

# mockito



Präsentation von Hermüller Philipp

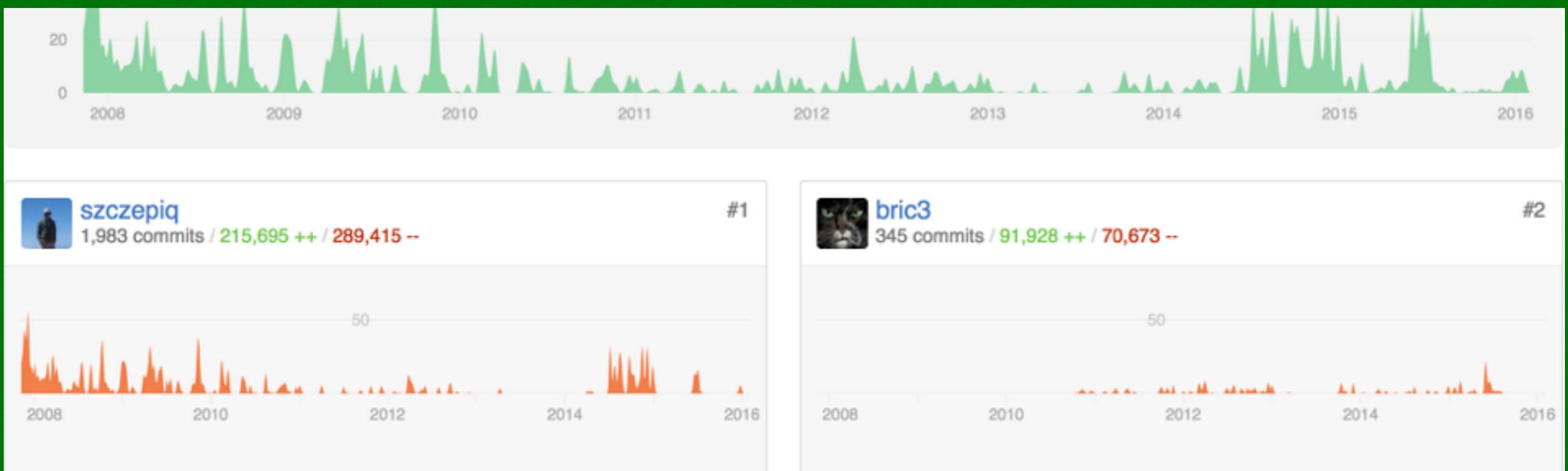
# Mockito on GitHub



- Last Commit >> 26.Jan.2016
- On GitHub since 11.Nov.2007
- before on GoogleCode

```
<dependency>
    <groupId>org.mockito</groupId>
    <artifactId>mockito-core</artifactId>
    <version>1.10.19</version>
    <scope>test</scope>
</dependency>
```

# Code Graph



# Latest Versions



## 2.0.41-beta (2016-01-26 22:52 UTC)

- Authors: 2
- Commits: 3
  - 2: Urs Metz
  - 1: Rafael Winterhalter
- Improvements: 1
  - Show correct location of unwanted interaction with mock when using MockitoJUnitRule (#344)

