Visualization of Code Changes Design Prototypes

Example changeset

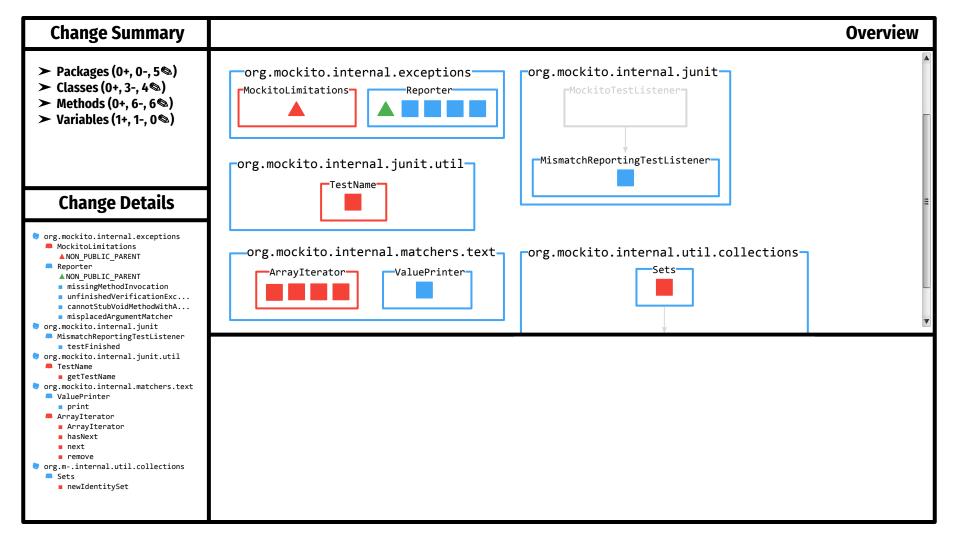
Software: Mockito, popular mocking framework for unit tests written in Java

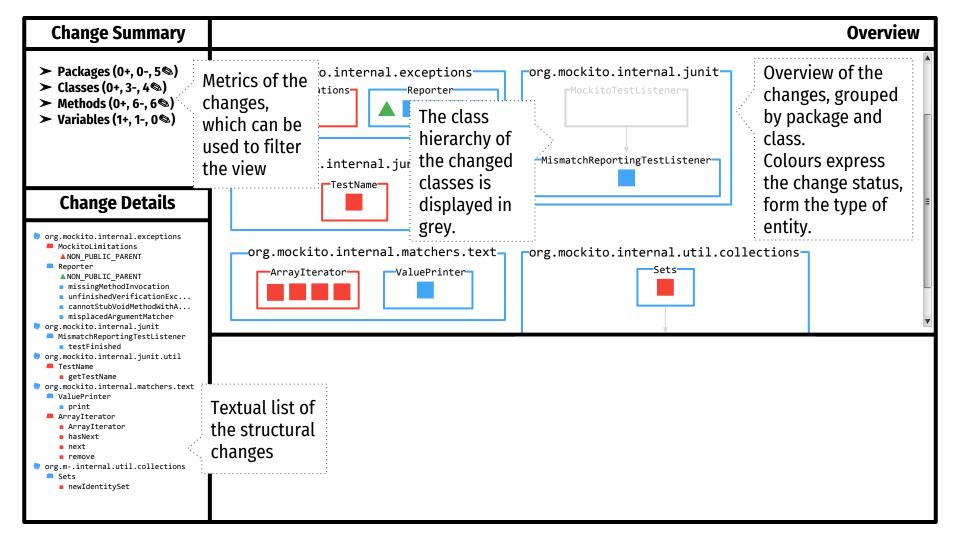
Changeset: Pull request #1247

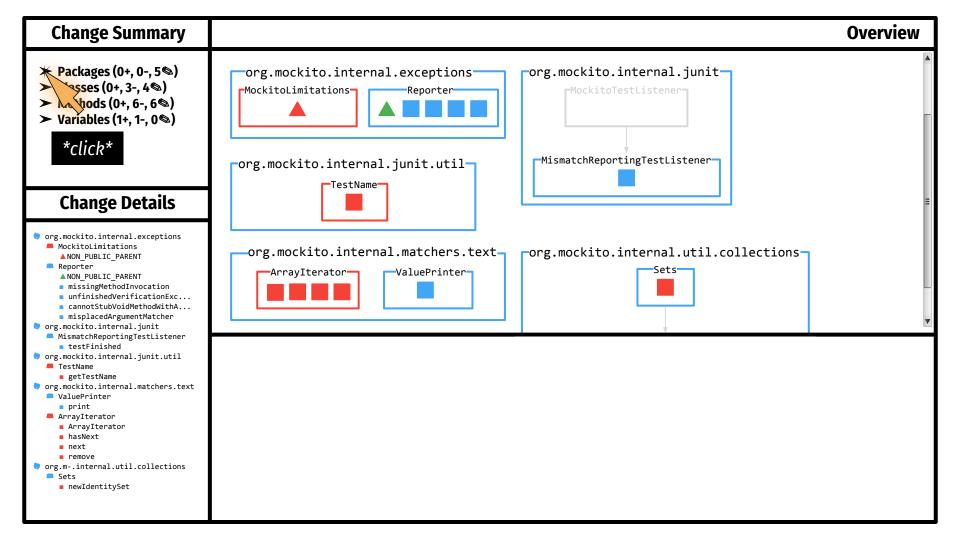
Change details:

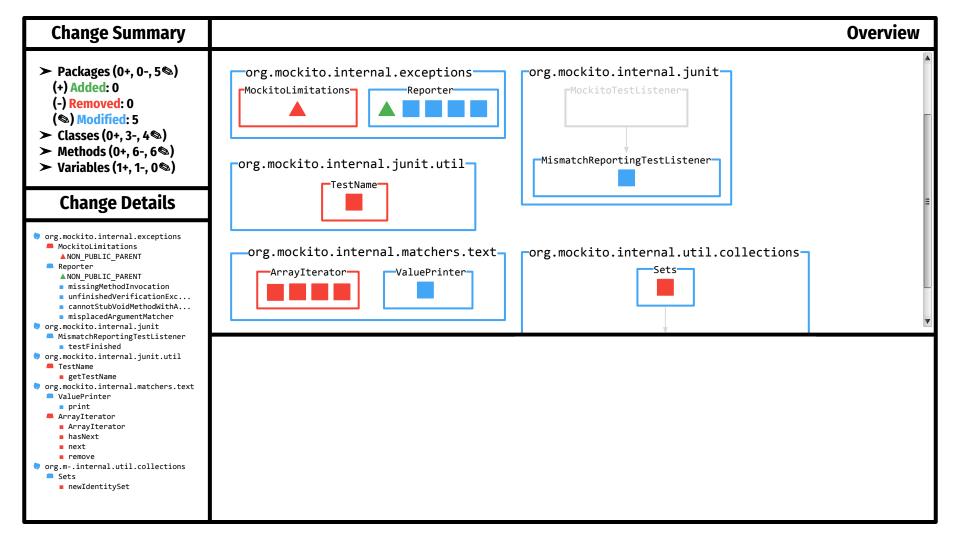
- package org.mockito.internal.exceptions modified
 - class MockitoLimitations deleted
 - class Reporter modified
 - method missingMethodInvocation modified (deleted call to MockitoLimitations)
 - method unfinishedVerificationException modified (deleted call to MockitoLimitations)
 - method cannotStubVoidMethodWithAReturnValue modified (deleted call to MockitoLimitations)
 - method misplacedArgumentMatcher modified (deleted call to MockitoLimitations)
- package org.mockito.internal.junit modified
 - class MismatchReportingTestListener modified
 - method testFinished modified (deleted call to org.mockito.internal.junit.util.TestName.getTestName)
- package org.mockito.internal.junit.util modified
 - class TestName deleted
 - method getTestName deleted
- package org.mockito.internal.matchers.text modified
 - class ArrayIterator deleted
 - method ArrayIterator deleted
 - method hasNext deleted
 - method next deleted
 - method remove deleted
 - class ValuePrinter modified
 - method print modified (deleted call to ArrayIterator)
- package org.mockito.internal.util.collections modified
 - class Sets modified
 - method newIdentitySet deleted1

