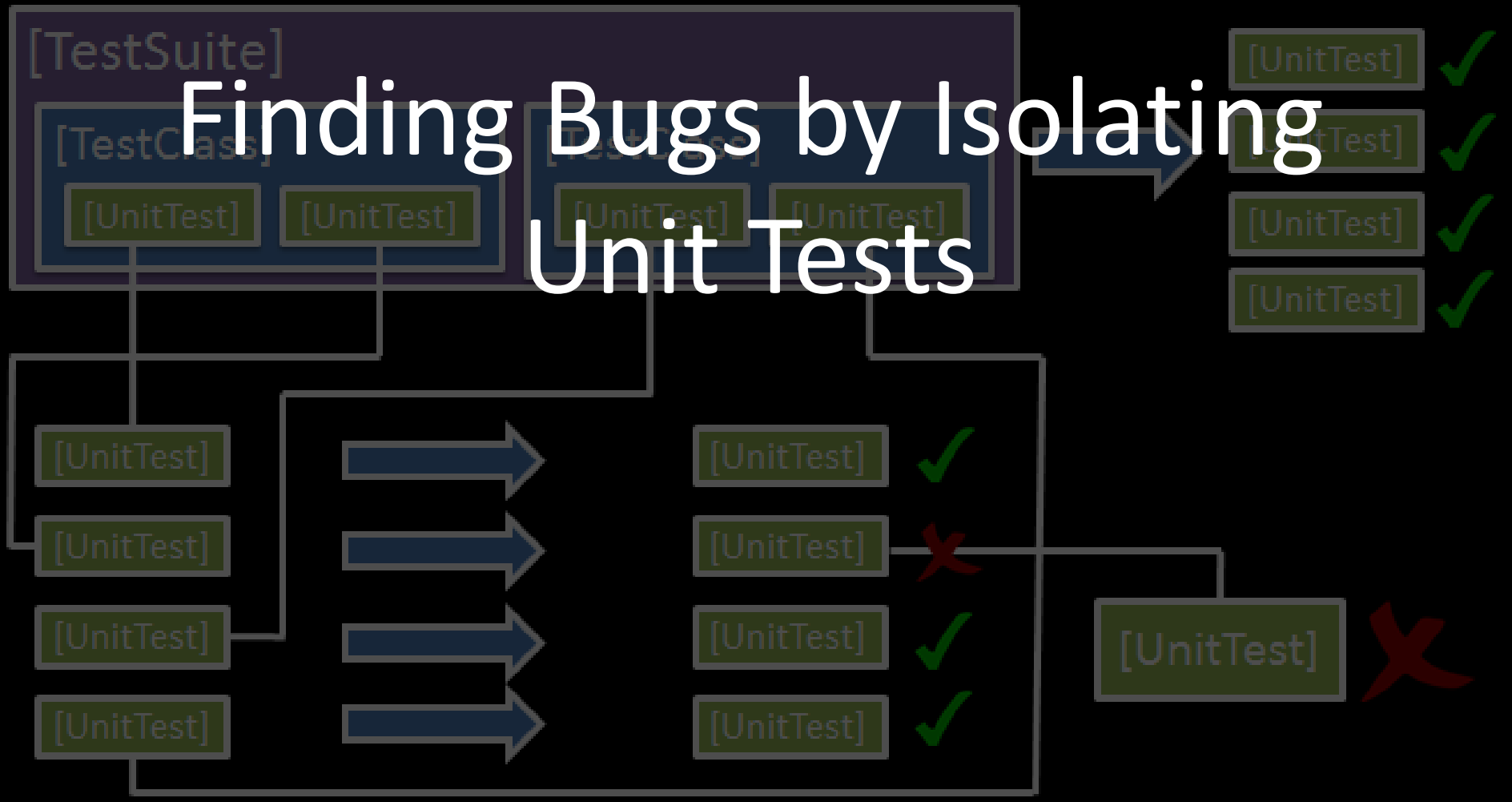
Finding Bugs by Isolating Unit Tests



Kıvanç MUŞLU, Bilge SORAN, Jochen WUTTKE

University of Washington, Computer Science & Engineering

Independence Assumption

- Unit tests are assumed to be independent
  - Test frameworks (e.g., JUnit)
  - Test selection and test prioritization

