



ExLi: An Inline Test Generation Tool for Java



Yu Liu¹



Aditya Thimmaiah¹



Owolabi Legunsen²



Milos Gligoric¹

Inline Tests

- We recently proposed **Inline tests**[1] that introduce **new granularity of tests** for checking individual program statements

```
class Convert{
    private static MutableFst convertFrom(...){
        ...
        while ((line = br.readLine()) != null){
            String[] tokens = line.split(regexToSplitOn);

        }
    }
}
```

Inline Tests

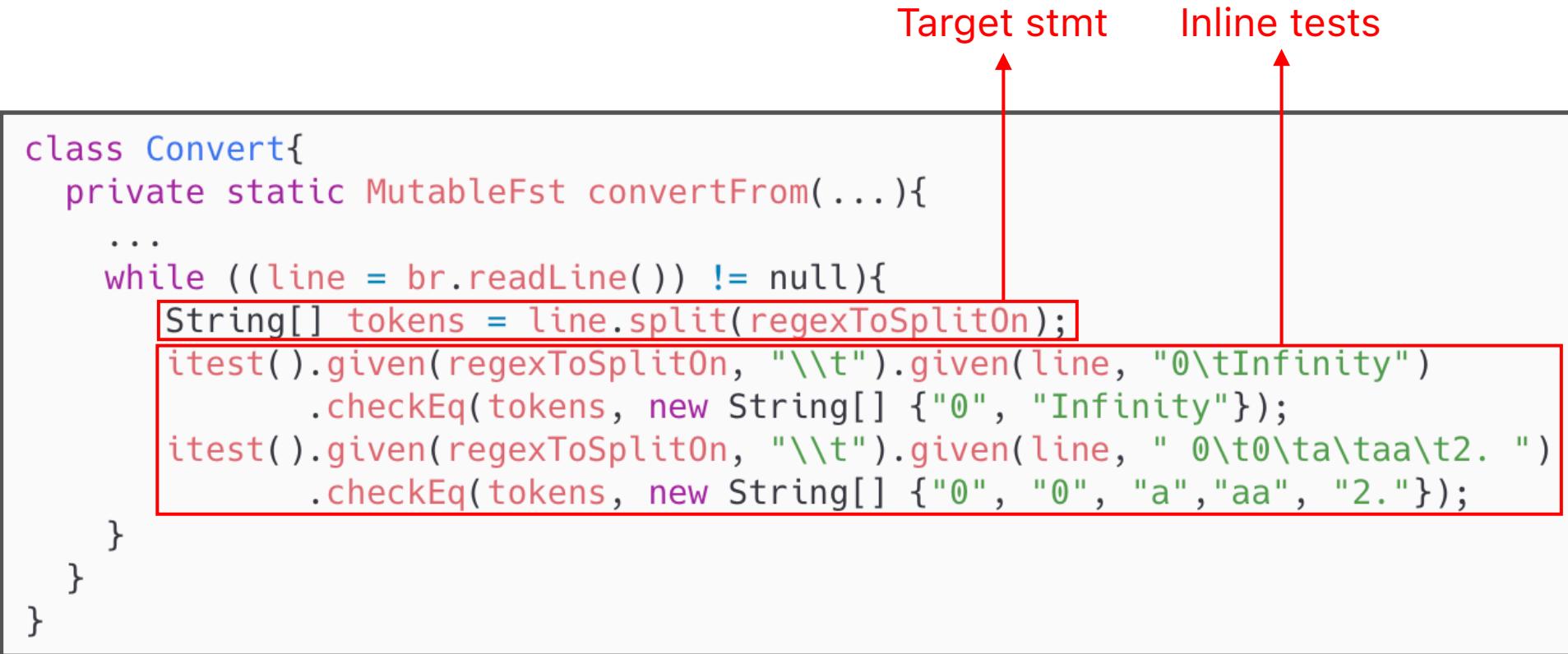
- We recently proposed **Inline tests**[1] that introduce **new granularity of tests** for checking individual program statements

The diagram shows a code snippet within a black-bordered box. A red box highlights the statement `String[] tokens = line.split(regexToSplitOn);`. A red arrow points from the text "Target stmt" at the top to this highlighted statement.

```
class Convert{
    private static MutableFst convertFrom(...){
        ...
        while ((line = br.readLine()) != null){
            String[] tokens = line.split(regexToSplitOn);
        }
    }
}
```