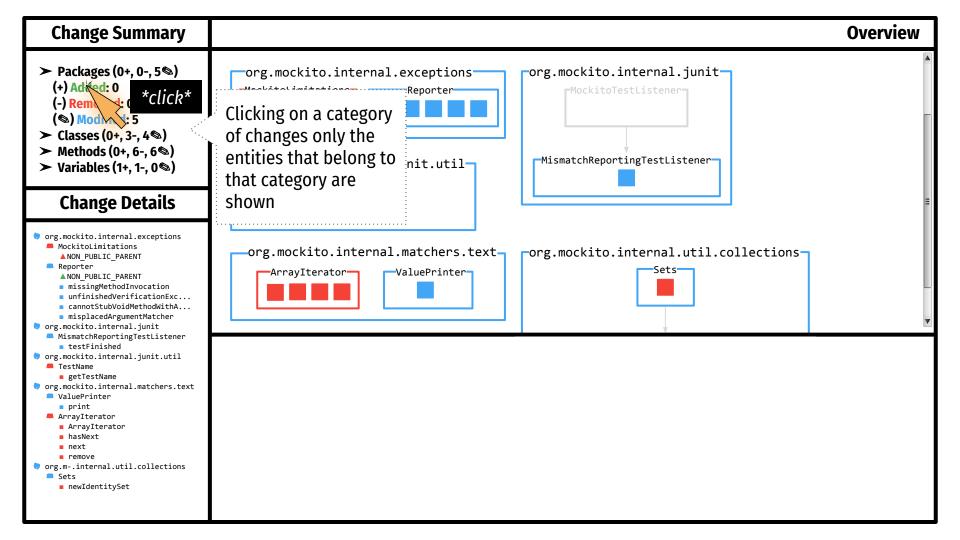
Prototype 1





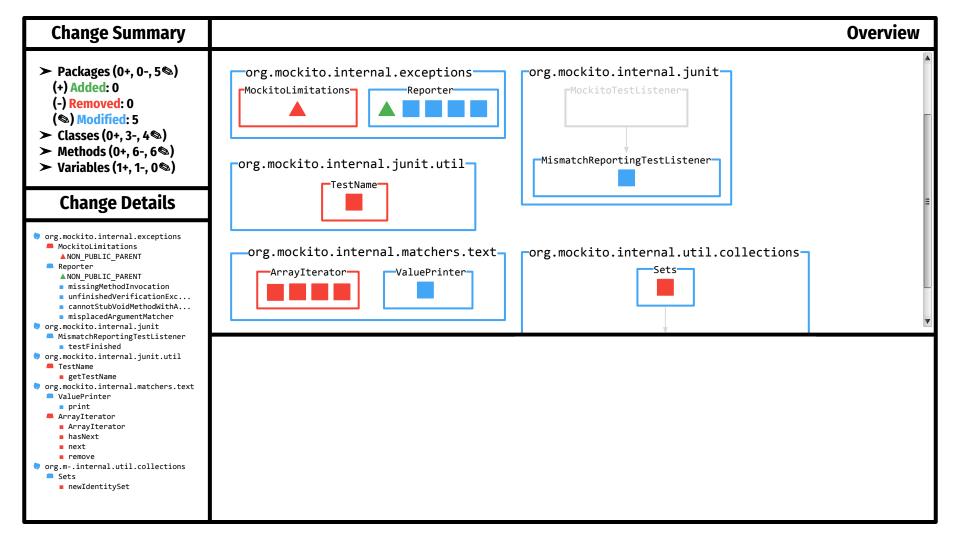


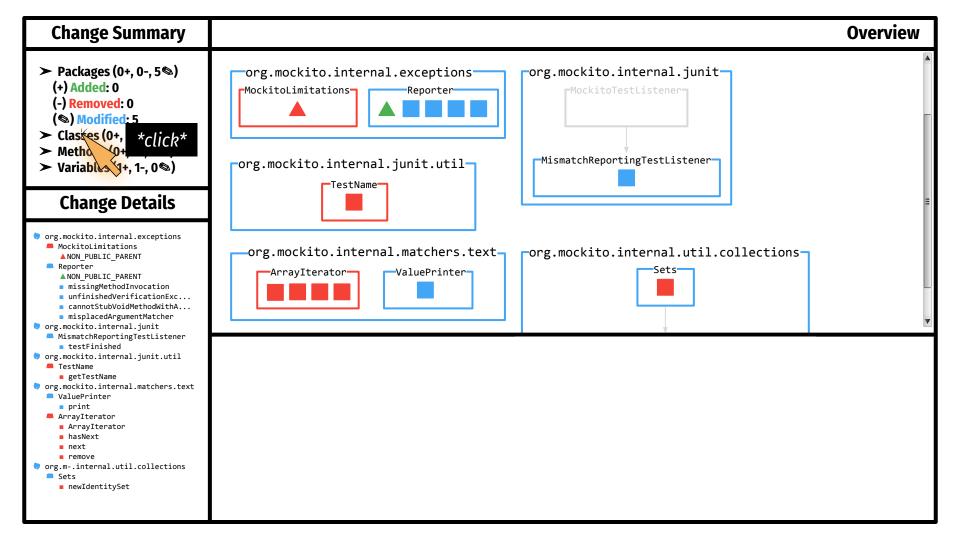


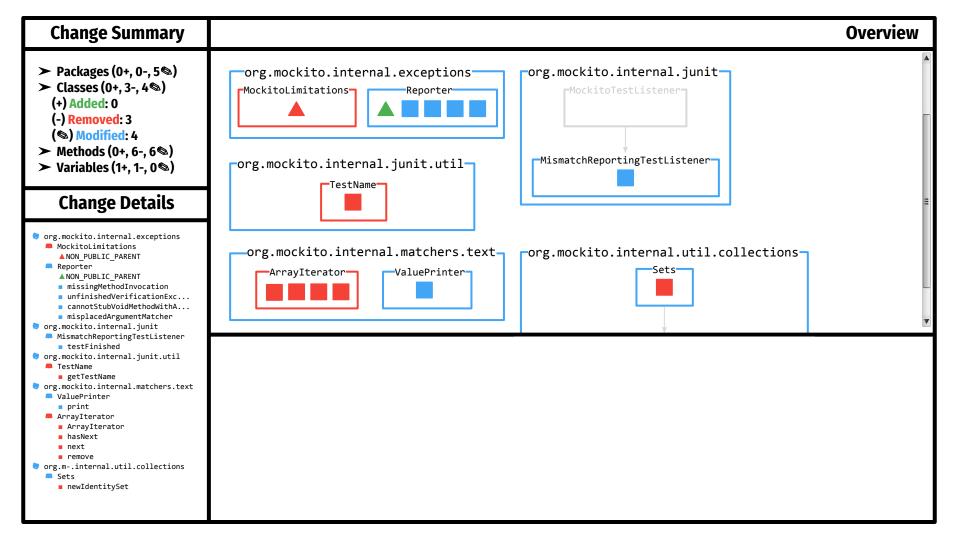


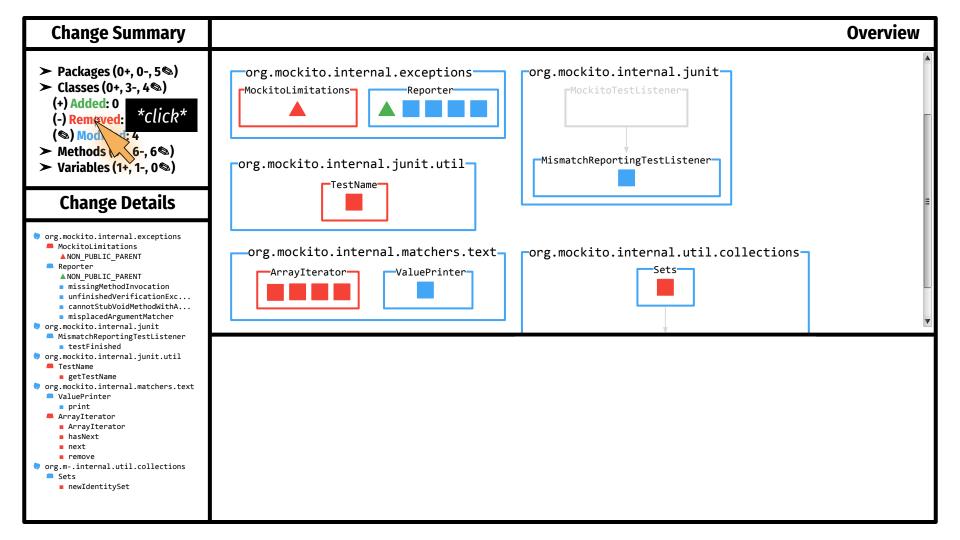
Change Summary	Overview
➤ Packages (0+, 0-, 5♠) (+) Added: 0 (-) Removed: 0 (♠) Modified: 5 ➤ Classes (0+, 3-, 4♠) ➤ Methods (0+, 6-, 6♠) ➤ Variables (1+, 1-, 0♠)	
Change Details	

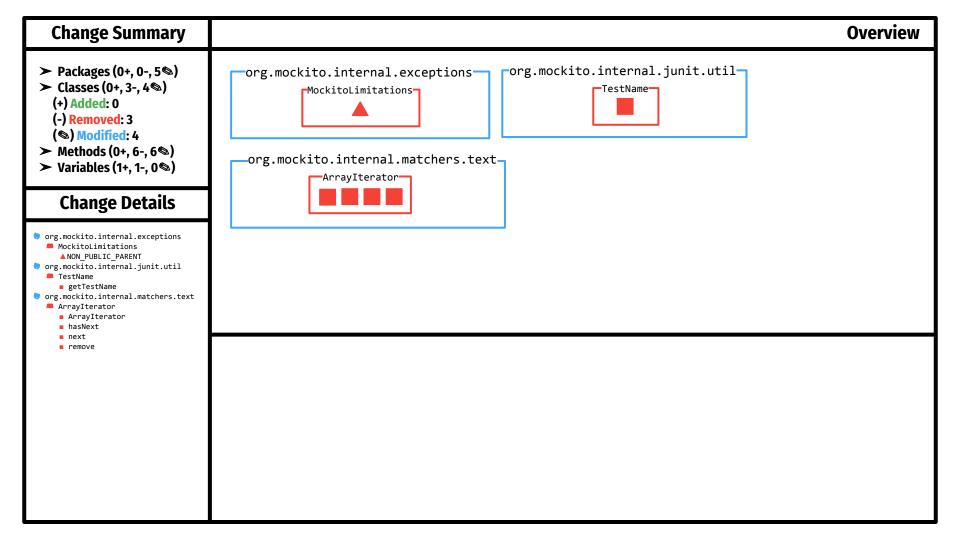
Change Summary	Overview
➤ Packages (0+, 0-, 5♠) (+) Added: 0 (-) Removed: 0 (♠) M*diffied: ➤ Classes 3 ➤ Methods 6-, 6♠) ➤ Variables (1+, 1-, 0♠)	
Change Details	