"For the most part we just ***assumed*** that test cases would be ***independent***, but we didn't think to make that ***explicit***. "

Michal Young
Author of *Software Testing and Analysis: Process, Principles, and Techniques*

Example (Apache Commons CLI)

Code

```
public class OptB {
   private static String name;

   public static Option create() {
      Option result = new Option();
      clear(); return result;
   }
   public static Option createWithArg() {
      Option result = new Option(name);
      clear(); return result;
   }
   private static void clear() { name = "arg"; }
```

*OptB = OptionBuilder

Test

```
public void testOption() {
   Option opt = OptB.create();
   // Test some property of opt.
```

Test

```
public void testOptionWithArg() {
   Option opt = OptB.createWithArg();
   assertEquals(opt.getName(), "arg")
```

Example (Apache Commons CLI)

Code

```
public class OptB {
   private static String name;

   public static Option create() {
      Option result = new Option();
      clear(); return result;
   }
   public static Option createWithArg() {
      Option result = new Option(name);
      clear(); return result;
   }
   private static void clear() { name = "arg"; }
```

*OptB = OptionBuilder

Test

```
public void testOption() {
   Option opt = OptB.create();
   // Test some property of opt.
```

Test

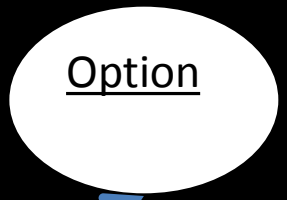
```
public void testOptionWithArg() {
   Option opt = OptB.createWithArg();
   assertEquals(opt.getName(), "arg")
```

Example (Apache Commons CLI)

Code

```
public class OptB {
   private static String name;

   public static Option create() {
      Option result = new Option();
      clear(); return result;
   }
   public static Option createWithArg() {
      Option result = new Option(name);
      clear(); return result;
   }
   private static void clear() { name = "arg"; }
```



*OptB = OptionBuilder

Test

```
public void testOption() {
   Option opt = OptB.create();
   // Test some property of opt.
```

Test

```
public void testOptionWithArg() {
   Option opt = OptB.createWithArg();
   assertEquals(opt.getName(), "arg")
```

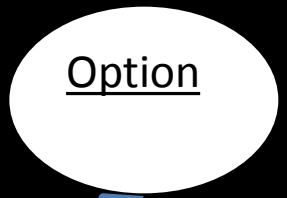
1

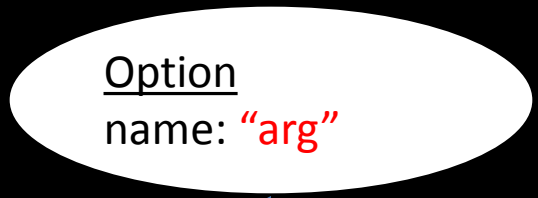
Example (Apache Commons CLI)

Code

```
public class OptB {
   private static String name;

   public static Option create() {
      Option result = new Option();
      clear(); return result;
   }
   public static Option createWithArg() {
      Option result = new Option(name);
      clear(); return result;
   }
   private static void clear() { name = "arg"; }
```





*OptB = OptionBuilder

Test

```
public void testOption() {
   Option opt = OptB.create();
   // Test some property of opt.
```

1

Test

```
public void testOptionWithArg() {
   Option opt = OptB.createWithArg();
   assertEquals(opt.getName(), "arg")
```

2

Example (Apache Commons CLI)

Code

```
public class OptB {
   private static String name;

   public static Option create() {
      Option result = new Option();
      clear(); return result;
   }
   public static Option createWithArg() {
      Option result = new Option(name);
      clear(); return result;
   }
   private static void clear() { name = "arg"; }
```

*OptB = OptionBuilder

Test

```
public void testOption() {
   Option opt = OptB.create();
   // Test some property of opt.
```

Test

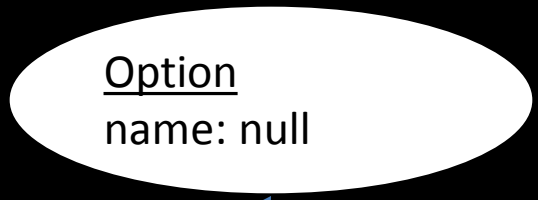
```
public void testOptionWithArg() {
   Option opt = OptB.createWithArg();
   assertEquals(opt.getName(), "arg")
```

