Systems
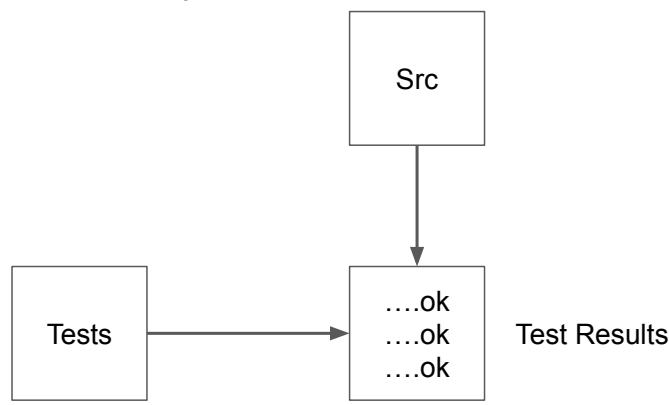
Dr Ramsay Taylor

Contents

- Test Adequacy for Test Generation Feedback
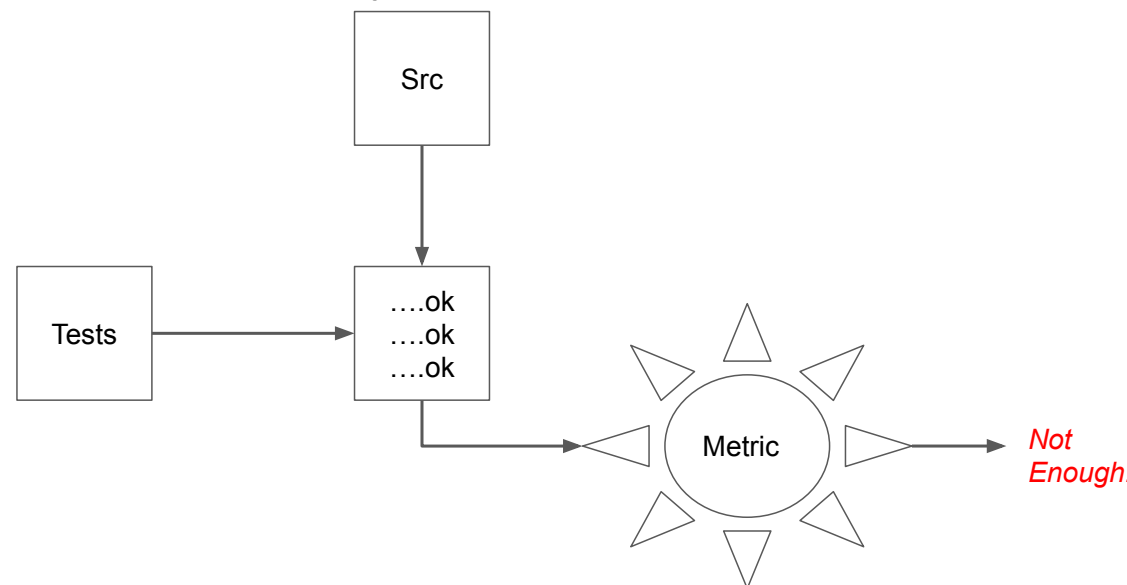- Code coverage information
- Mutation Testing counter examples

