

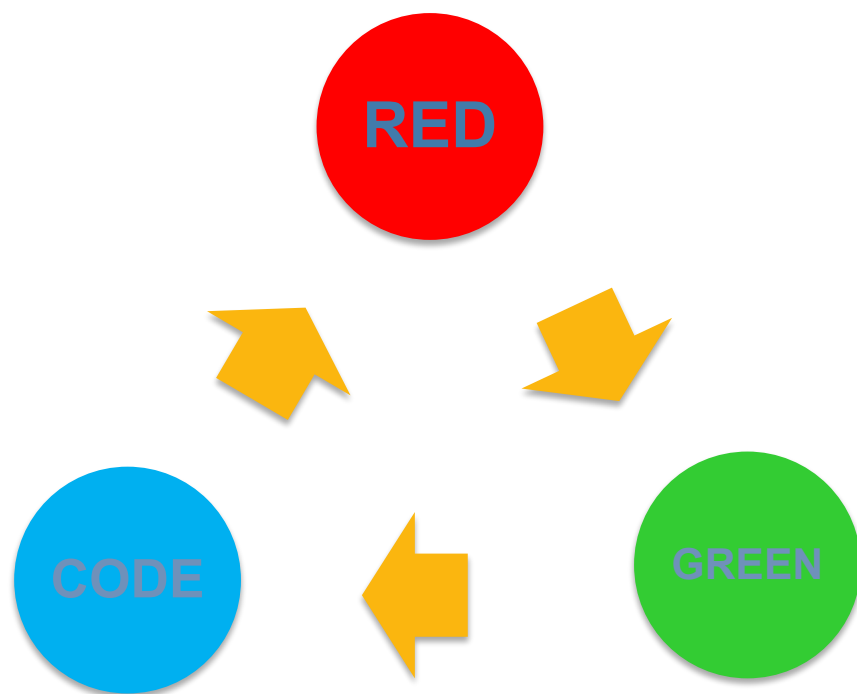


Testing for Java Developers

Agenda

- Test Driven Development (TDD)
- JUnit — библиотека для модульного тестирования
- Mock-библиотеки
- xUnit patterns

Сначала тесты, а потом код!



Проблемы TDD

- Люди с трудом используют TDD сам по себе, когда они не имеют большого опыта работы с ним.
- В обучение TDD используются упрощенные примеры, в отличие от реальных систем.
- Требуется больше времени, чтобы экспериментировать и пробовать без обычного давления со стороны менеджмента, который требует выпускать ПО регулярно.
- И как всегда нет времени на учебу — нужно выпускать продукт.
- «Лжемнение» - отнимает время

Advantages of TDD

- Writing tests first require you to really consider what you want from the code
- Short feedback loop
- Creates a detailed specification
- Reduced time in rework
- Less time spent in the debugger and when it is required you usually get closer to problem quickly
- Tells you whether your last change (or refactoring) has broken previously working code
- Allows the design to evolve and adapt to your changing understanding of the problem.



Введение аннотаций Java 5 «облегчило» технологию JUnit 4 и повысило ее гибкость. Это позволило отказаться от строгих соглашений о присваивании имён и иерархий наследования в пользу ряда новых впечатляющих функций. Ниже приведен краткий перечень новинок, появившихся в версии JUnit 4:

- Параметризованные тесты
- Тесты исключений
- Тесты с ограничением по времени
- Гибкие фикстуры
- Простые средства игнорирования тестов
- Новый способ логического группирования тестов