Example (Apache Commons CLI)

Code

```
public class OptB {
   private static String name;

   public static Option create() {
      Option result = new Option();
      clear(); return result;
   }
   public static Option createWithArg() {
      Option result = new Option(name);
      clear(); return result;
   }
   private static void clear() { name = "arg"; }
```



*OptB = OptionBuilder

Test

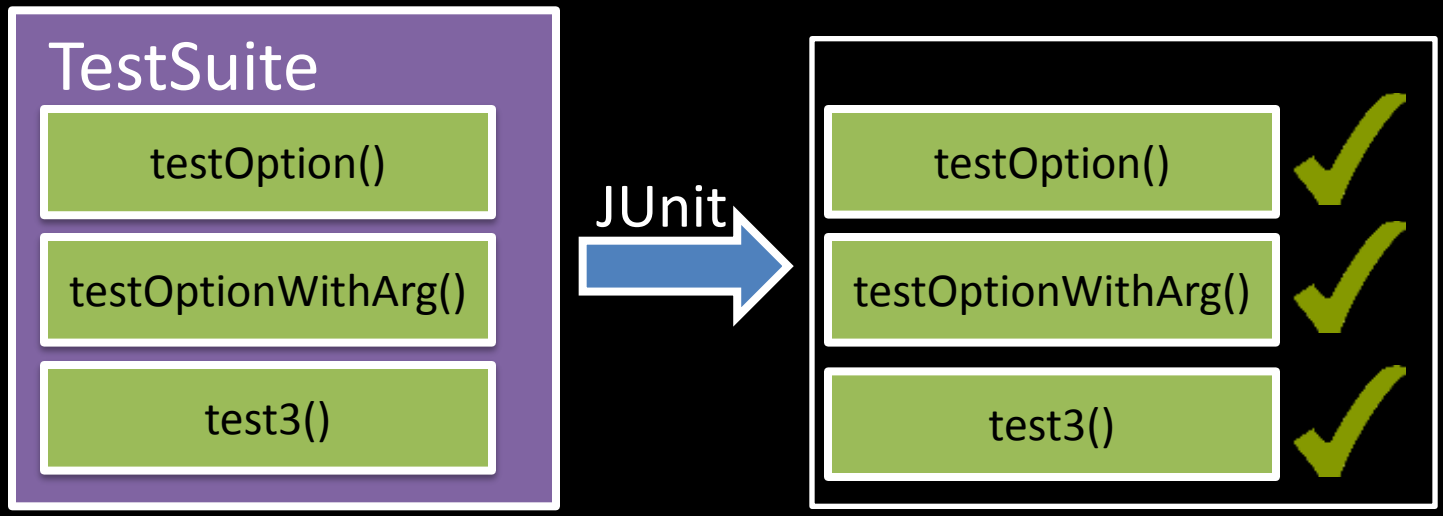
```
public void testOption() {
   Option opt = OptB.create();
   // Test some property of opt.
```

