Operias: Visualizing Code and Coverage Changes for Code Review

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



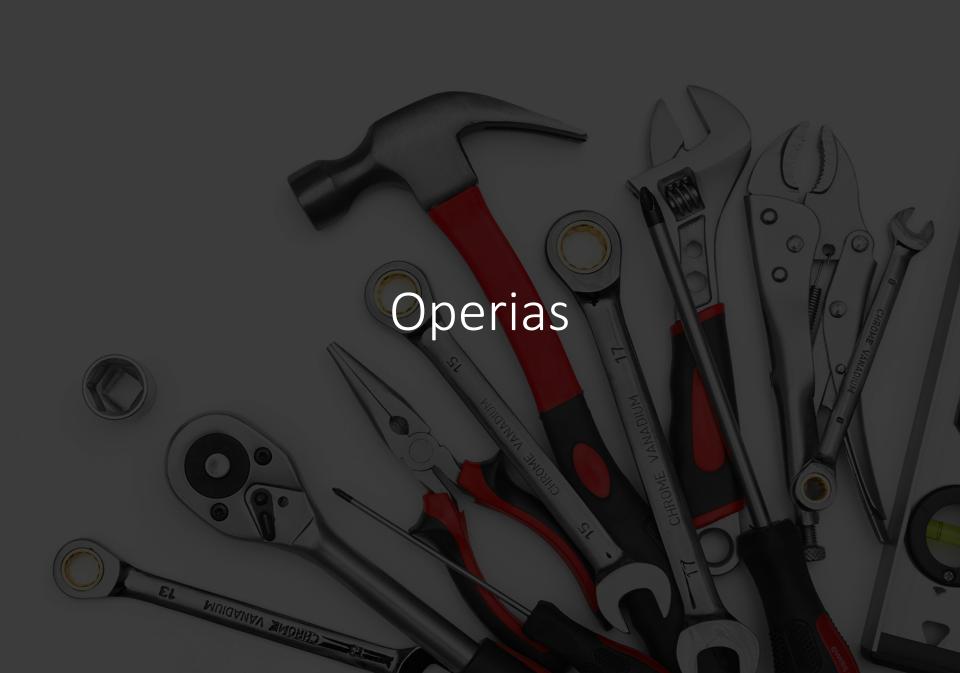
Code Review

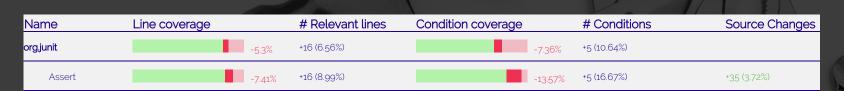
Lack of coverage information

A new piece of code may go completely untested

A change may reduce the code coverage

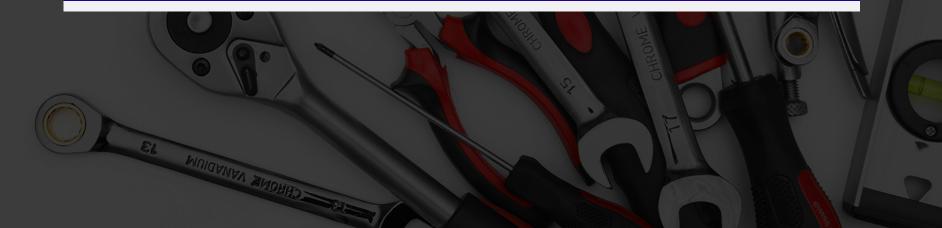




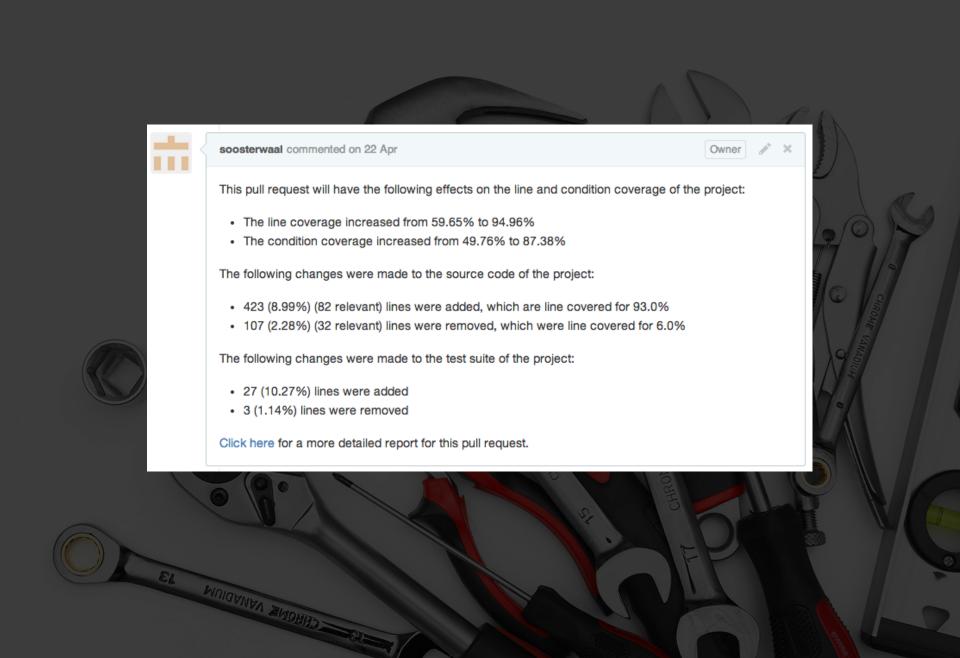


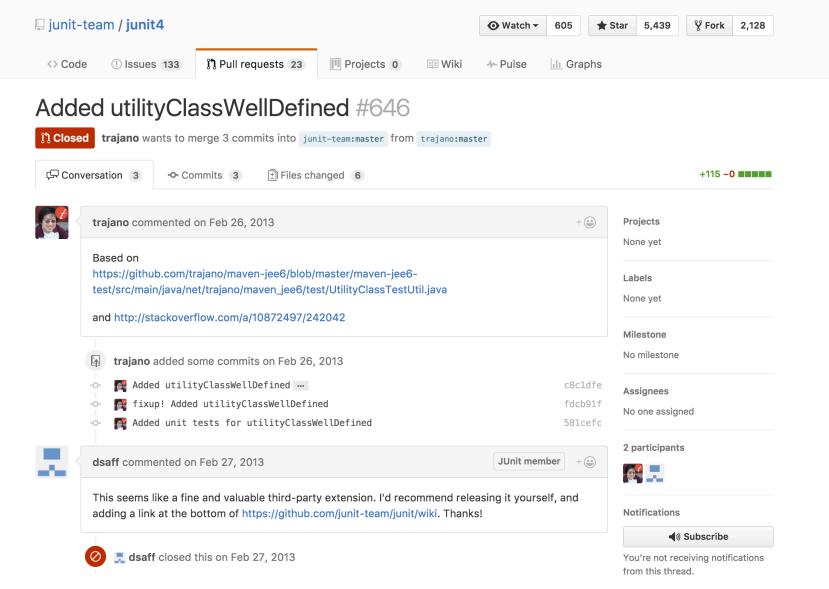
Test Classes