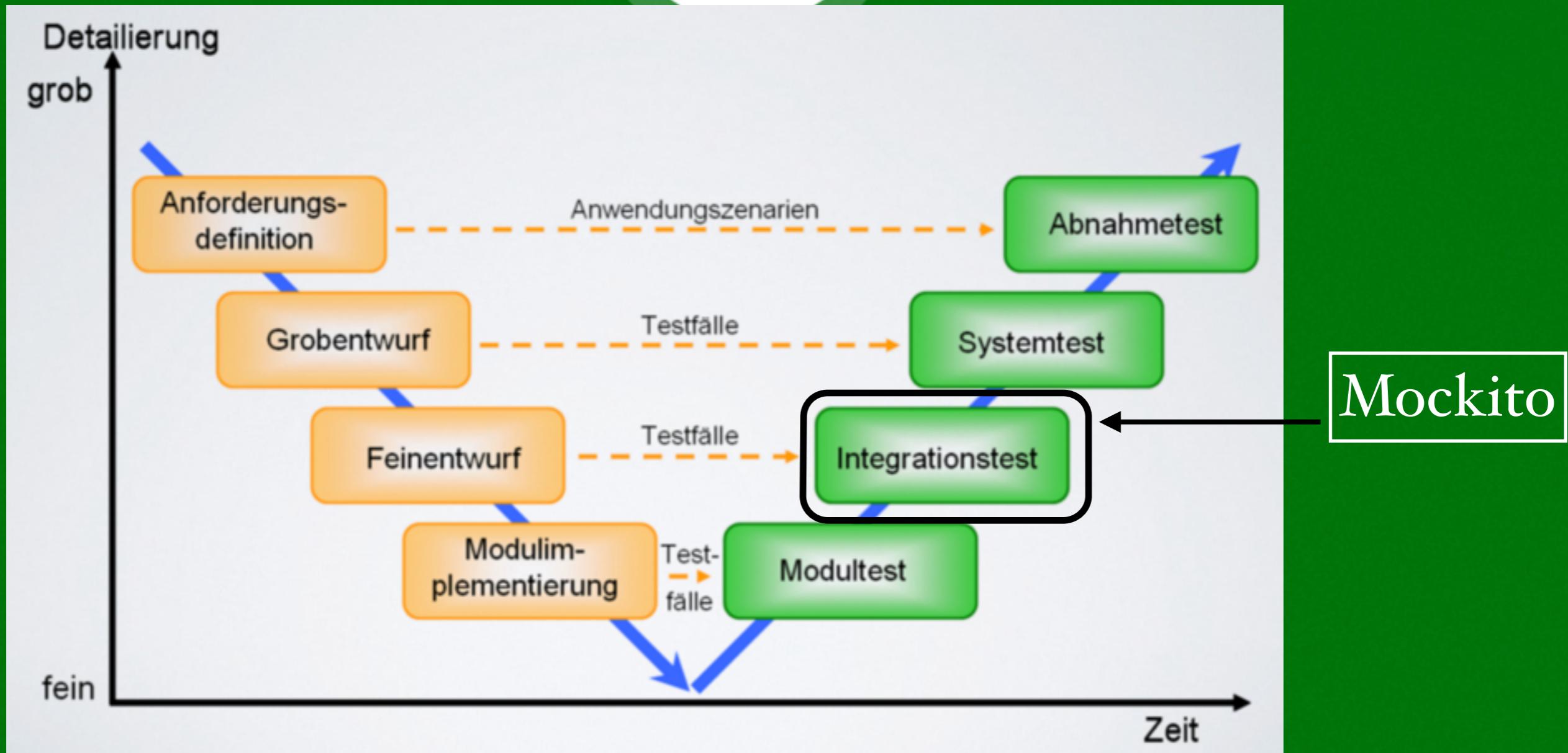
## 2.0.40-beta (2016-01-20 21:06 UTC)

- Authors: 3
- Commits: 4
  - 2: Rafael Winterhalter
  - 1: Michael Pinnegar
  - 1: Rafael Winterhalter
- Improvements: 3
  - Fixes #256 :Alternative fix to #259, windows build (#342)
  - Removing new line in bottom script. it seems that javadoc... (#259)
  - task mockitoJavadoc fails when compiling in windows (#256)

## 2.0.39-beta (2016-01-18 20:57 UTC)

- Authors: 1
- Commits: 1
  - 1: Rafael Winterhalter
- No notable improvements. See the commits for detailed changes.

# V-Modell



# Warum Mockito benutzen?



- gut leserlicher code
- einfach zu verstehen
- erzeugen schöne Fehlermeldungen

# Saubere Fehlermeldungen

vorher: `verify(ml, times(3)).addMessageAt(anyInt(), "Good Day");`

```
org.mockito.exceptions.misusing.InvalidUseOfMatchersException:  
Invalid use of argument matchers!  
2 matchers expected, 1 recorded:  
-> at MessageTest.t102_CountMessages(MessageTest.java:70)
```

This exception may occur if matchers are combined with raw values:

```
//incorrect:  
someMethod(anyObject(), "raw String");
```

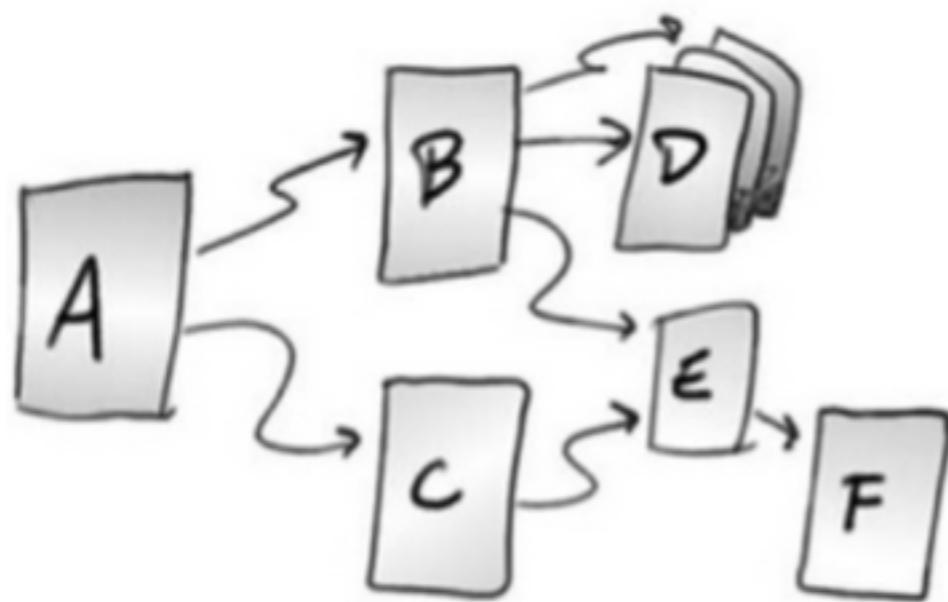
When using matchers, all arguments have to be provided by matchers.  
For example:

```
//correct:  
someMethod(anyObject(), eq("String by matcher"));
```