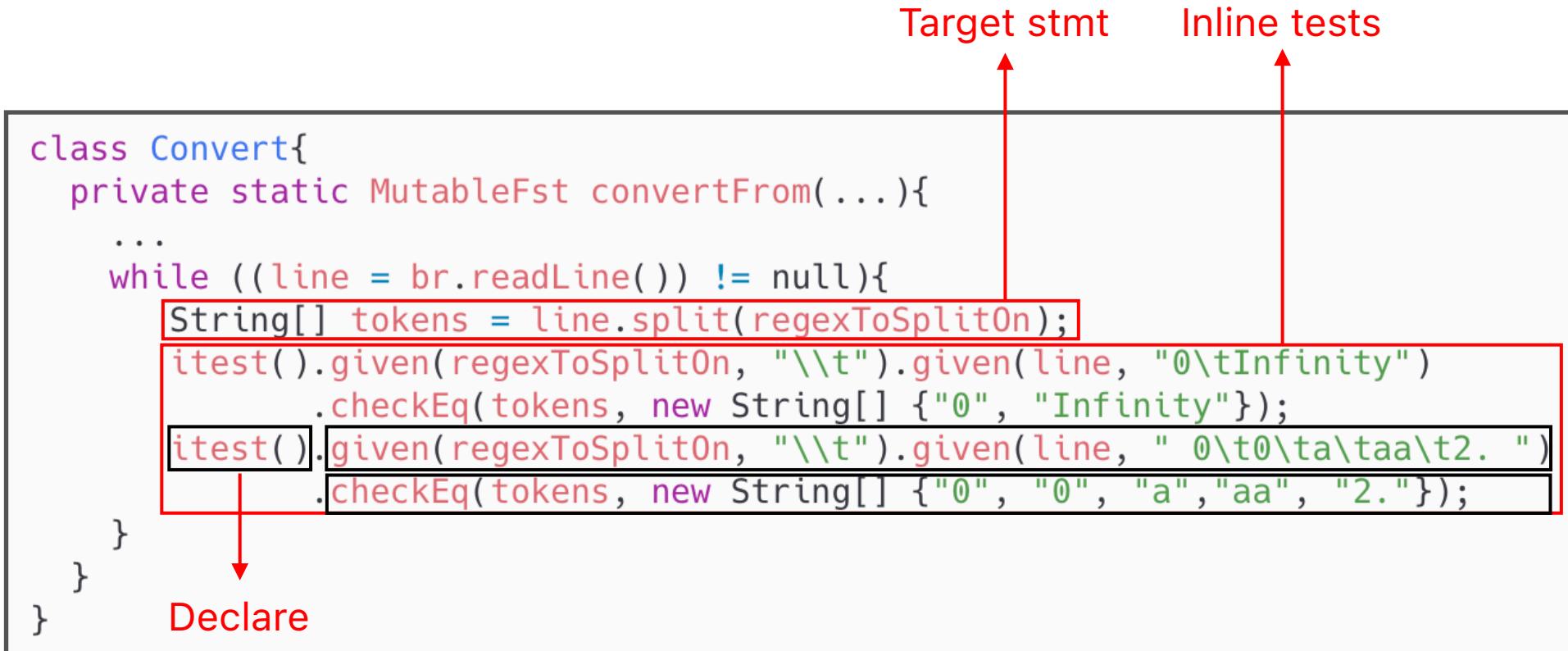
Inline Tests

- We recently proposed **Inline tests**[1] that introduce **new granularity of tests** for checking individual program statements