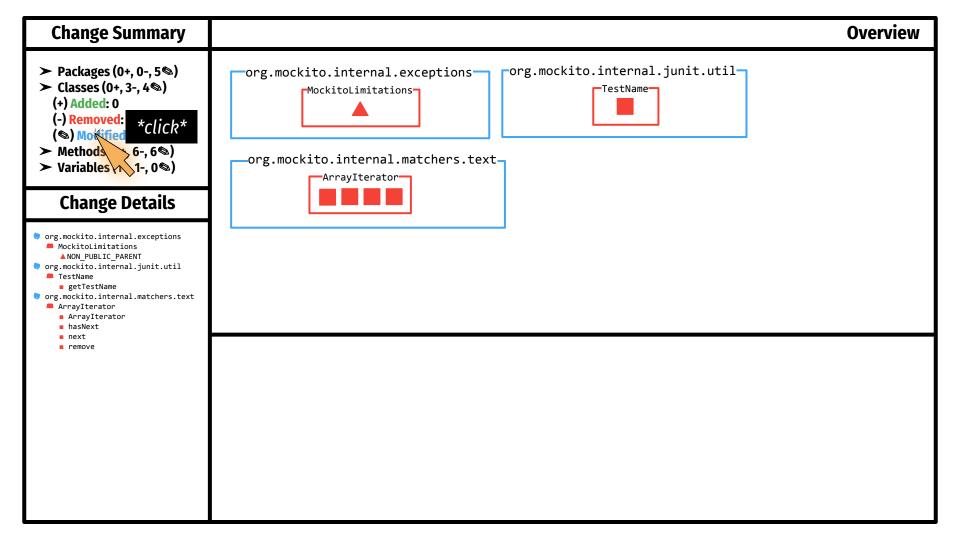


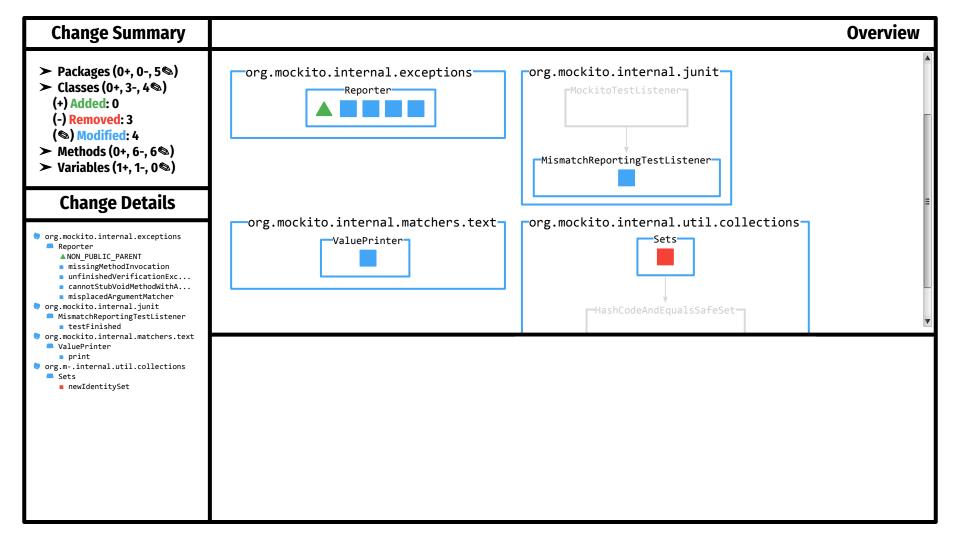


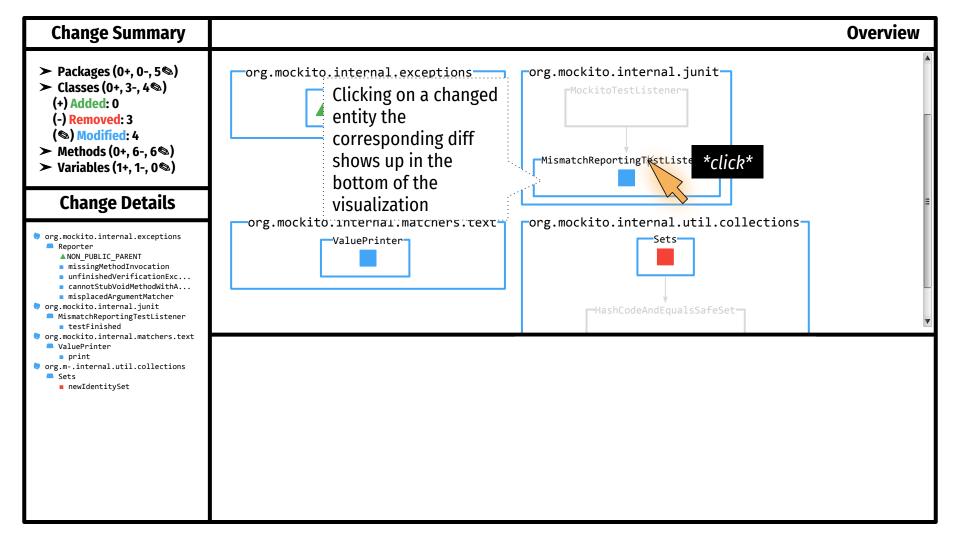




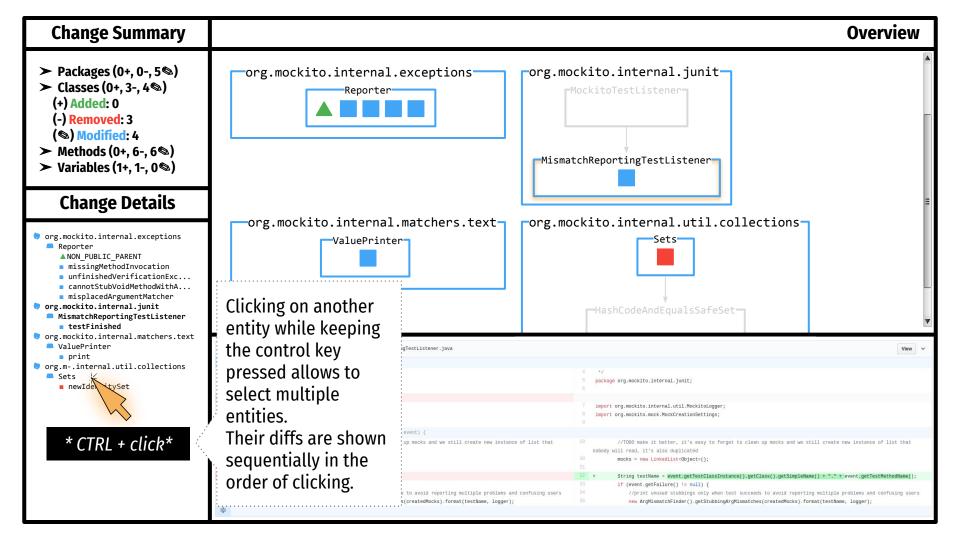








Change Summary Overview ➤ Packages (0+, 0-, 5^(a)) org.mockito.internal.exceptions rorg.mockito.internal.junit-➤ Classes (0+, 3-, 4®) Reporter (+) Added: 0 (-) Removed: 3 (®) Modified: 4 ➤ Methods (0+, 6-, 6[®]) MismatchReportingTestListener ➤ Variables (1+, 1-, 0�) **Change Details** -org.mockito.internal.matchers.textrorg.mockito.internal.util.collections org.mockito.internal.exceptions -ValuePrinter-Reporter ▲ NON PUBLIC PARENT missingMethodInvocation unfinishedVerificationExc... cannotStubVoidMethodWithA... misplacedArgumentMatcher org.mockito.internal.junit MismatchReportingTestListener testFinished org.mockito.internal.matchers.text ValuePrinter 3 src/main/java/org/mockito/internal/junit/MismatchReportingTestListener.java View print 00 -4.7 +4.6 00 org.m-.internal.util.collections Sets package org.mockito.internal.junit; package org.mockito.internal.junit; newIdentitySet -import org.mockito.internal.junit.util.TestName import org.mockito.internal.util.MockitoLogger; import org.mockito.internal.util.MockitoLogger; import org.mockito.mock.MockCreationSettings; import org.mockito.mock.MockCreationSettings; 00 -30.7 +29.7 00 public void testFinished(TestFinishedEvent event) (//TODO make it better, it's easy to forget to clean up mocks and we still create new instance of list that //TODO make it better, it's easy to forget to clean up mocks and we still create new instance of list that nobody will read, it's also duplicated nobody will read, it's also duplicated mocks = new LinkedList<Object>(); mocks = new LinkedList<Object>(); String testName = TestName.getTestName(event) String testName = event.getTestClassInstance().getClass().getSimpleName() + "." + event.getTestMethodName(); if (event.getFailure() != null) { if (event.getFailure() != null) { //print unused stubbings only when test succeeds to avoid reporting multiple problems and confusing users //print unused stubbings only when test succeeds to avoid reporting multiple problems and confusing users new ArgMismatchFinder().getStubbingArgMismatches(createdMocks).format(testName, logger); new ArgMismatchFinder().getStubbingArgMismatches(createdMocks).format(testName, logger);



Change Summary Overview ➤ Packages (0+, 0-, 5^(a)) org.mockito.internal.exceptions rorg.mockito.internal.junit ➤ Classes (0+, 3-, 4®) Reporter (+) Added: 0 (-) Removed: 3 (®) Modified: 4 ➤ Methods (0+, 6-, 6[®]) MismatchReportingTestListener ➤ Variables (1+, 1-, 0�) **Change Details** -org.mockito.internal.matchers.textrorg.mockito.internal.util.collections org.mockito.internal.exceptions -ValuePrinter-Reporter ▲ NON PUBLIC PARENT missingMethodInvocation unfinishedVerificationExc... cannotStubVoidMethodWithA... misplacedArgumentMatcher org.mockito.internal.junit MismatchReportingTestListener testFinished org.mockito.internal.matchers.text ValuePrinter 3 src/main/java/org/mockito/internal/junit/MismatchReportingTestListener.java View ~ print 00 -4,7 +4,6 00 org.m-.internal.util.collections Sets package org.mockito.internal.junit; package org.mockito.internal.junit; newIdentitySet -import org.mockito.internal.junit.util.TestName import org.mockito.internal.util.MockitoLogger; import org.mockito.internal.util.MockitoLogger; import org.mockito.mock.MockCreationSettings; import org.mockito.mock.MockCreationSettings; The other diffs are -30,7 +29,7 00 public void testFinished(TestFinishedEvent event) { //TODO make it better, it's easy to for //TODO make it better, it's easy to forget to clean up mocks and we still create new instance of list that nobody will read, it's also duplicated nobody will read, it's also duplicated accessed by scrolling mocks = new LinkedList<Object>(): mocks = new LinkedList<Object>(): String testName = TestName.getTestName(event); if (event.getFailure() != null) { if (event.getFailure() != null) { //print unused stubbings only when test succeeds to avoid reporting multiple problems and confusing users //print unused stubbings only when test succeeds to avoid reporting multiple problems and confusing users new ArgMismatchFinder().getStubbingArgMismatches(createdMocks).format(testName, logger); new ArgMismatchFinder().getStubbingArgMismatches(createdMocks).format(testName, logger);

Change Summary Overview ➤ Packages (0+, 0-, 5^(a)) rorg.mockito.internal.junit org.mockito.internal.exceptions ➤ Classes (0+, 3-, 4®) *double click* (+) Added: 0 (-) Removed: 3 (®) Modified: 4 ➤ Methods (0+, 6-, 6[®]) MismatchReportingTestListener The tool offers a ➤ Variables (1+, 1-, 0�) "callgraph view" to **Change Details** explore the callgraph org.mock rorg.mockito.internal.util.collectionsof a chosen method. org.mockito.internal.exceptions Reporter It is triggered by a ▲ NON PUBLIC PARENT missingMethodInvocation double click on a unfinishedVerificationExc... cannotStubVoidMethodWithA... misplacedArgumentMatcher class. org.mockito.internal.junit MismatchReportingTestListener testFinished org.mockito.internal.matchers.text ValuePrinter print if (event.getFailure() != null) { org.m-.internal.util.collections //print unused stubbings only when test succeeds to avoid reporting multiple problems and confusing users //print unused stubbings only when test succeeds to avoid reporting multiple problems and confusing users new ArgMismatchFinder().getStubbingArgMismatches(createdMocks).format(testName, logger); new ArgMismatchFinder().getStubbingArgMismatches(createdMocks).format(testName, logger); Sets newIdentitySet 4 **** src/main/java/org/mockito/internal/util/collections/Sets.java return HashCodeAndEqualsSafeSet.of(mocks); return HashCodeAndEqualsSafeSet.of(mocks); public static IdentitySet newIdentitySet() { return new IdentitySet(); public static <T> Set<T> newSet(T ... elements) { public static <T> Set<T> newSet(T ... elements) { if (elements == null) { if (elements == null) { throw new IllegalArgumentException("Expected an array of elements (or empty array) but received a null."); throw new IllegalArgumentException("Expected an array of elements (or empty array) but received a null.");

org.mockito.internal.exceptions.Reporter **Changed methods (4) →** missingMethodInvocation Callees org.mockito.internal.exceptions.MockitoLimitations org.mockito.exceptions.misusing.MissingMethodInvocationException org.mockito.internal.MockitoCore ■ when(T methodCall) • unfinishedVerificationException org.mockito.internal.exceptions.MockitoLimitations org.mockito.exceptions.misusing.UnfinishedVerificationException org.mockito.internal.progress.MockingProgressImpl ■ validateMostStuff() +) cannotStubVoidMethodWithAReturnValue Callees org.mockito.internal.exceptions.MockitoLimitations org.mockito.exceptions.misusing.CannotStubVoidMethodWithReturnValue org.mockito.internal.stubbing.answers **Unchanged methods (4)**

org.mockito.exceptions.misusing.MissingMethodInvocationException org.mockito.internal.exceptions.MockitoLimitations missingMethodInvocation The center of the view is one of the methods of the chosen class, which can be selected using org.mockito.internal.MockitoCore the sidebar. when(T methodCall) Arrows represent call relationships.