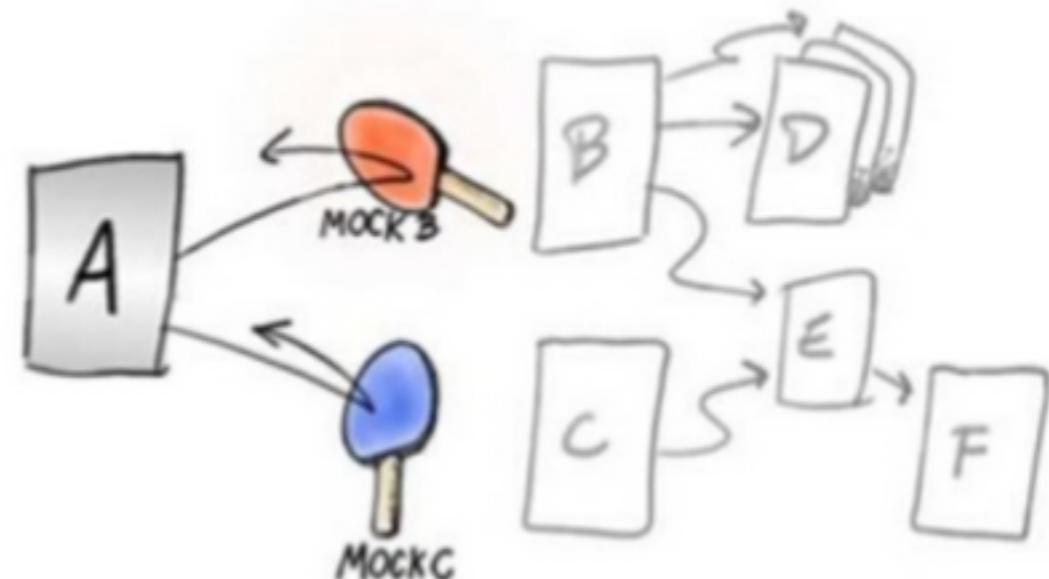
For more info see javadoc for Matchers class.

nachher: `verify(ml, times(3)).addMessageAt(anyInt(), eq("Good Day"));`

# Darstellung



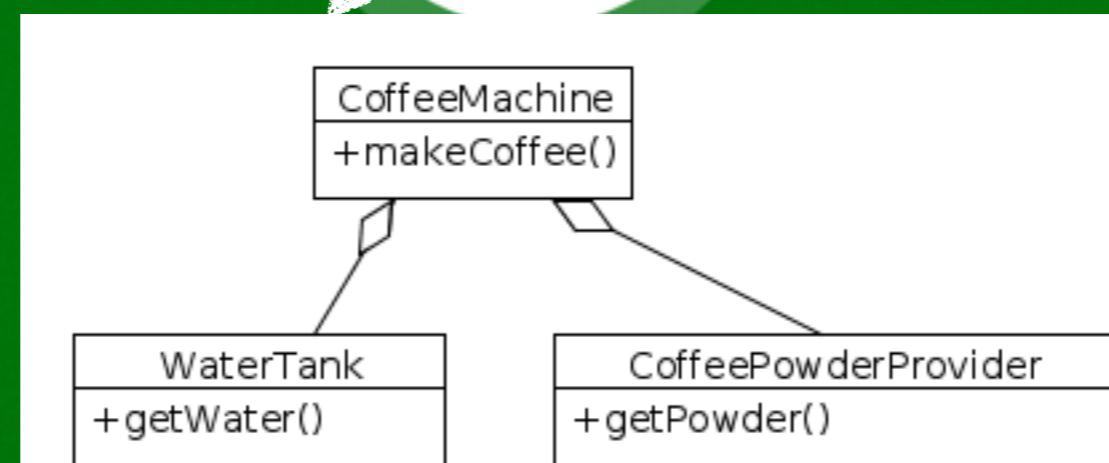
real objects



mocked objects

# Übersicht des Beispiels

zu testende  
Klasse



Klasse die  
gemockt wird

# Wichtige Befehle



- `when(...).thenReturn(...);`
- `doReturn(...), doNothing(), ...`
- `verify(mock, timeout(...).times(...)).method();`
- `inOrder`
- `verifyNoMoreInteractions(mock);`  
`verifyZeroInteractions(mock);`
- `anyInt(), anyString(), anyObject()`...
- bei konstanten `>>> eq( ... )`
- `times(...), atLeastOnce(), atLeast(...), atMost(...)`

# Links



- <http://site.mockito.org/>
- <http://github.com/mockito/mockito>
- <http://fruzenshtein.com/junit-and-mockito/>



„Danke für Eure  
Aufmerksamkeit“