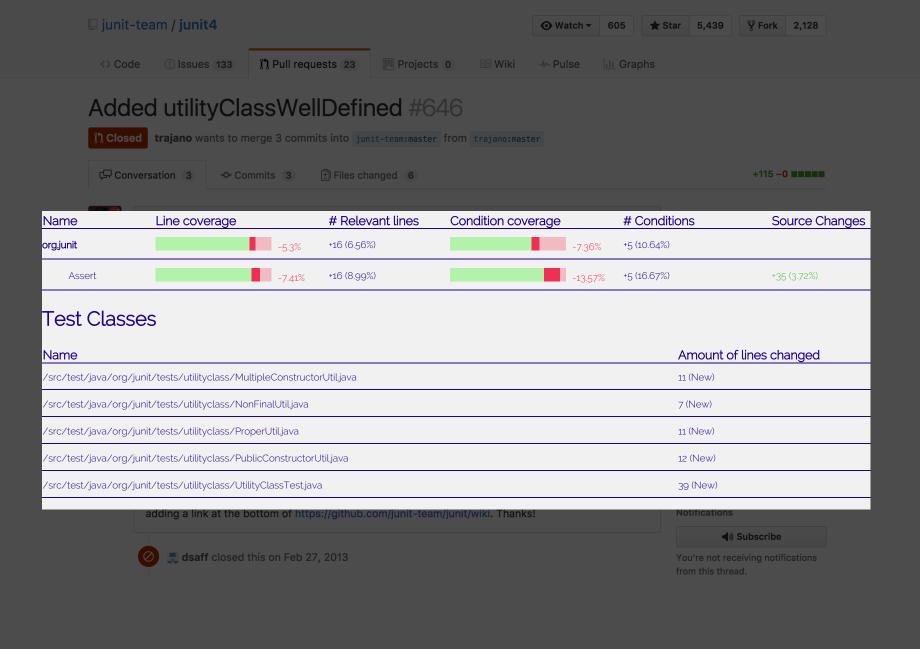
Name	Amount of lines changed
/src/test/java/org/junit/tests/utilityclass/MultipleConstructorUtil.java	11 (New)
/src/test/java/org/junit/tests/utilityclass/NonFinalUtil.java	7 (New)
/src/test/java/org/junit/tests/utilityclass/ProperUtil.java	11 (New)
/src/test/java/org/junit/tests/utilityclass/PublicConstructorUtil.java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



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







DEMO