org.mockito.internal.exceptions.Reporter **Changed methods (4) →** missingMethodInvocation Callees org.mockito.internal.exceptions.MockitoLimitations org.mockito.exceptions.misusing.MissingMethodInvocationException org.mockito.internal.MockitoCore ■ when(T methodCall) +) unfinishedVericationException m org.mockito.interna org.mockito.exceptio rificationException org.mockito.internal.progress.mockingprogressImpl ■ validateMostStuff() +) cannotStubVoidMethodWithAReturnValue Callees org.mockito.internal.exceptions.MockitoLimitations m org.mockito.exceptions.misusing.CannotStubVoidMethodWithReturnValue org.mockito.internal.stubbing.answers **Unchanged methods (4)**

```
checkedExceptionInvalid
Callees
ororg.mockito.internal.util.StringUtil
join(Object ... linesToBreak)
Callers
org.mockito.internal.stubbing.answers
validateFor(InvocationOnMock invocation)
cannotStubWithNullThrowable
unfinishedStubbing
incorrectUseOfApi
```

< Back

S (4)

org.mockito.internal.MockitoCore
when(T methodCall)

org.mockito.internal.exceptions.MockitoLimitations

org.mockito.exceptions.misusing.MissingMethodInvocationException

missingMethodInvocation

org.mockito.internal.exceptions.Reporter Changed methods (4)

```
    missingMethodInvocation

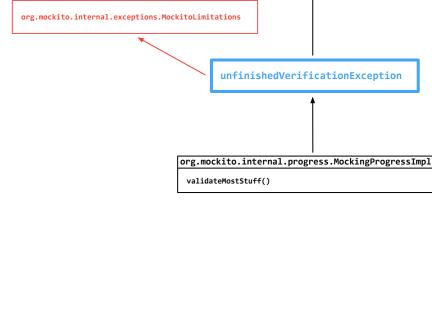
Callees
 m org.mockito.internal.exceptions.MockitoLimitations
 org.mockito.exceptions.misusing.MissingMethodInvocationException
 org.mockito.internal.MockitoCore
    ■ when(T methodCall)

→ unfinishedVerificationException

 org.mockito.internal.exceptions.MockitoLimitations
 org.mockito.exceptions.misusing.UnfinishedVerificationException
 org.mockito.internal.progress.MockingProgressImpl
    ■ validateMostStuff()
+) cannotStubVoidMethodWithAReturnValue
Callees
 org.mockito.internal.exceptions.MockitoLimitations
 m org.mockito.exceptions.misusing.CannotStubVoidMethodWithReturnValue
 org.mockito.internal.stubbing.answers
```

Unchanged methods (4)

```
checkedExceptionInvalid
callees
ororg.mockito.internal.util.StringUtil
join(Object ... linesToBreak)
callers
org.mockito.internal.stubbing.answers
validateFor(InvocationOnMock invocation)
cannotStubWithNullThrowable
unfinishedStubbing
incorrectUseOfApi
```



org.mockito.exceptions.misusing.UnfinishedVerificationException

org.mockito.internal.exceptions.Reporter

Changed methods (4)

- missingMethodInvocation
- Callees
- m org.mockito.internal.exceptions.MockitoLimitations
- org.mockito.exceptions.misusing.MissingMethodInvocationException Callers
- org.mockito.internal.MockitoCore
 - when(T methodCall)
- unfinishedVerificationException

- org.mockito.internal.exceptions.MockitoLimitations
- org.mockito.exceptions.misusing.UnfinishedVerificationException
- org.mockito.internal.progress.MockingProgressImpl ■ validateMostStuff()
- +) cannotStubVoidMethodWithAReturnValue Callees
- org.mockito.internal.exceptions.MockitoLimitations
- m org.mockito.exceptions.misusing.CannotStubVoidMethodWithReturnValue
- org.mockito.internal.stubbing.answers

Unchanged methods (4)



- validateFor(InvocationOnMock invocation)
- ⊕ cannotStubWithNullThrowable
- unfinishedStubbing
- ⊕ incorrectUseOfApi

The methods caller/callee information can be collapsed and expanded

org.mockito.exceptions.misusing.UnfinishedVerificationException org.mockito.internal.exceptions.MockitoLimitations unfinishedVerificationException org.mockito.internal.progress.MockingProgressImpl validateMostStuff()

org.mockito.internal.exceptions.Reporter

Changed methods (4)

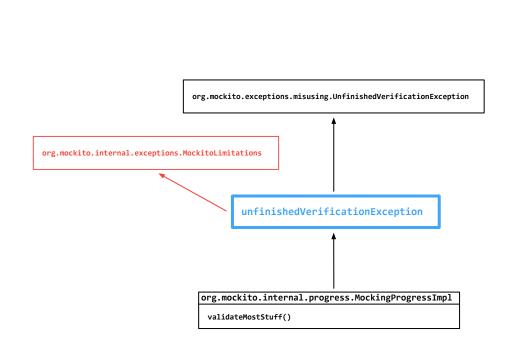
- + missingMethodInvocation
- Callees
- org.mockito.internal.exceptions.MockitoLimitations
- m org.mockito.exceptions.misusing.MissingMethodInvocationException
 Callons
- org.mockito.internal.MockitoCore
 - when(T methodCall)
- **→** unfinishedVerificationException

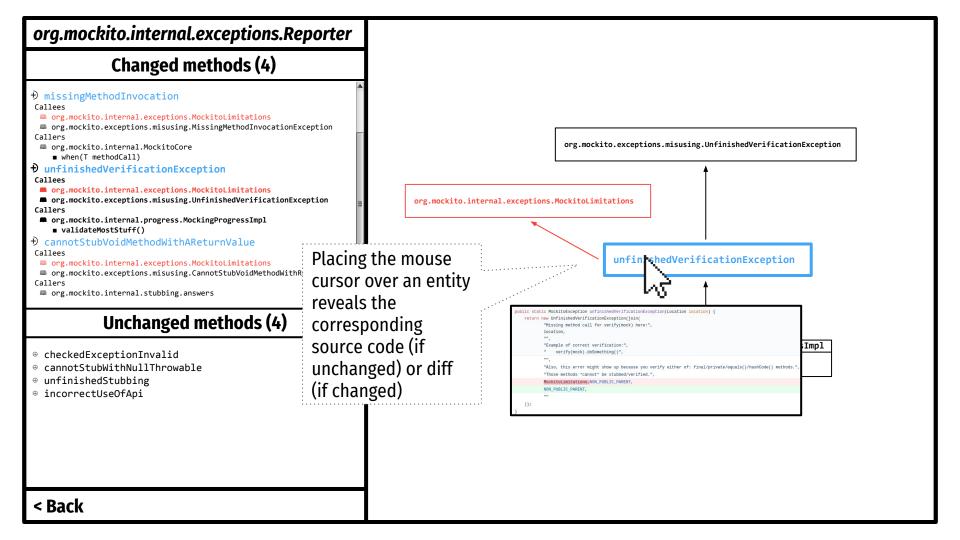
Callees

- org.mockito.internal.exceptions.MockitoLimitations
- org.mockito.exceptions.misusing.UnfinishedVerificationException
 Callers
- $\qquad \qquad \texttt{org.mockito.internal.progress.MockingProgressImpl}$
- validateMostStuff()
- → cannotStubVoidMethodWithAReturnValue
- Callee
- mockito.internal.exceptions.MockitoLimitations
- org.mockito.internal.stubbing.answers

Unchanged methods (4)

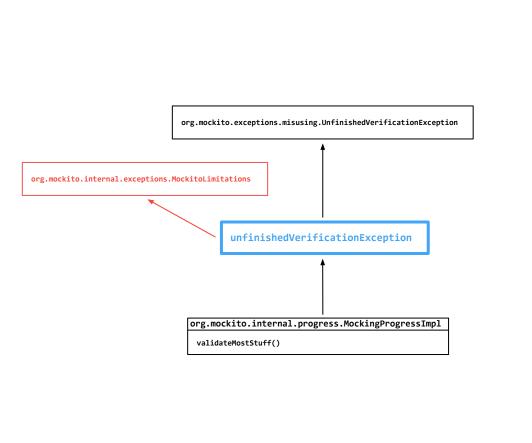
- ⊕ checkedExceptionInvalid
- ⊕ cannotStubWithNullThrowable
- ⊕ unfinishedStubbing
- ⊕ incorrectUseOfApi





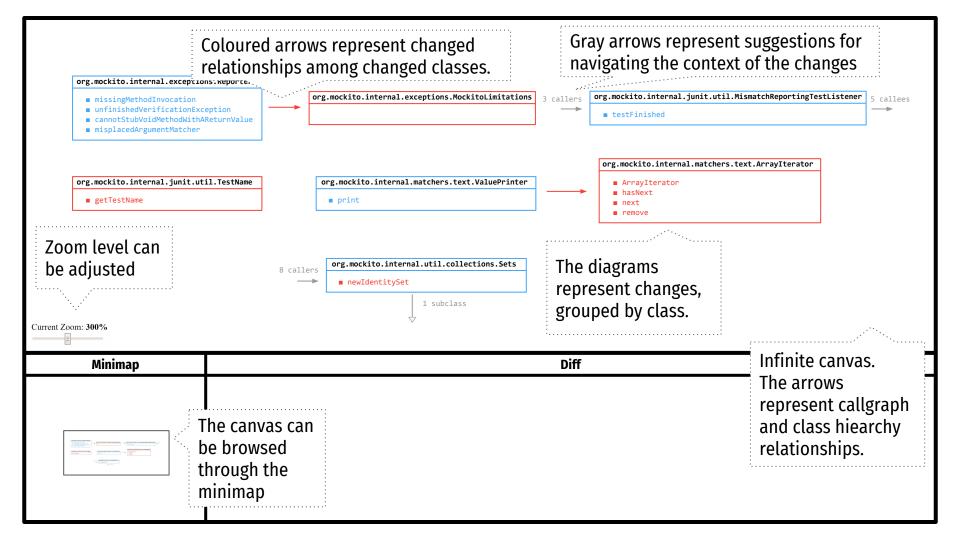
org.mockito.internal.exceptions.Reporter **Changed methods (4)** missingMethodInvocation Callees org.mockito.internal.exceptions.MockitoLimitations org.mockito.exceptions.misusing.MissingMethodInvocationException Callers org.mockito.internal.MockitoCore ■ when(T methodCall) unfinishedVerificationException org.mockito.internal.exceptions.MockitoLimitations org.mockito.exceptions.misusing.UnfinishedVerificationException org.mockito.internal.progress.MockingProgressImpl ■ validateMostStuff() +) cannotStubVoidMethodWithAReturnValue org.mockito.internal.exceptions.MockitoLimitations m org.mockito.exceptions.misusing.CannotStubVoidMethodWithReturnValue org.mockito.internal.stubbing.answers **Unchanged methods (4)** .checkedExceptionInvalid..... Pressing the "back" button we go back to the initial overview