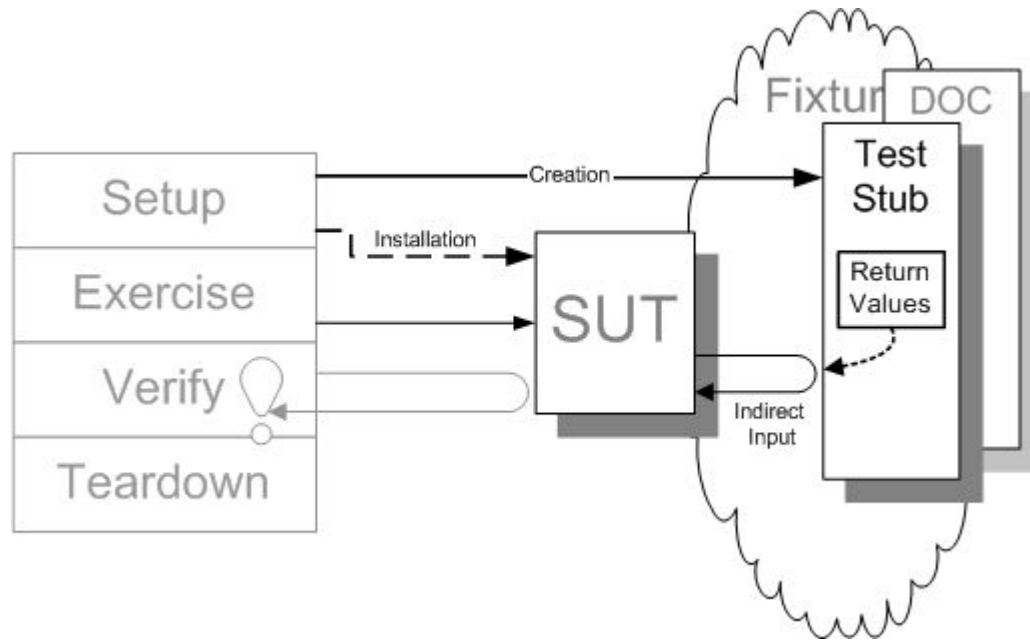


- EasyMock

xUnit patterns

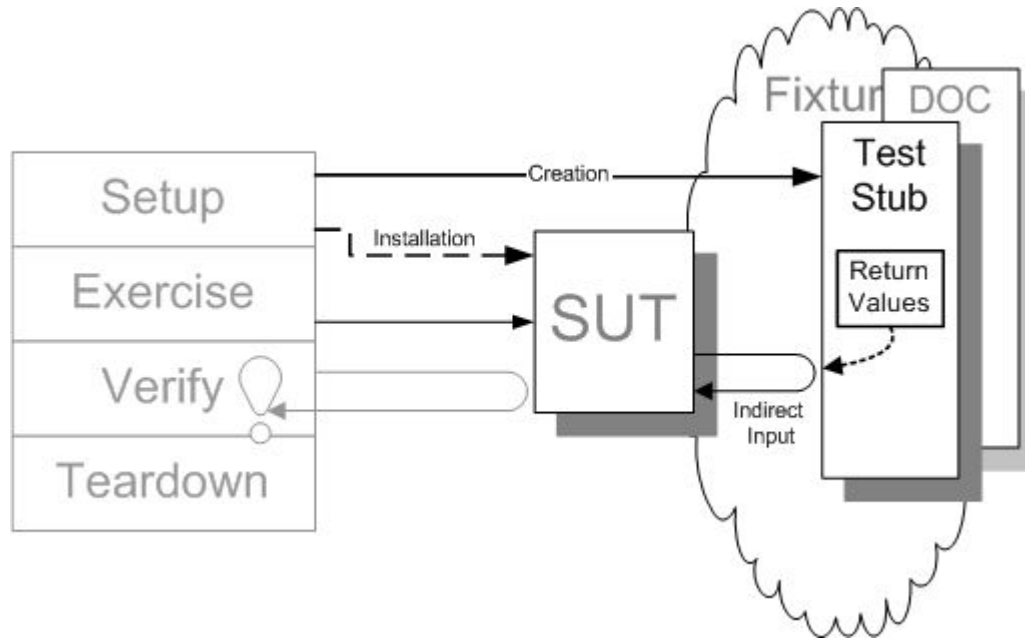
- Dummy
- Fake
- Stub
- Mock

Test Stub



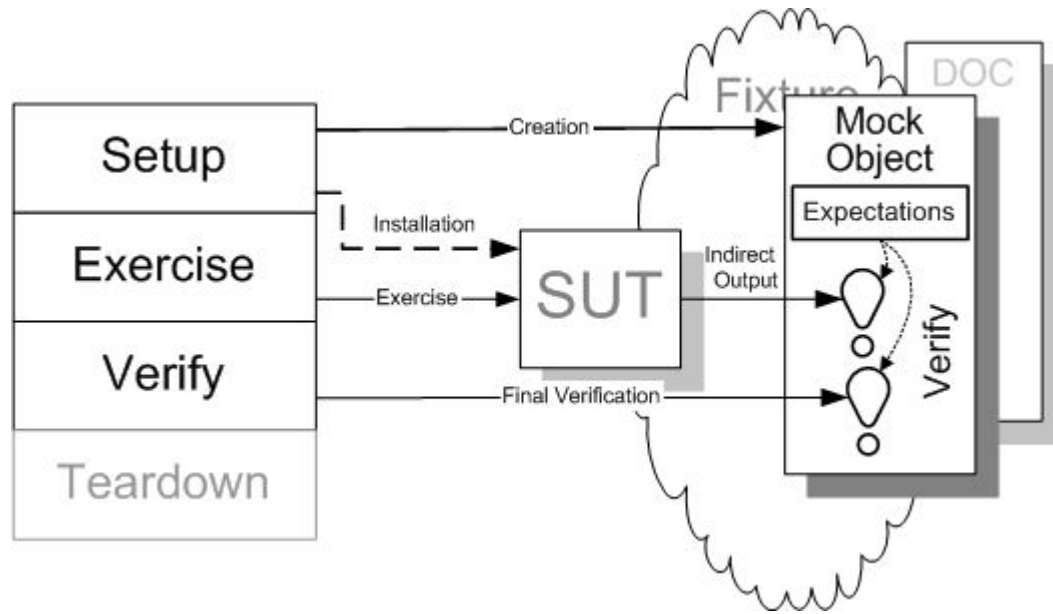
..T..Systems.....