Test

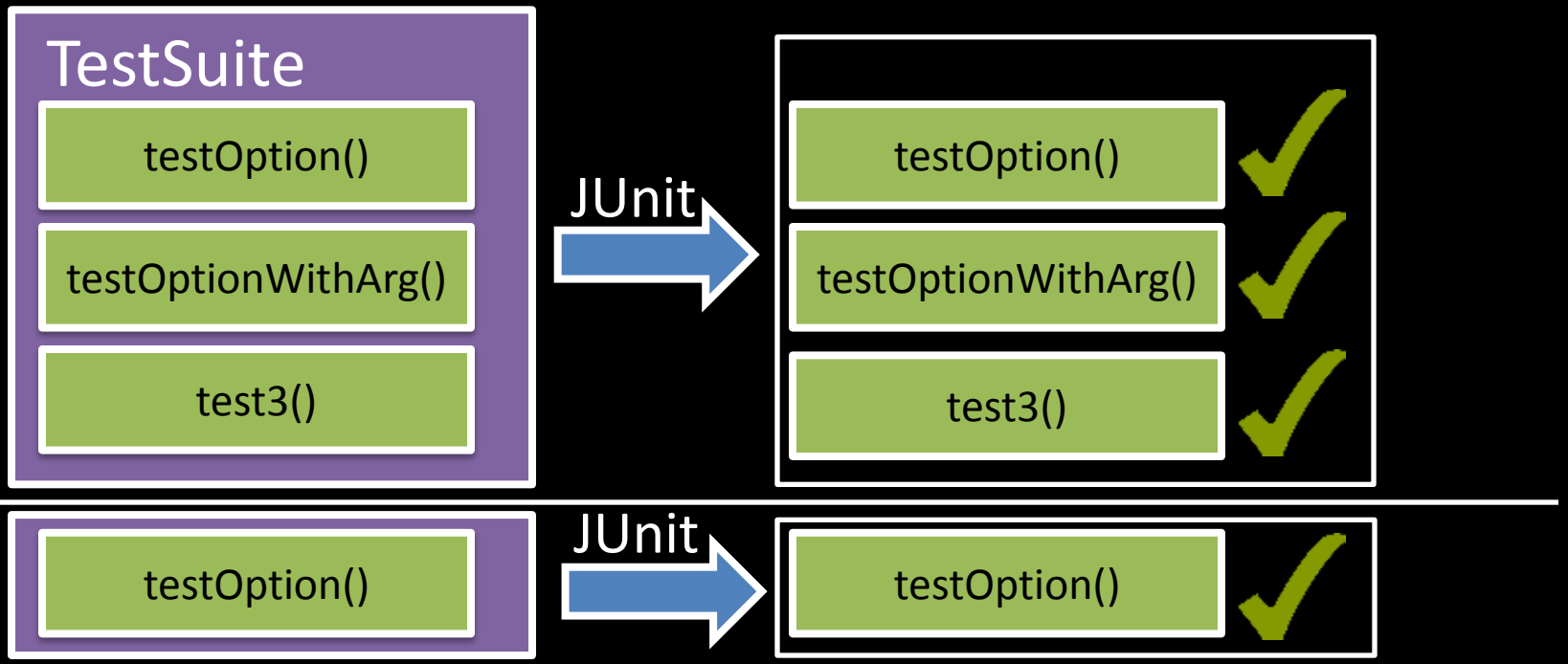
```
public void testOptionWithArg() {
   Option opt = OptB.createWithArg();
   assertEquals(opt.getName(), "arg")
```