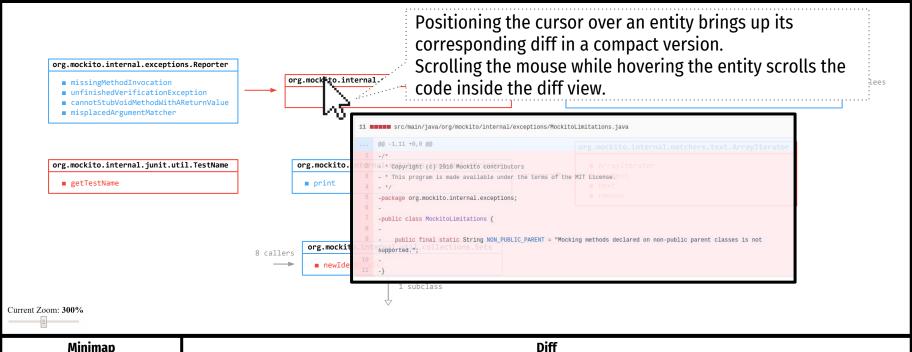
click



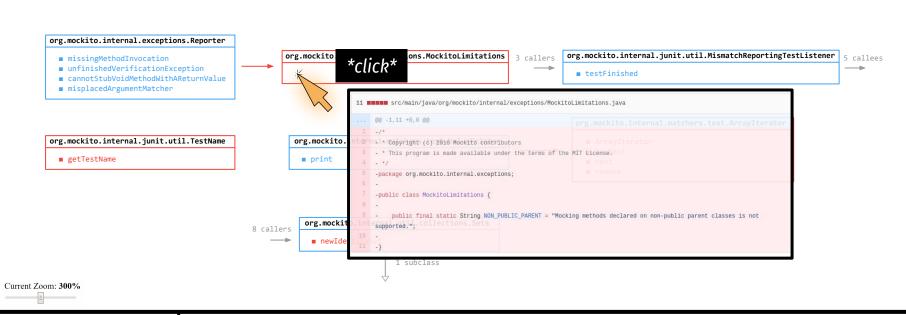
Change Summary Overview ➤ Packages (0+, 0-, 5�) org.mockito.internal.exceptions rorg.mockito.internal.junit-➤ Classes (0+, 3-, 4®) Reporter (+) Added: 0 (-) Removed: 3 (®) Modified: 4 ➤ Methods (0+, 6-, 6[®]) MismatchReportingTestListener ➤ Variables (1+, 1-, 0�) **Change Details** -org.mockito.internal.matchers.textrorg.mockito.internal.util.collections org.mockito.internal.exceptions -ValuePrinter-Reporter ▲ NON PUBLIC PARENT missingMethodInvocation unfinishedVerificationExc... cannotStubVoidMethodWithA... misplacedArgumentMatcher org.mockito.internal.junit MismatchReportingTestListener testFinished org.mockito.internal.matchers.text ValuePrinter String testName = TestName.getTestName(event) print if (event.getFailure() != null) { if (event.getFailure() != null) { org.m-.internal.util.collections //print unused stubbings only when test succeeds to avoid reporting multiple problems and confusing users //print unused stubbings only when test succeeds to avoid reporting multiple problems and confusing users new ArgMismatchFinder().getStubbingArgMismatches(createdMocks).format(testName, logger); new ArgMismatchFinder().getStubbingArgMismatches(createdMocks).format(testName, logger); Sets newIdentitySet 4 STC/main/java/org/mockito/internal/util/collections/Sets.java return HashCodeAndEqualsSafeSet.of(mocks); return HashCodeAndEqualsSafeSet.of(mocks); public static IdentitySet newIdentitySet() { return new IdentitySet(); public static <T> Set<T> newSet(T ... elements) { public static <T> Set<T> newSet(T ... elements) { if (elements == null) { if (elements == null) { throw new IllegalArqumentException("Expected an array of elements (or empty array) but received a null."); throw new IllegalArgumentException("Expected an array of elements (or empty array) but received a null.");

Prototype 2

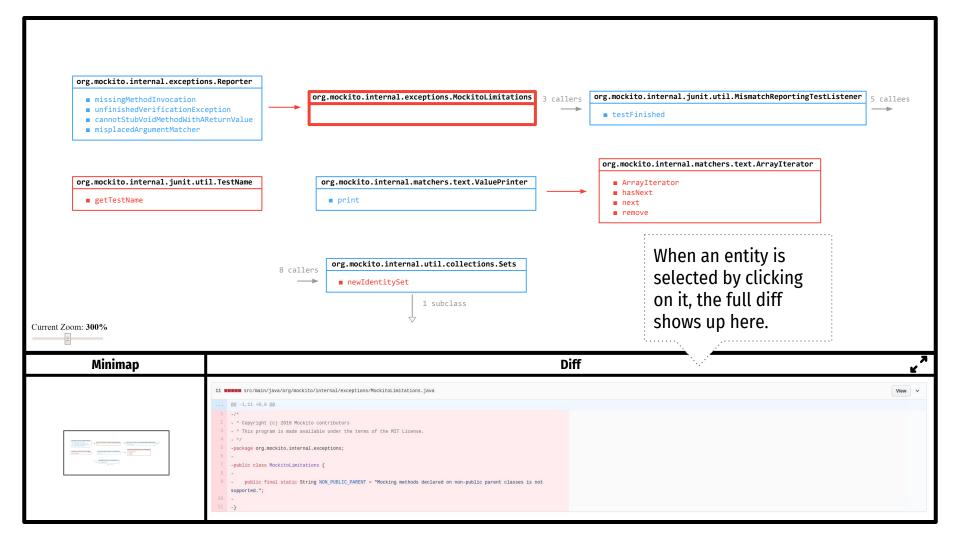


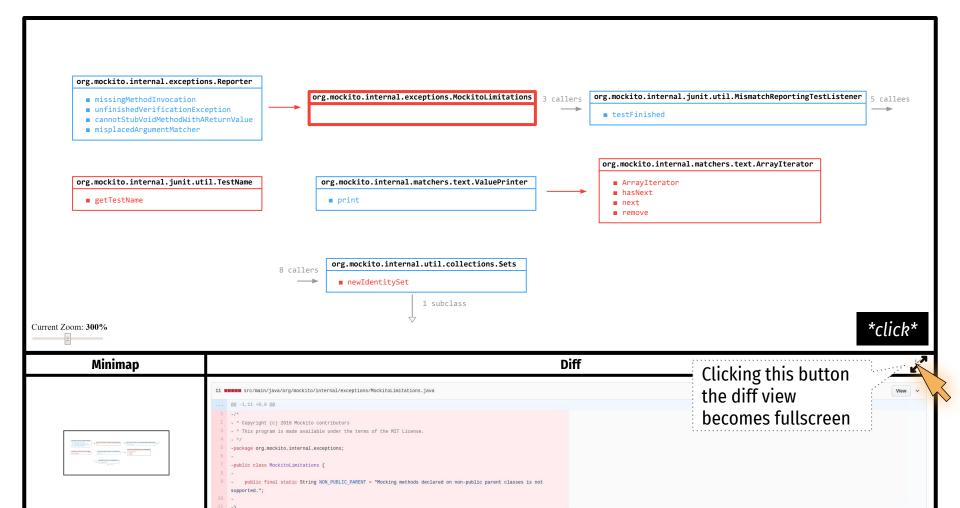


l	Minimap	Diff
ſ		
ı		
ı		
ı	AND THE PROPERTY OF THE PROPER	
ı	The second secon	
ı		
ı		
ı		



Minimap	Diff
The second sec	
Total Section (1997)	