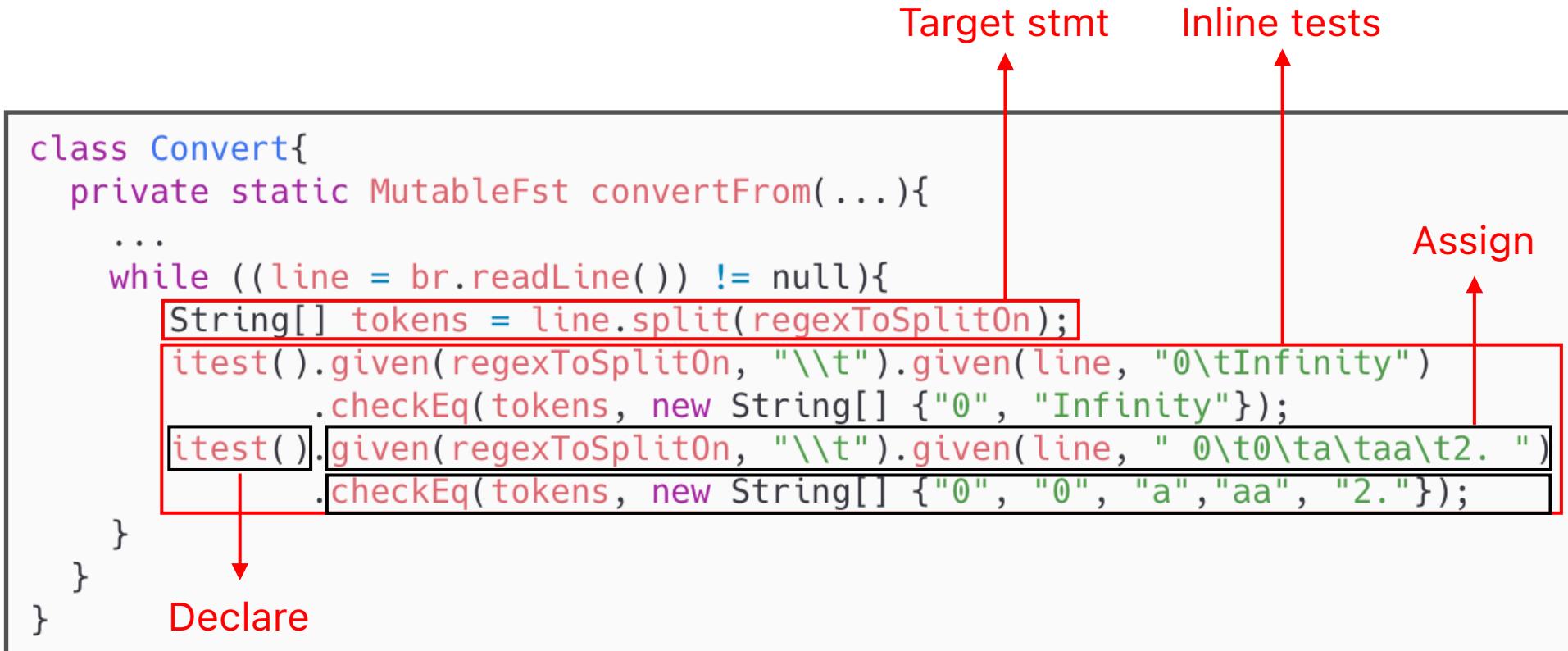
Inline Tests

- We recently proposed **Inline tests**[1] that introduce **new granularity of tests** for checking individual program statements