Test Spy



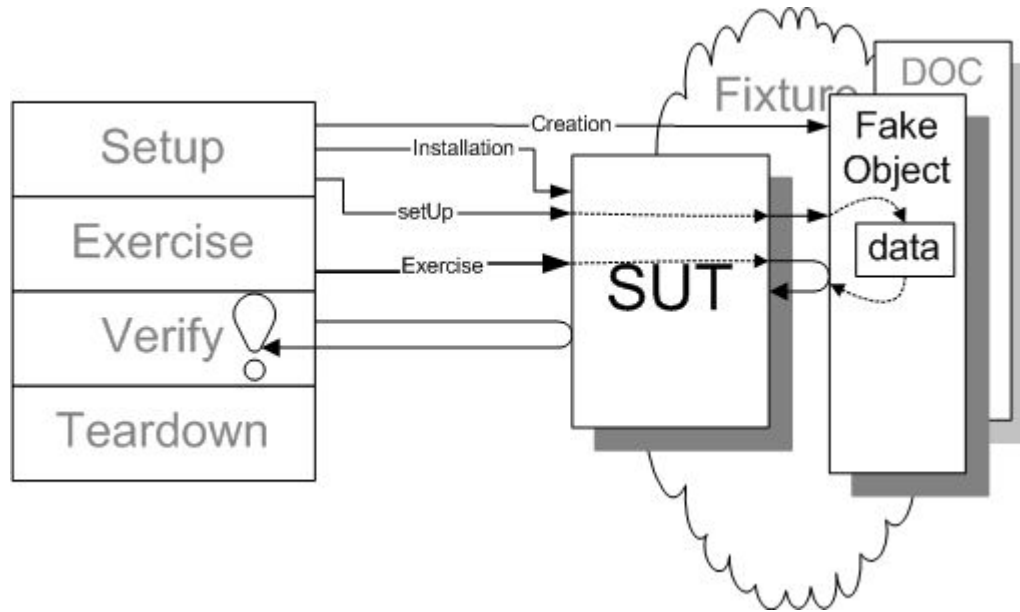
... T ... Systems ...

Mock Object



..T..Systems.....

Fake object



... T ... Systems ...

References

- Gerard Meszaros. **Xunit Test Patterns**: Refactoring Test Code.