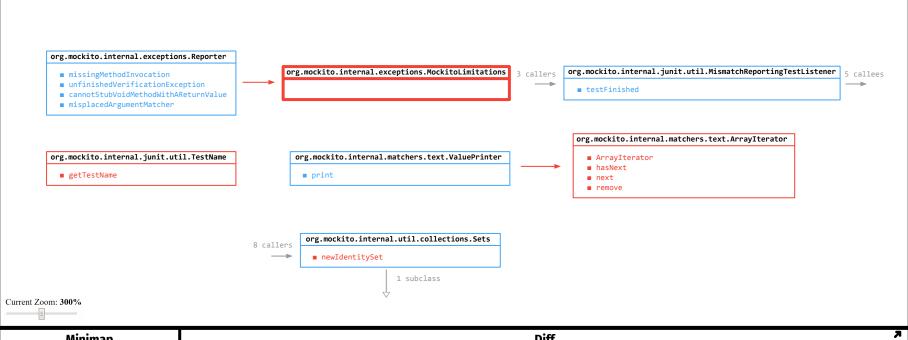




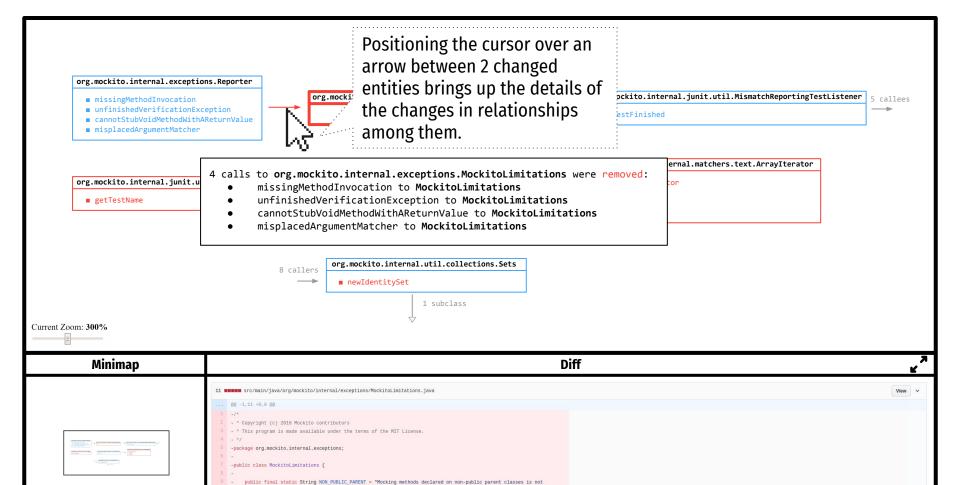


Diff





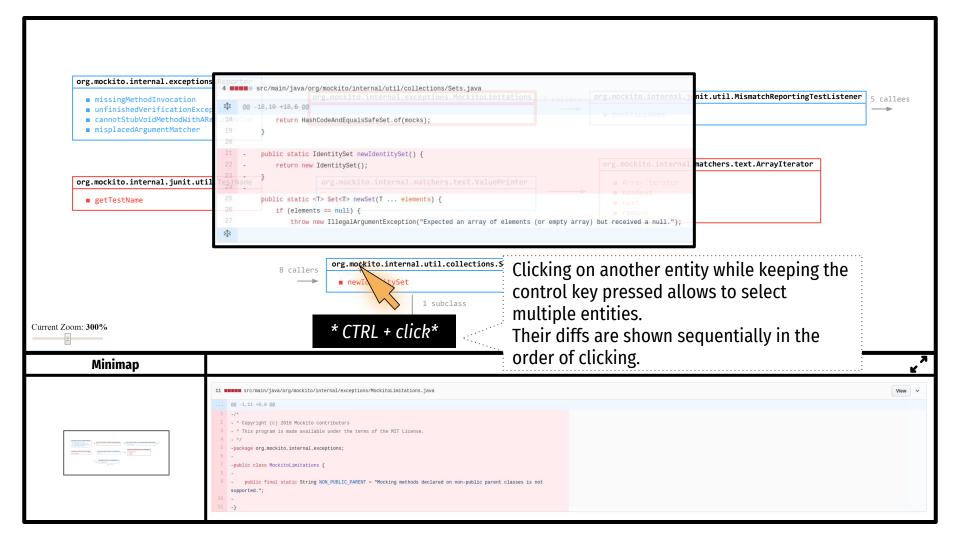


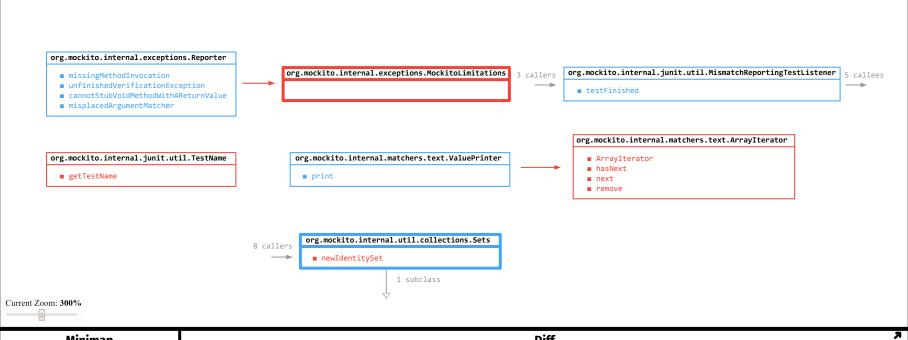


supported.":

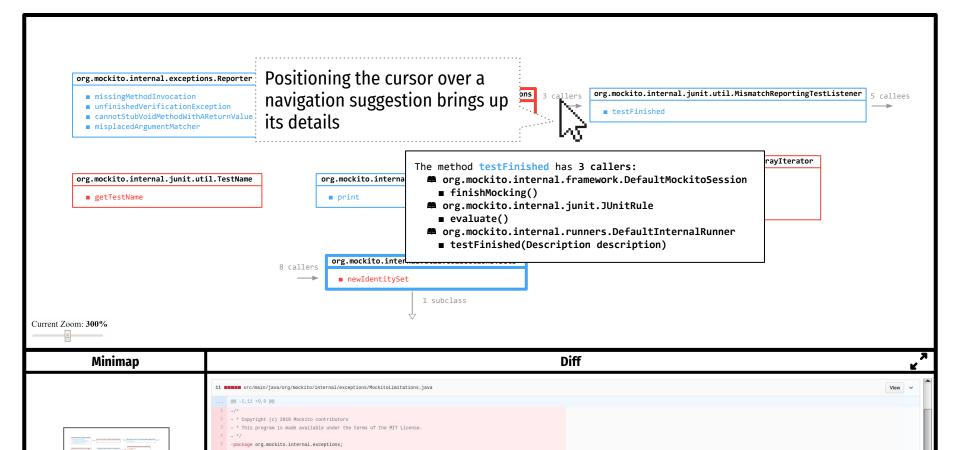








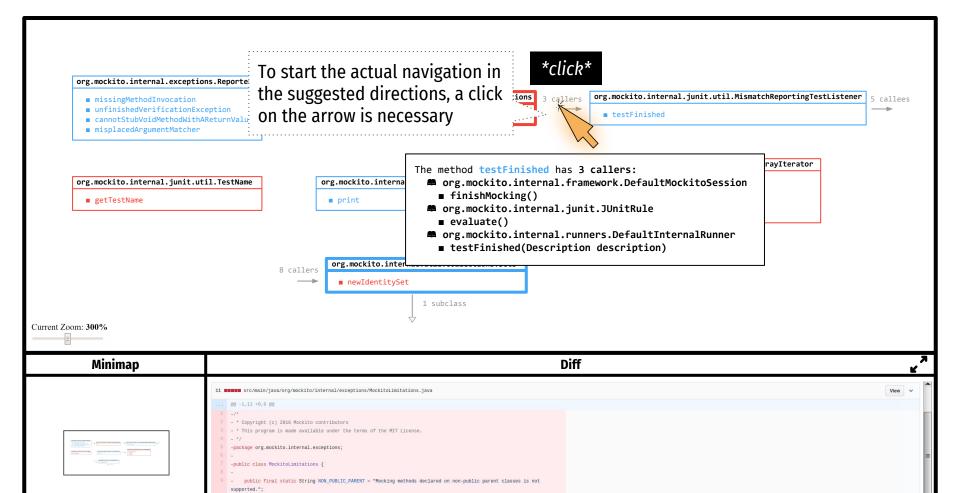


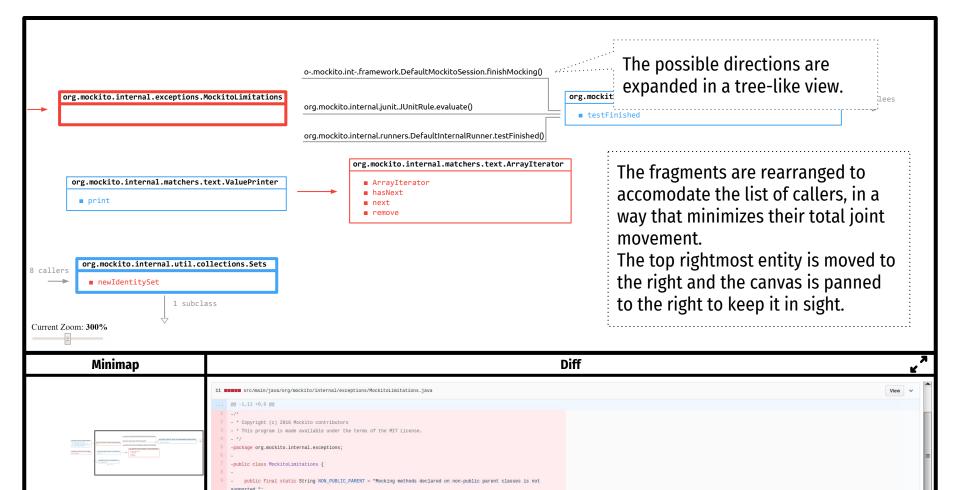


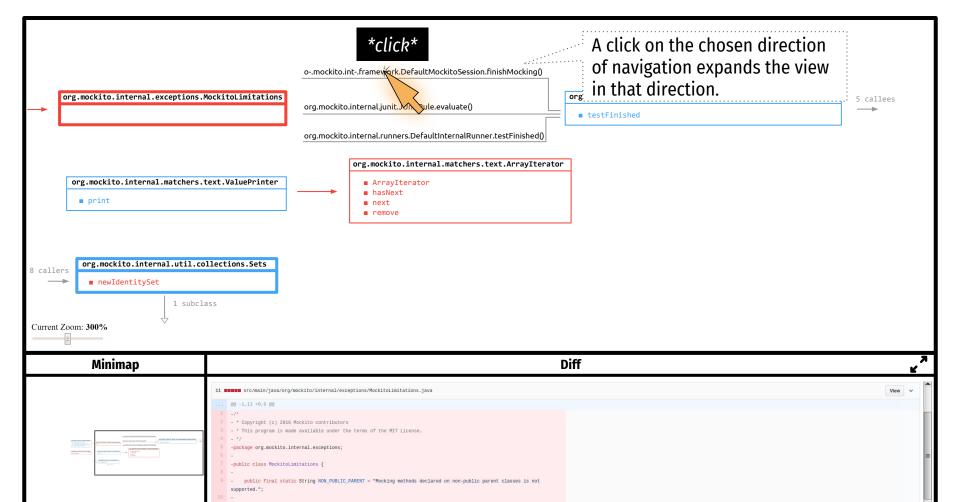
-public class MockitoLimitations {

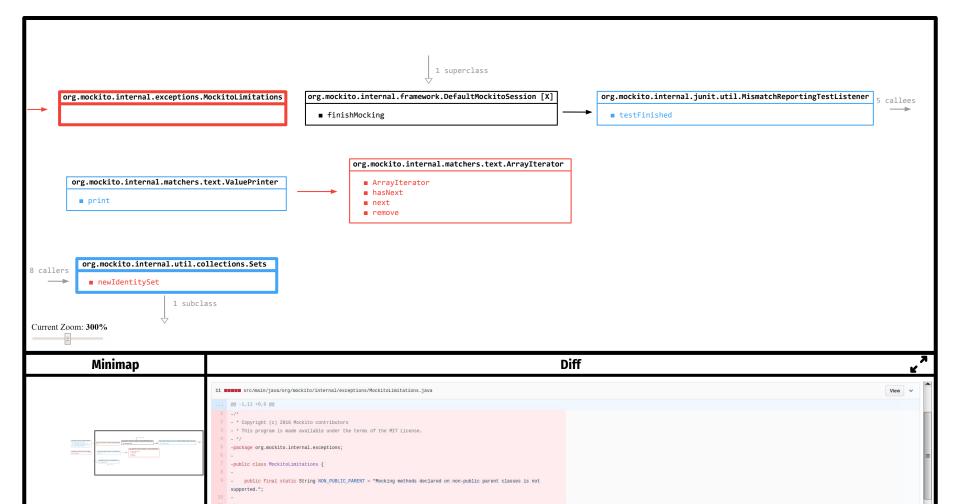
supported.";