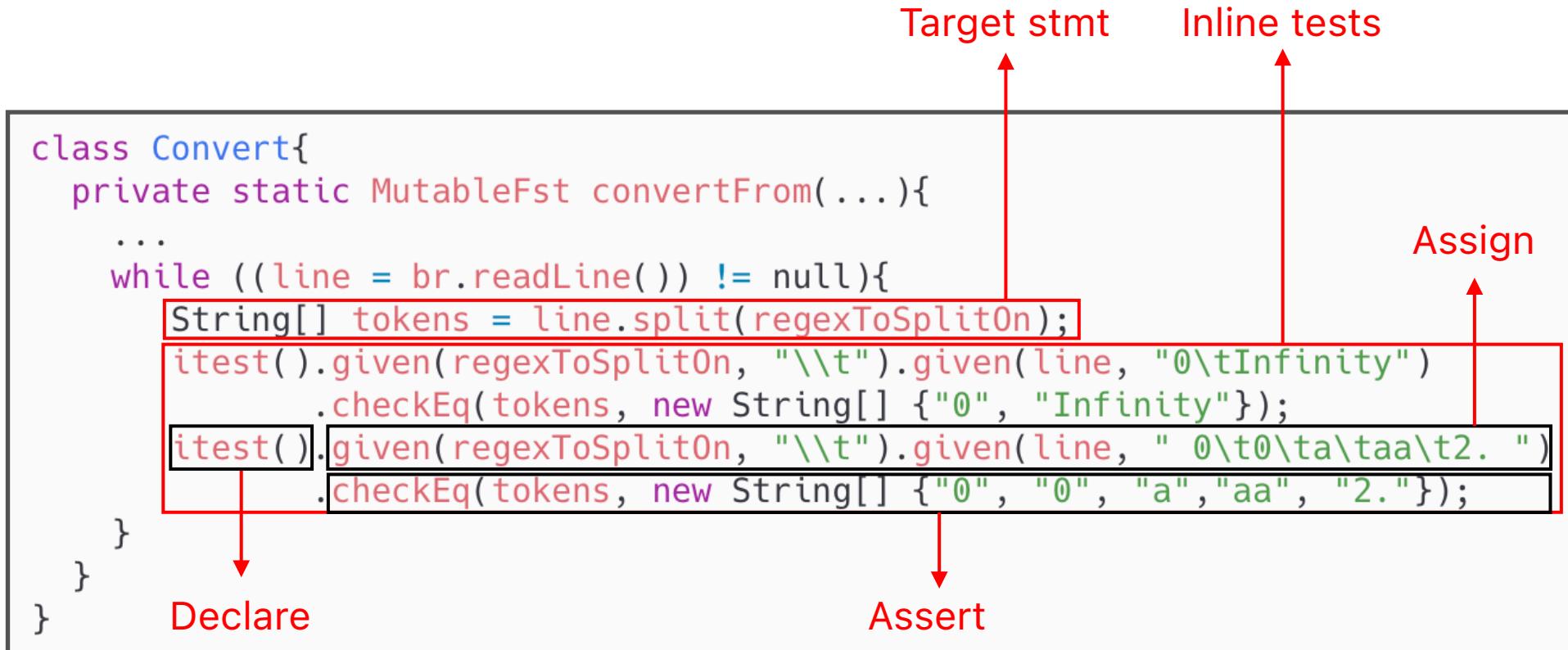
Inline Tests

- We recently proposed **Inline tests**[1] that introduce **new granularity of tests** for checking individual program statements



