

# Operias: Visualizing Code and Coverage Changes for Code Review

Sebastiaan Oosterwaal, Arie van Deursen, Roberta Coelho, *Anand Ashok Sawant* and Alberto Bacchelli

# Code Review

Lack of coverage information

A new piece of code may go completely untested

A change may reduce the code coverage





Operias

| Name      | Line coverage                            | # Relevant lines | Condition coverage                        | # Conditions | Source Changes |
|-----------|--|------------------|---|--------------|----------------|
| org.junit | <div><div></div><div></div></div> -5.3%  | +16 (6.56%)      | <div><div></div><div></div></div> -7.36%  | +5 (10.64%)  |                |
| Assert    | <div><div></div><div></div></div> -7.41% | +16 (8.99%)      | <div><div></div><div></div></div> -13.57% | +5 (16.67%)  | +35 (3.72%)    |

## Test Classes

| Name   | Amount of lines changed |
|--|-------------------------|
| /src/test/java/org/junit/tests/utilityclass/MultipleConstructorUtil.java | 11 (New)                |
| /src/test/java/org/junit/tests/utilityclass/NonFinalUtil.java            | 7 (New)                 |
| /src/test/java/org/junit/tests/utilityclass/ProperUtil.java              | 11 (New)                |
| /src/test/java/org/junit/tests/utilityclass/PublicConstructorUtil.java   | 12 (New)                |
| /src/test/java/org/junit/tests/utilityclass/UtilityClassTest.java        | 39 (New)                |

```
65 72
73     private Plugin[] constructPlugins(Class<?> klass)
74         throws InitializationError {
75         Plug annotation = klass.getAnnotation(Plug.class);
76         if (annotation == null) {
77             return null;
78         }
79
80         Class<? extends Plugin>[] pluginClasses = annotation.value();
81         if (pluginClasses == null || pluginClasses.length == 0) {
82             return null;
83         }
84
85         Plugin[] plugins = new Plugin[pluginClasses.length];
86         for (int i = 0; i < pluginClasses.length; ++i) {
87             Class<? extends Plugin> pluginClass = pluginClasses[i];
88             Plugin plugin;
89             try {
90                 plugin = pluginClass.newInstance();
91             } catch (ReflectiveOperationException e) {
92                 throw new InitializationError(
93                     String.format(
94                         "Plugin class %s should have a public constructor with no parameters",
95                         pluginClass.getSimpleName());
96                 );
97             }
98             plugins[i] = plugin;
99         }
100         return plugins;
101
66 102    //
```





soosterwaal commented on 22 Apr

Owner



This pull request will have the following effects on the line and condition coverage of the project:

- The line coverage increased from 59.65% to 94.96%
- The condition coverage increased from 49.76% to 87.38%

The following changes were made to the source code of the project:

- 423 (8.99%) (82 relevant) lines were added, which are line covered for 93.0%
- 107 (2.28%) (32 relevant) lines were removed, which were line covered for 6.0%

The following changes were made to the test suite of the project:

- 27 (10.27%) lines were added
- 3 (1.14%) lines were removed

[Click here](#) for a more detailed report for this pull request.

# Added utilityClassWellDefined #646



trajano wants to merge 3 commits into `junit-team:master` from `trajano:master`

💬 Conversation 3

🔗 Commits 3

📄 Files changed 6

+115 -0



trajano commented on Feb 26, 2013



Based on

[https://github.com/trajano/maven-jee6/blob/master/maven-jee6-test/src/main/java/net/trajano/maven\\_jee6/test/UtilityClassTestUtil.java](https://github.com/trajano/maven-jee6/blob/master/maven-jee6-test/src/main/java/net/trajano/maven_jee6/test/UtilityClassTestUtil.java)

and <http://stackoverflow.com/a/10872497/242042>

trajano added some commits on Feb 26, 2013

- Added utilityClassWellDefined ...
- fixup! Added utilityClassWellDefined
- Added unit tests for utilityClassWellDefined

c8c1dfe

fdcb91f

581cefc



dsaff commented on Feb 27, 2013

JUnit member



This seems like a fine and valuable third-party extension. I'd recommend releasing it yourself, and adding a link at the bottom of <https://github.com/junit-team/junit/wiki>. Thanks!



dsaff closed this on Feb 27, 2013

## Projects

None yet

## Labels

None yet

## Milestone

No milestone

## Assignees

No one assigned

## 2 participants



## Notifications

🔔 Subscribe

You're not receiving notifications from this thread.



# Added utilityClassWellDefined #646

**Closed**

trajano wants to merge 3 commits into junit-team:master from trajano:master

💬 Conversation 3

🔗 Commits 3

📄 Files changed 6

+115 -0

| Name      | Line coverage                 | # Relevant lines | Condition coverage             | # Conditions | Source Changes |
|-----------|-------------------------------|------------------|--------------------------------|--------------|----------------|
| org.junit | <div><div></div></div> -5.3%  | +16 (6.56%)      | <div><div></div></div> -7.36%  | +5 (10.64%)  |                |
| Assert    | <div><div></div></div> -7.41% | +16 (8.99%)      | <div><div></div></div> -13.57% | +5 (16.67%)  | +35 (3.72%)    |

## Test Classes

| Name   | Amount of lines changed |
|--|-------------------------|
| /src/test/java/org/junit/tests/utilityclass/MultipleConstructorUtil.java | 11 (New)                |
| /src/test/java/org/junit/tests/utilityclass/NonFinalUtil.java            | 7 (New)                 |
| /src/test/java/org/junit/tests/utilityclass/ProperUtil.java              | 11 (New)                |
| /src/test/java/org/junit/tests/utilityclass/PublicConstructorUtil.java   | 12 (New)                |
| /src/test/java/org/junit/tests/utilityclass/UtilityClassTest.java        | 39 (New)                |

adding a link at the bottom of <https://github.com/junit-team/junit/wiki>. Thanks!

dsaff closed this on Feb 27, 2013

### Notifications

🔔 Subscribe

You're not receiving notifications from this thread.



DEMO