1

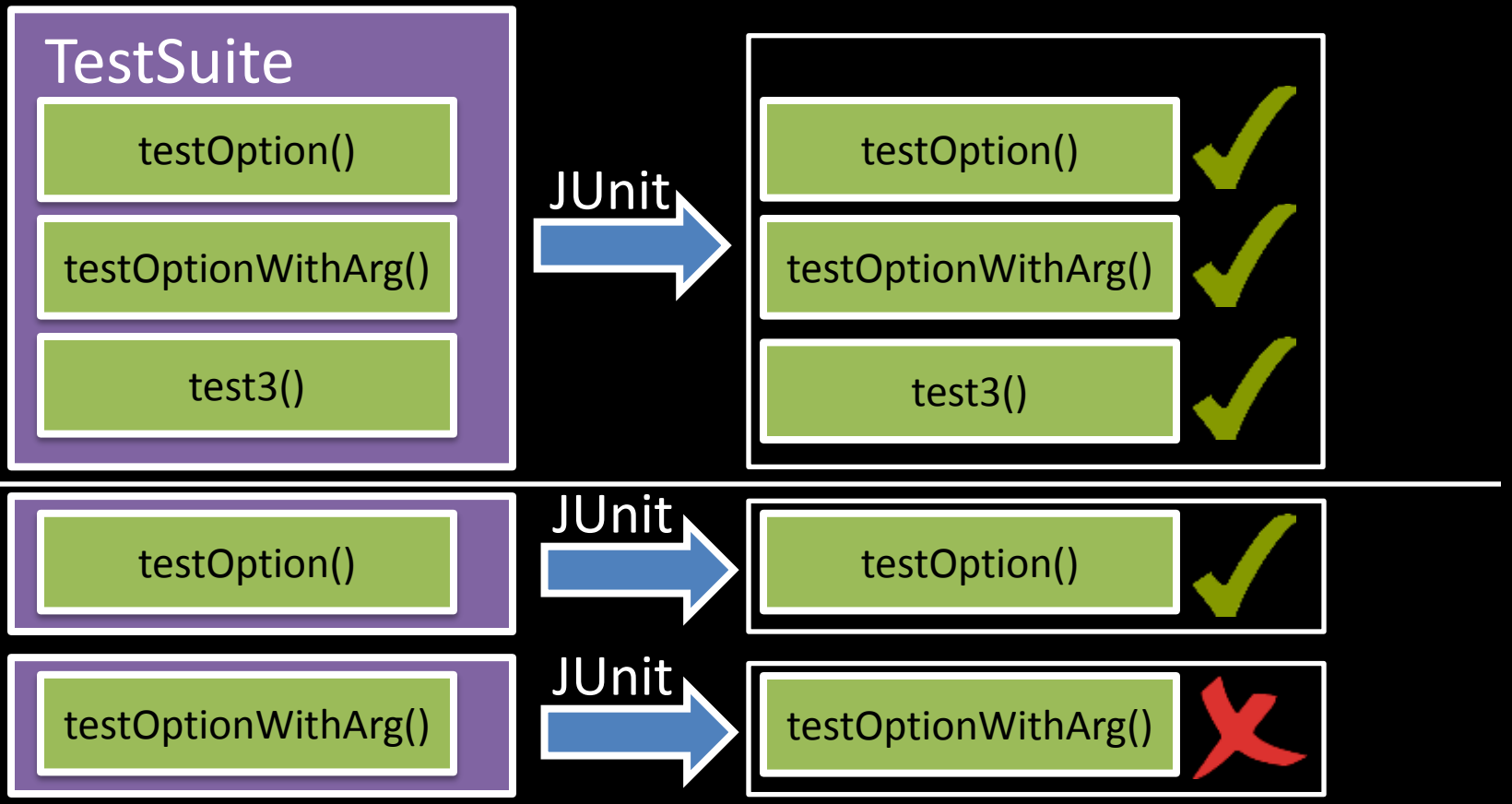
Test Isolation



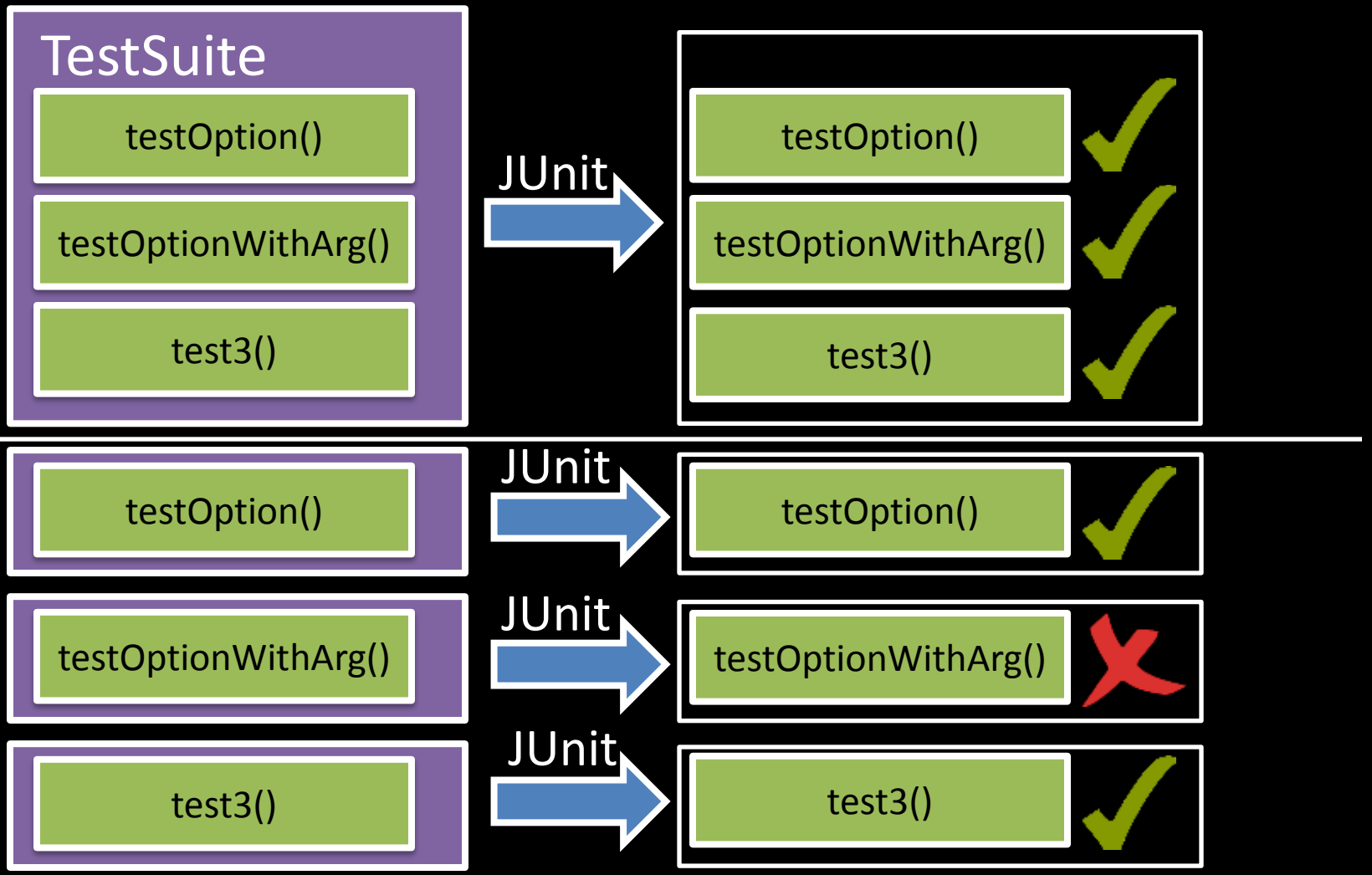
Test Isolation



Test Isolation



Test Isolation

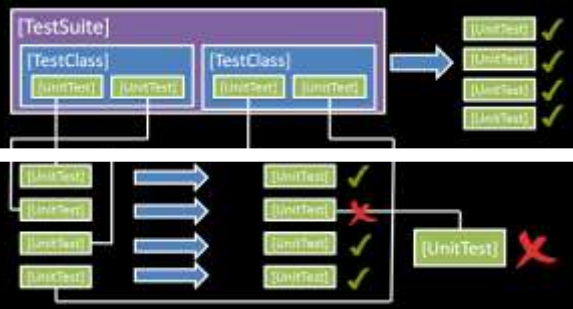


Preliminary Results

153 Revisions

commons CLI

Test Isolator



3 Failing Tests

test27635 test13666

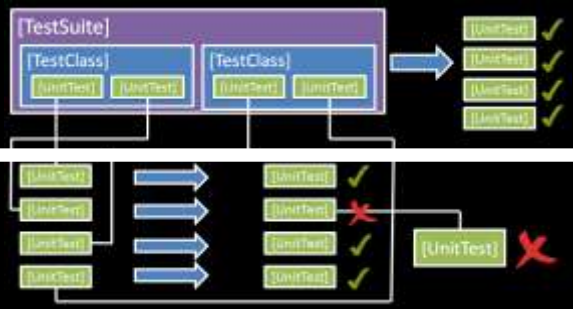
testOptionWithout ShortFormat2

Preliminary Results

153 Revisions

commons CLI

Test Isolator



3 Failing Tests

test27635 test13666

testOptionWithout ShortFormat2

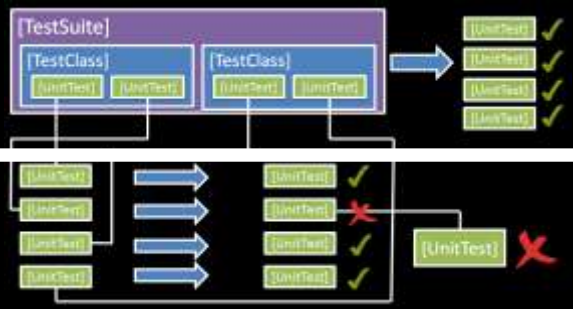
OptionBuilder.java

Preliminary Results

153 Revisions

commons CLI

Test Isolator



3 Failing Tests

test27635 test13666

testOptionWithoutShortFormat2

OptionBuilder.java

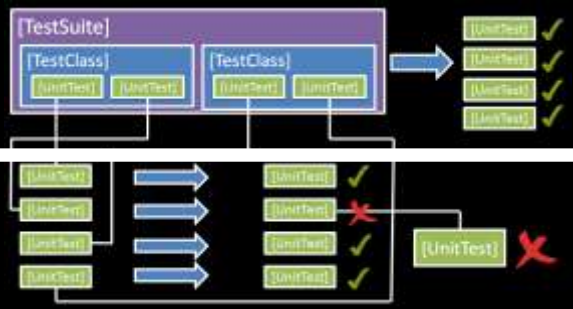
2004 — Reported to the bug database by a user.

Preliminary Results

153 Revisions

commons CLI