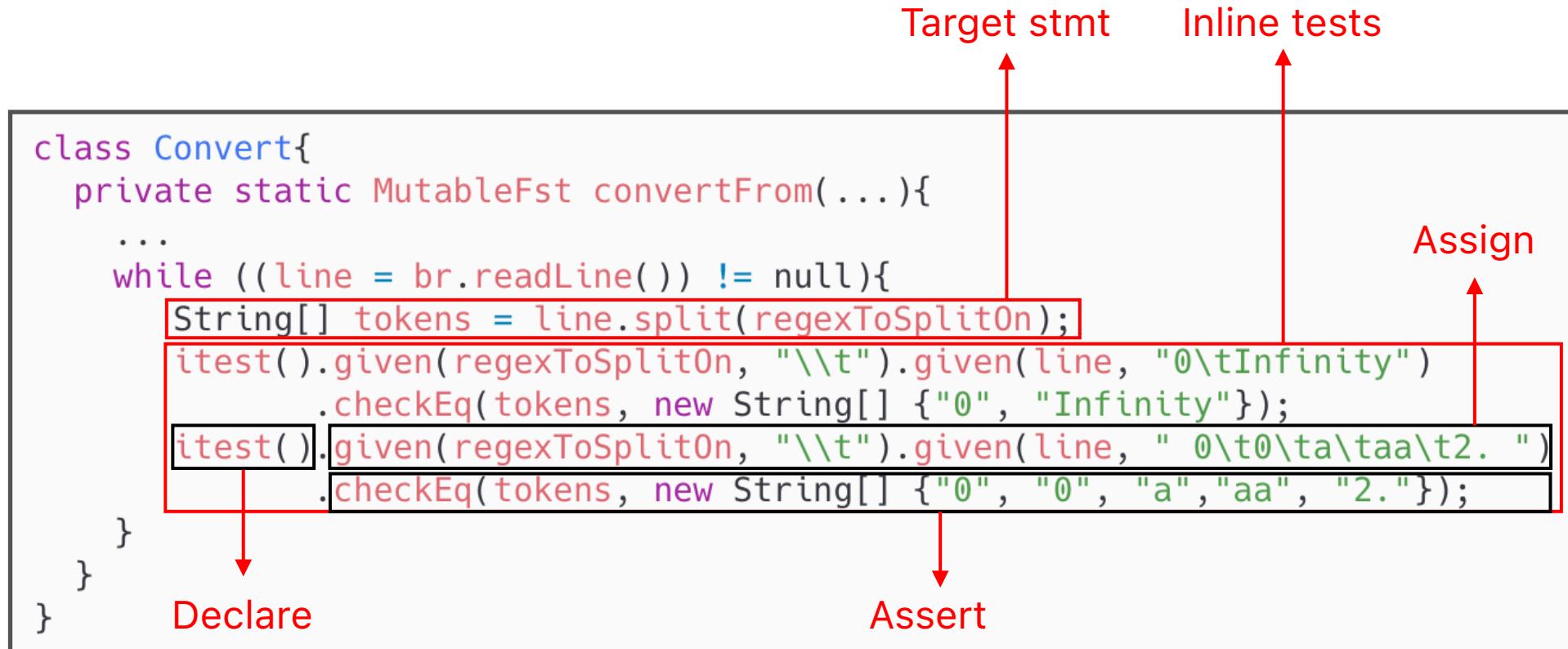
Inline Tests

- We recently proposed **Inline tests**[1] that introduce **new granularity of tests** for checking individual program statements



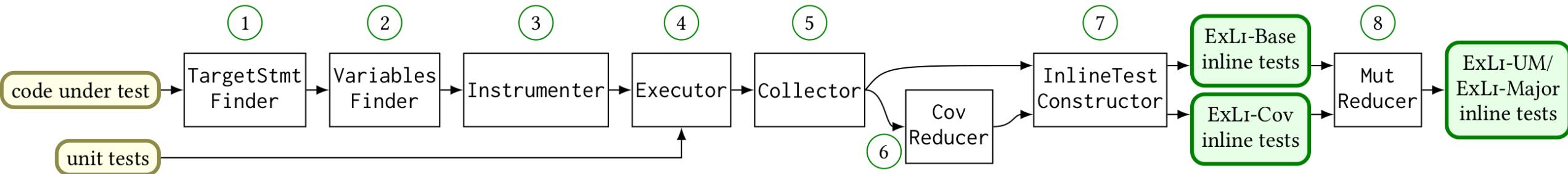
Inline Tests

- We recently proposed **Inline tests**[1] that introduce **new granularity of tests** for checking individual program statements

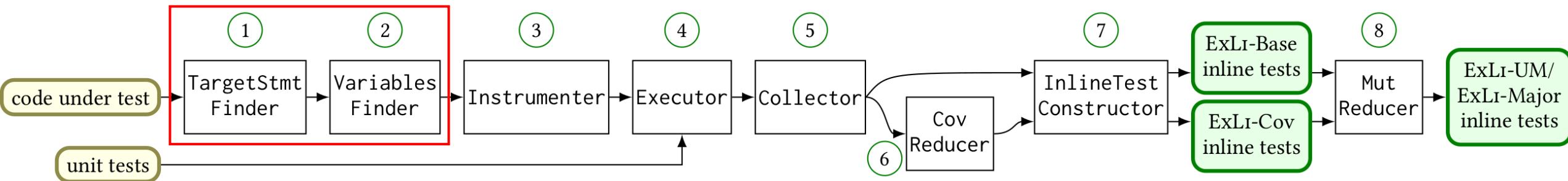


Can we automatically extract inline tests from unit tests?