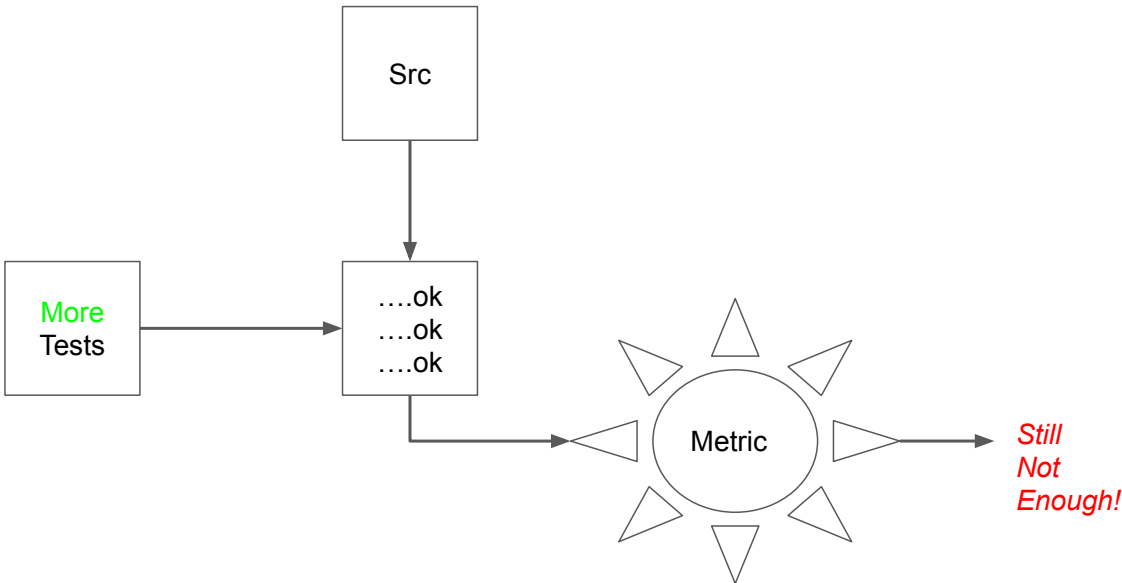
Test Adequacy

Src

Tests → ….ok ….ok ….ok   Test Results

"Everything passes, its done, I'm going to the pub…"

"No, you need to do more!"

Test Adequacy

Src

Tests → ….ok ….ok ….ok → Metric → *Not Enough!*

Test Adequacy

Src

More
Tests

....ok
....ok
....ok

Metric

Still Not Enough!

Test Adequacy Useful Feedback

```
    |    -module(abiftest).
    |    -export([dv/2]).
    |
    |    dv(A,B) ->
1..|        if (A == 0) and (B > 4) ->
1..|            B;
    |        true ->
0..|            B / A
    |    end.
```

dv(0,5)

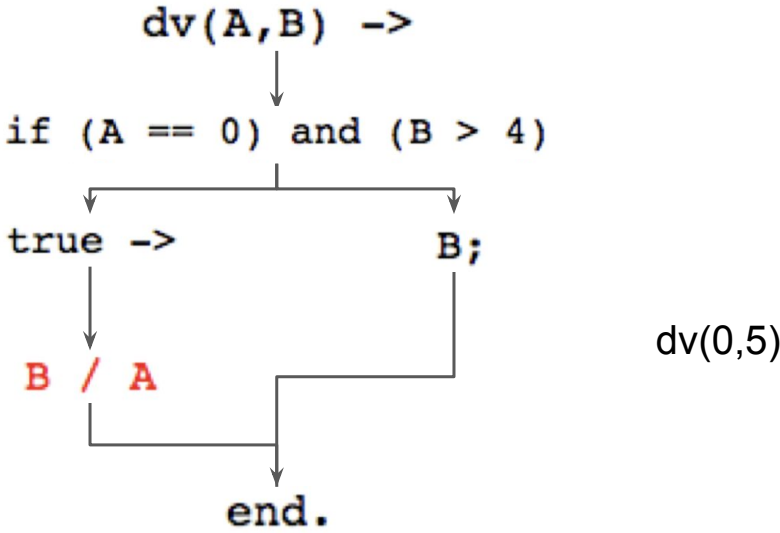## Test Adequacy Useful Feedback

```
    |    -module(abiftest).
    |    -export([dv/2]).
    |
    |    dv(A,B) ->
1..|        if (A == 0) and (B > 4) ->
1..|            B;
    |        true ->
0..|            B / A
    |    end.
```

True, so need to make it false to hit the other line/branch

dv(0,5)

Test Adequacy Useful Feedback

```
dv(A,B) ->

if (A == 0) and (B > 4)

true ->          B;

B / A

end.
```

dv(0,5)

Test Adequacy Useful Feedback

(A == 0) and (B > 4)

- matched: 1
- non-matched: 2

dv(0,5)
dv(5,5)
dv(5,0)

When false:

	matched	non-matched
A == 0	0	2
B > 4	1	1

Mutation Testing Counterexamples
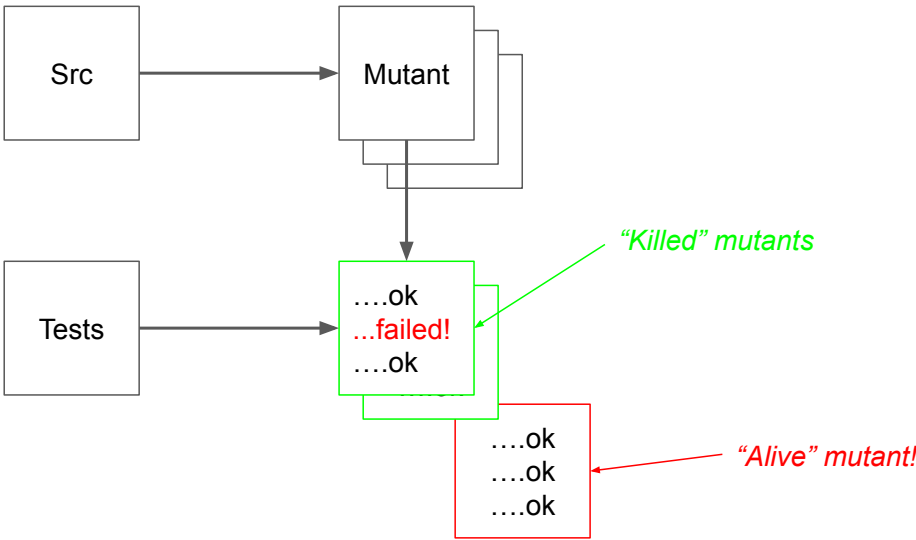
Src → Mutant

Tests → ....ok
...failed!
....ok

"Killed" mutants

....ok
....ok
....ok

"Alive" mutant!

Mutation Testing Counterexamples

Src → Mutant

Tests → ....ok
...failed!
....ok

....ok
....ok
....ok

"Alive" mutant!
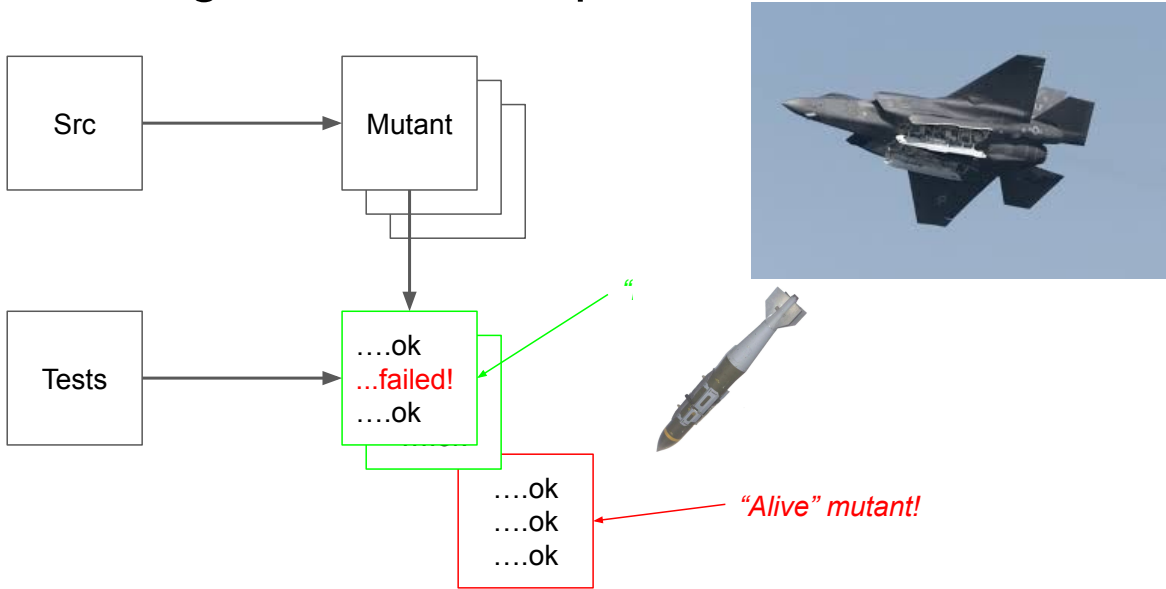
Mutation Testing Counterexamples

```
public class FtoC {

    public static double ftoc(double tempf) {
        double x, y, result;

        x = tempf-32;
        y = 5.0 / 9.0;
        result = x * y;

        return(result);
    }

    public static double ctof(double tempc) {
        double x, y, result;

        y = 9.0 / 5.0;
        x = tempc * y;
        result = x - 32;

        return(result);
    }
}
```

Live mutants are *specific counterexamples* that show a weakness of the test set.

Test to kill them can be crafted.

The mutant should be kept around as a test set test!

Summary

- Test adequacy metrics give more than just a numeric metric of test set quality
- Various code coverage results can suggest new tests with some analysis
- Living mutants - if they aren't semantically equivalent to the original - also direct new tests