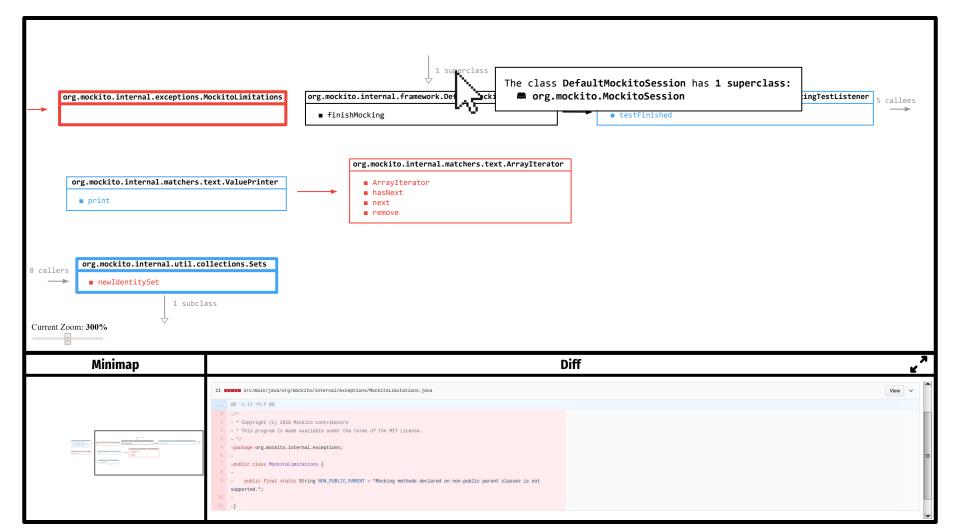
- public final static String NON PUBLIC PARENT = "Mocking methods declared on non-public parent classes is not