ExLi: Extracting Inline Tests from Unit Tests

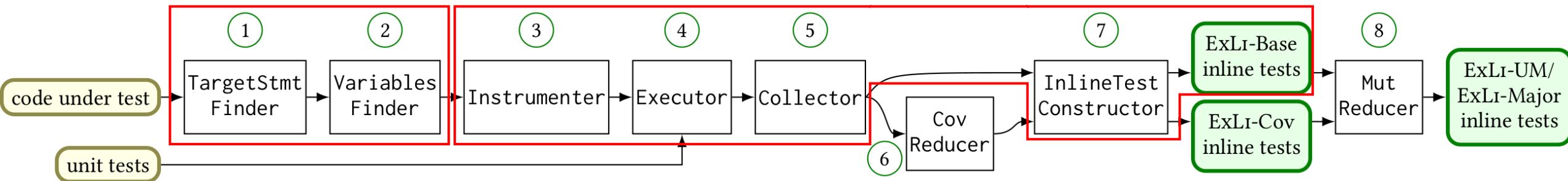


ExLi: Extracting Inline Tests from Unit Tests



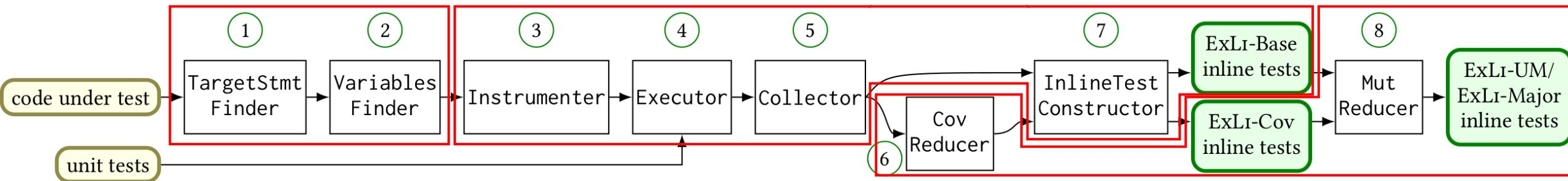
- Finding and analyzing target statements

ExLi: Extracting Inline Tests from Unit Tests



- Finding and analyzing target statements
- Generating inline tests

ExLi: Extracting Inline Tests from Unit Tests



- Finding and analyzing target statements
- Generating inline tests
- Reducing inline tests using **coverage-then-mutants**-based algorithm

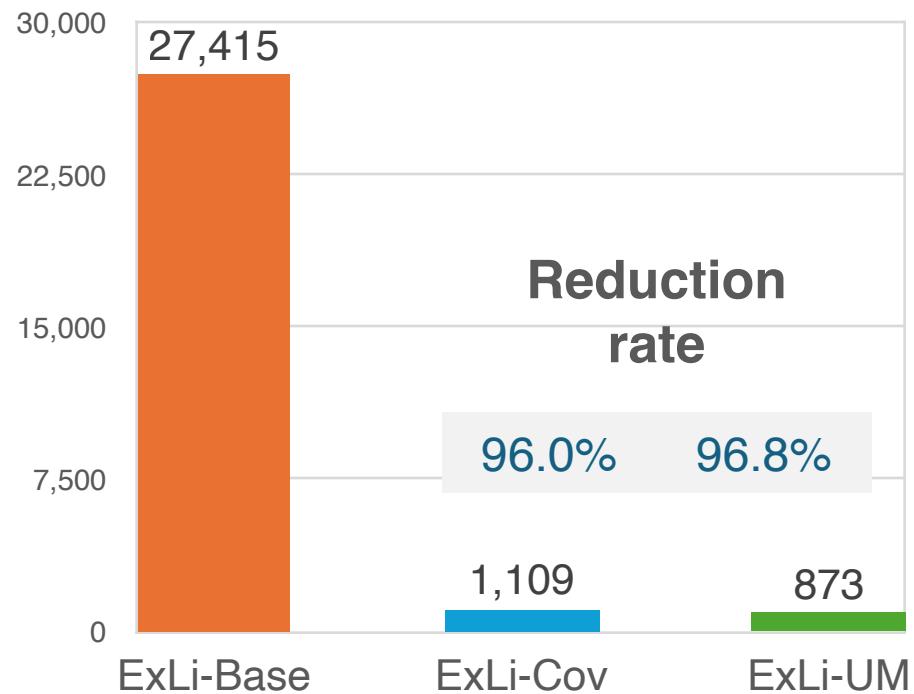
For more details, please refer to the paper