Test Isolator



3 Failing Tests

test27635 test13666

testOptionWithout ShortFormat2

OptionBuilder.java

2004 — Reported to the bug database by a user.

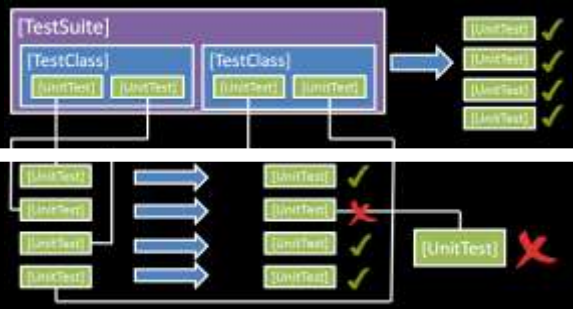
2007 — Marked as resolved.

Preliminary Results

153 Revisions

commons CLI

Test Isolator



3 Failing Tests

test27635 test13666

testOptionWithout ShortFormat2

OptionBuilder.java

2004 — Reported to the bug database by a user.

2007 — Marked as resolved.

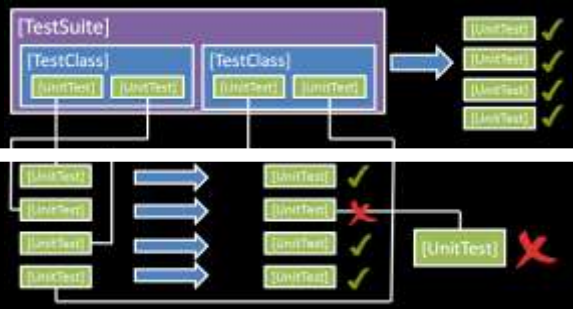
2009 — Reopened by another user submission.

Preliminary Results

153 Revisions

commons CLI

Test Isolator



3 Failing Tests

test27635 test13666

testOptionWithout ShortFormat2

OptionBuilder.java

2004 — Reported to the bug database by a user.

2007 — Marked as resolved.

2009 — Reopened by another user submission.

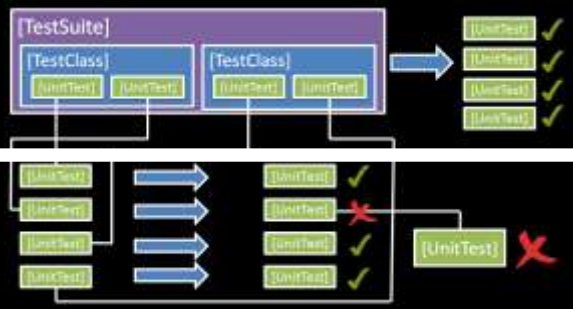
2010 — Fixed.

Preliminary Results

153 Revisions

commons CLI

Test Isolator



3 Failing Tests

test27635 test13666

testOptionWithout ShortFormat2

OptionBuilder.java

2004 — Reported to the bug database by a user.

2007 — Marked as resolved.

2009 — Reopened by another user submission.

2010 — Fixed.

120 Revisions
3 Changes to
OptionBuilder.java

Related Work

- Independence between unit tests is widely assumed
- Problem is noted by other work on testing

  [Robinson2011]

Future Work

- Analyzing more OSS projects
- Characterizing data dependency types
- Pinpointing the location of the dependency
- Extending the existing testing frameworks to support running tests in isolation

Contributions

- Counterexample to the independence assumption
  - Real system
  - Real cost
  - Easy to confirm
- Adds incentive to testing R&D to reconsider independence assumption