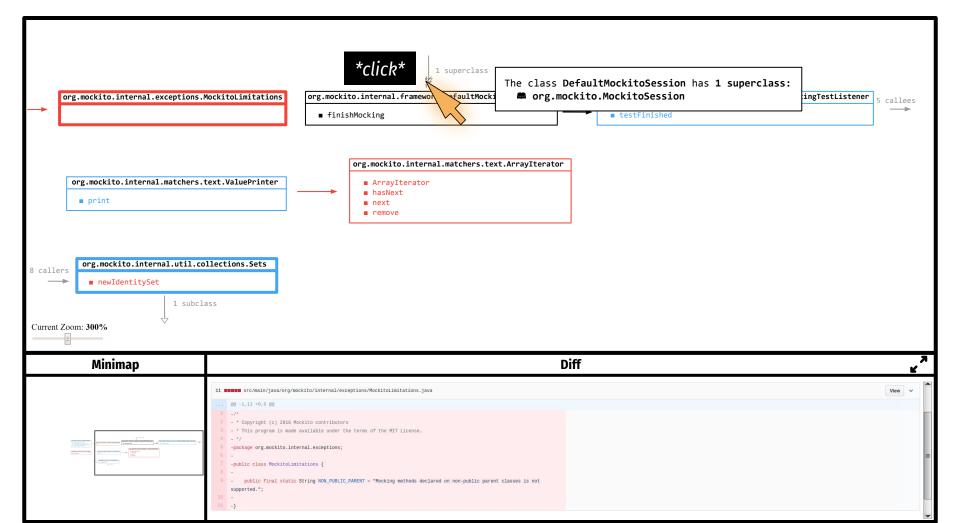


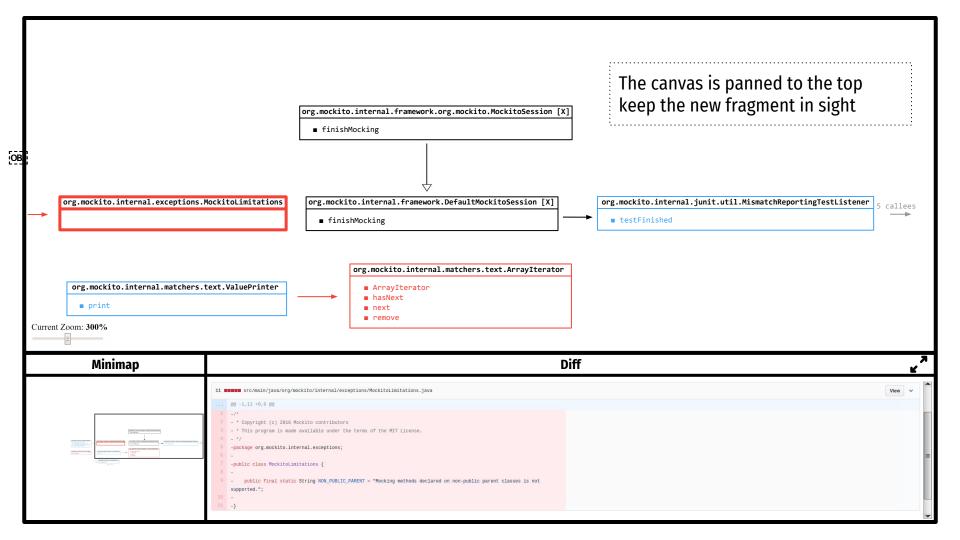


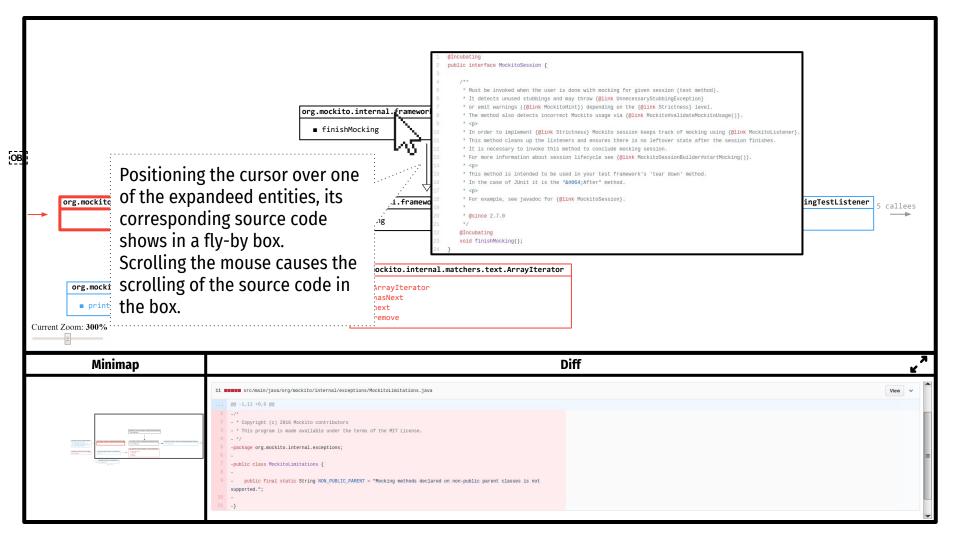


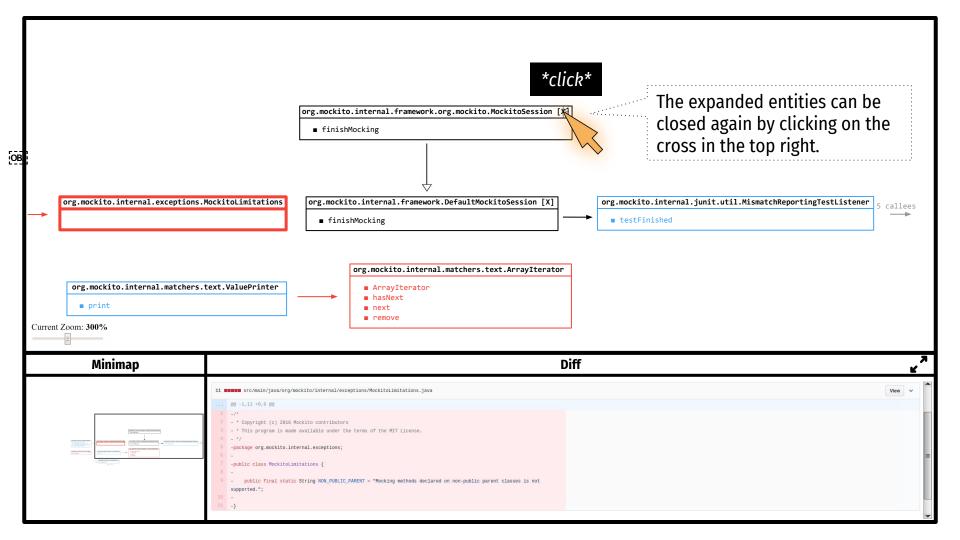


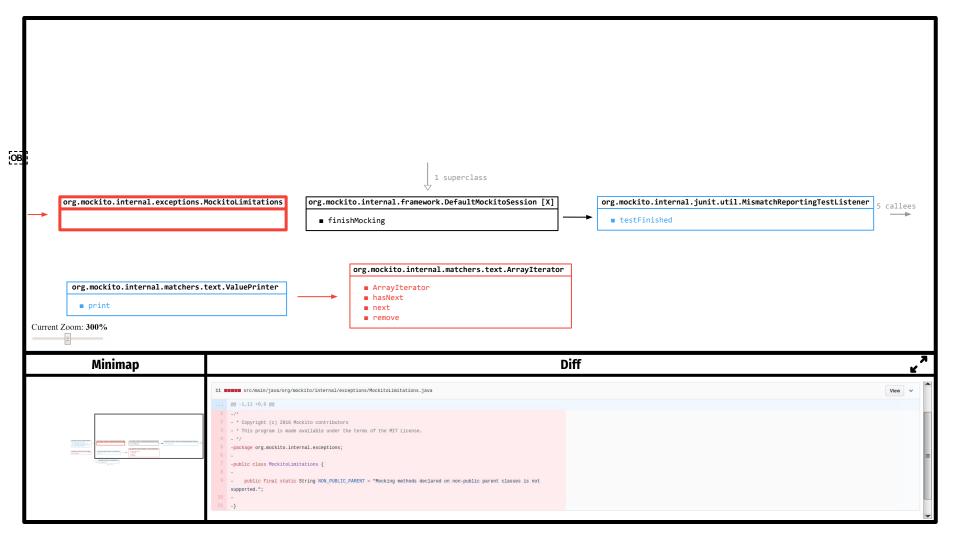


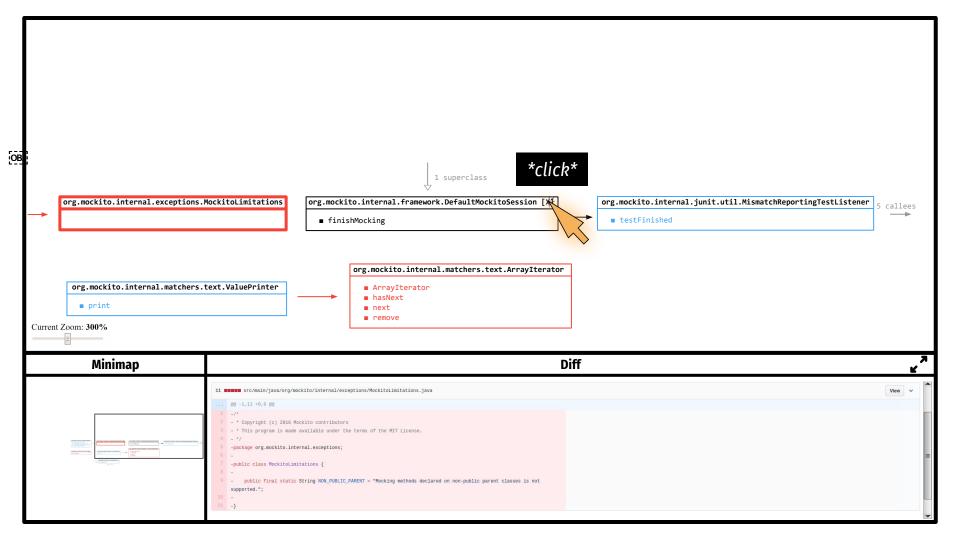




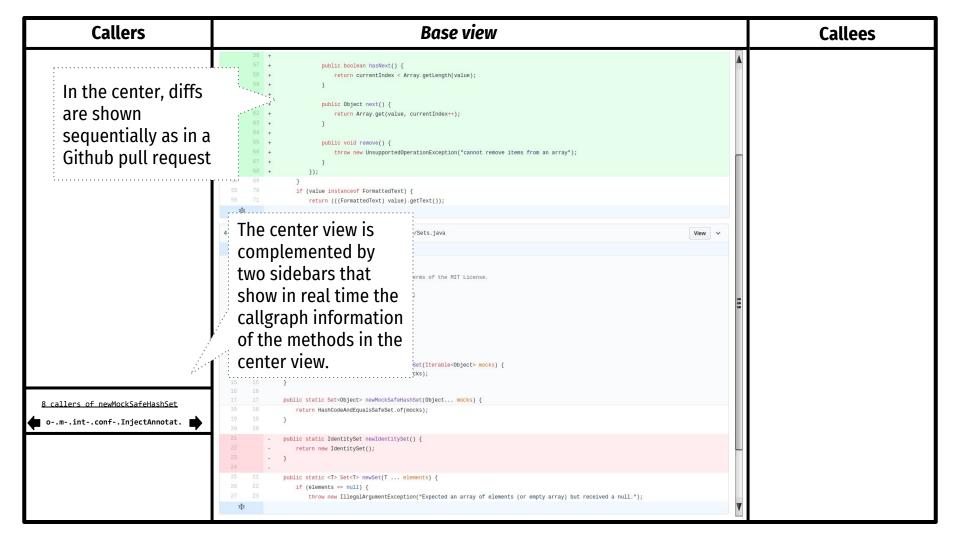


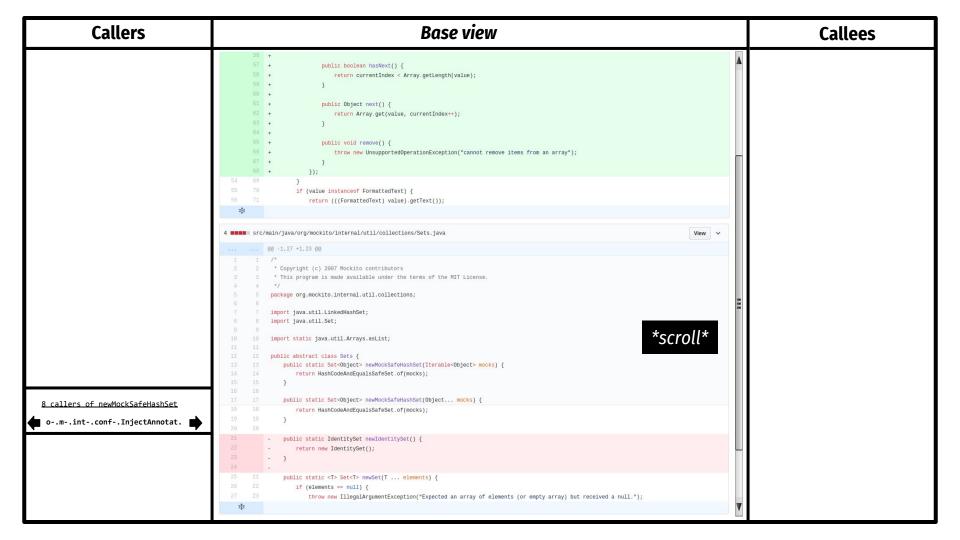


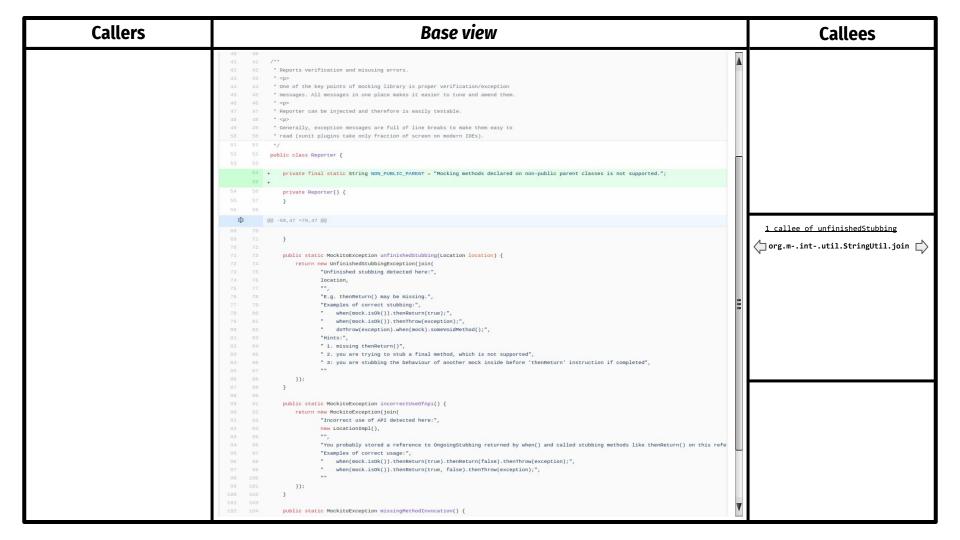


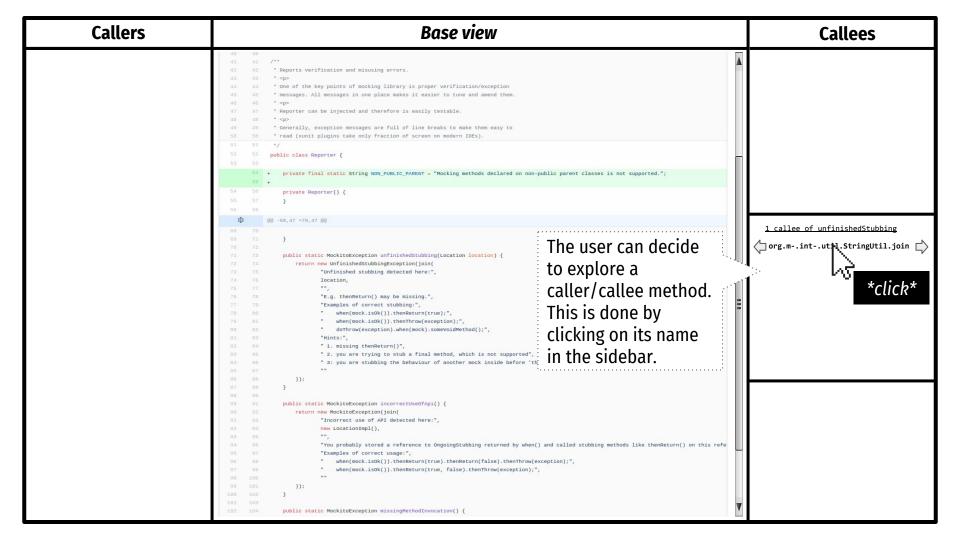


Prototype 3









Callers Base view [X] org.mockito.internal.util.StringUtil [X] Callees * Copyright (c) 2017 Mockito contributors * This program is made available under the terms of the MIT License package org.mockito.internal.util; A new tab with the import static java.util.Arrays.asList; source code of the import java.util.Collection; import java.util.regex.Matcher; import java.util.regex.Pattern; class containing the public class StringUtil { selected caller/callee private static final Pattern CAPS = Pattern.compile("([A-Z\\d][^A-Z\\d]*)"); is opened. private StringUtil() {} The sidebars to have the first line removed * Øreturn less first line represent now the public static String removeFirstLine(String text) { caller/callee return text.replaceFirst(".*?\n", ""); information of this The file is * Joins Strings with line break character. It adds line break in front, too. tab/class, which is in * This makes it something like 'format' no really 'join' positioned at public static String join(Object ... linesToBreak) { focus (italic title). the line return join("\n", asList(linesToBreak)); corresponding * Joins Strings with EOL character to the selected * @param start the starting String method, which * @param lines collection to join public static String join(String start, Collection<?> lines) { is also return join(start, "", lines); highlighted. * Joins Strings with EOL character * Oparam start the starting String * @param linePrefix the prefix for each line to be joined * Oparam lines collection to join

```
Callers
                                   Base view [X]
                                                                                                org.mockito.internal.util.StringUtil [X]
                                                                                                                                                                                                      Callees
                                       * Copyright (c) 2017 Mockito contributors
                                       * This program is made available under the terms of the MIT License.
                                       package org.mockito.internal.util;
                                      import static java.util.Arrays.asList;
                                       import java.util.Collection;
                                   import java.util.regex.Matcher;
                                   import java.util.regex.Pattern;
                                   13 public class StringUtil {
                                          private static final Pattern CAPS = Pattern.compile("([A-Z\\d][^A-Z\\d]*)");
                                          private StringUtil() {}
                                           * @param text
                                                       to have the first line removed
                                           * @return less first line
                                          public static String removeFirstLine(String text) {
                                              return text.replaceFirst(".*?\n", "");
                                           * Joins Strings with line break character. It adds line break in front, too.
                                           * This makes it something like 'format' no really 'join'.
                                          public static String join(Object ... linesToBreak) {
                                              return join("\n", asList(linesToBreak));
                                           * Joins Strings with EOL character
                                           * @param start the starting String
                                           * @param lines collection to join
                                          public static String join(String start, Collection<?> lines) {
                                              return join(start, "", lines);
                                           * Joins Strings with EOL character
                                           * Oparam start the starting String
                                                                                                                     *scroll to left*
                                           * @param linePrefix the prefix for each line to be joined
                                           * Oparam lines collection to join
```