Evaluation Setup

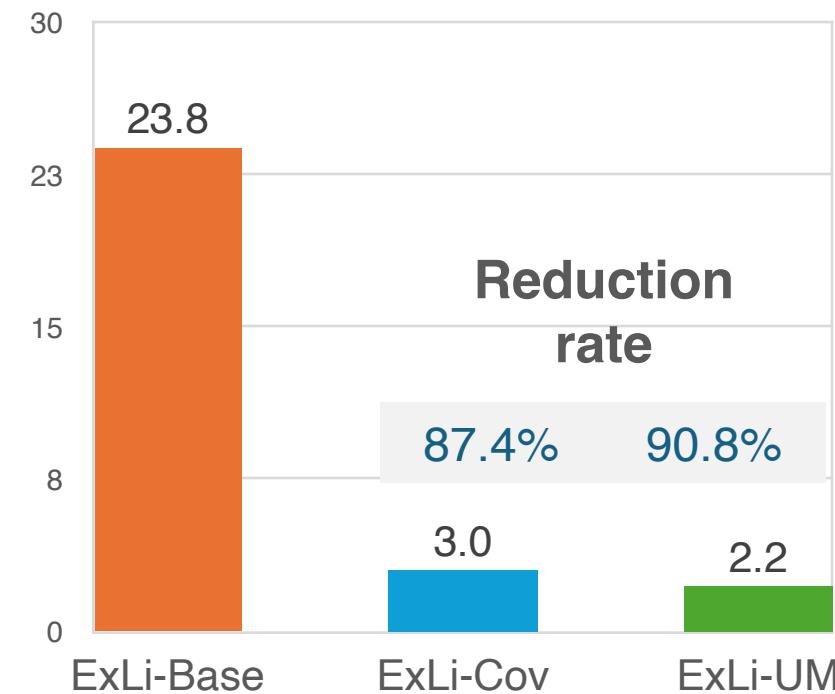
- Dataset: 31 Java projects with 423K LOC
- Extracted inline tests from 237K unit tests for 645 target statements
 - 11K developer-written
 - 215K Randoop-generated
 - 11K EvoSuite-generated
- Research questions
 - RQ1: Target statements
 - RQ2: Inline tests
 - RQ3: Mutation analysis

RQ2 Inline Tests

#inline tests



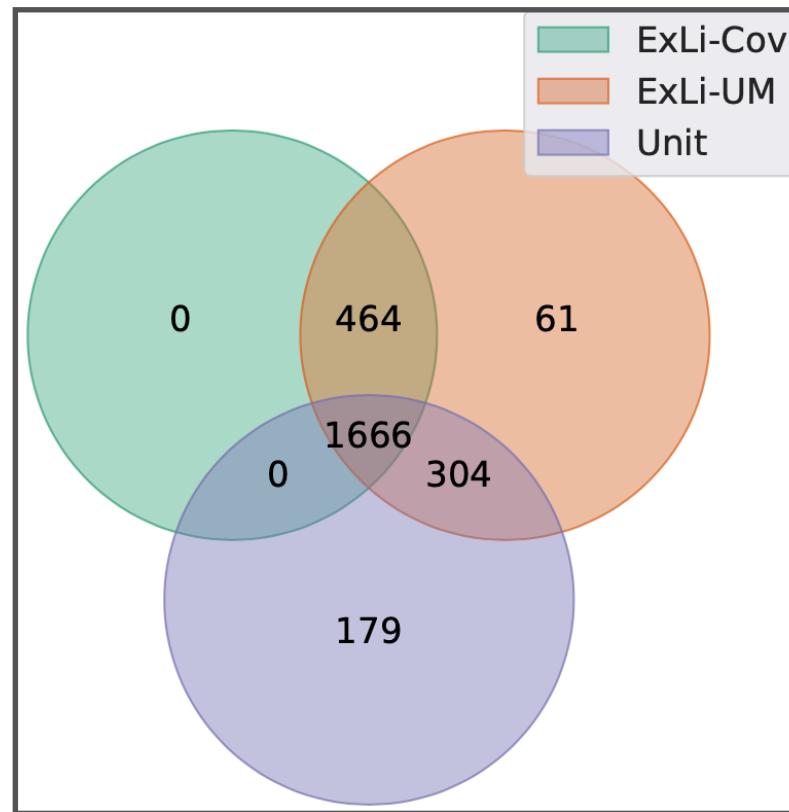
inline testing time [s]



	ExLi-Base	ExLi-Cov	ExLi-UM
avg	42.5	1.7	1.4
median	29.0	1.0	1.0

#inline tests / statement

RQ3 Mutation Analysis



Conclusion

- Evaluated ExLi on 31 Java projects with 423K LOC
- Extracted inline tests from 237K unit tests for 645 target statements
- ExLi automatically generates inline tests from unit tests
- ExLi reduces redundancy in extracted inline tests while preserving fault detection capability

ExLi : <https://github.com/EngineeringSoftware/exli>

Contact : Aditya Thimmaiah (auditt@utexas.edu)

Appendix

RQ1 Target Statements

