

System and Testing Validation (STV08)

A template based method for automatic unit
test case generation of web applications

Boni García



Introduction

- Web applications are more and more popular
- Web Application Framework simplifies development
 - Java EE, Spring Framework
- Agile WAF
 - Ruby on Rails, Grails, Trails, Roma
- Software testing: important but complex and costly
 - Automatic testing could save time and effort
- Problem: **automatic testing** for applications based on **agile WAF**

Table of Contents

- Introduction
- **Background**
- Automatic Testing Platform
- ATP4Romulus
- Conclusions & Future Work

Background - WAF

- Java
 - Java EE
 - Spring Framework
- Agile development:
 - Ruby on Rails
 - Grails
 - Trails
 - Roma/Romulus

Background – Auto Testing

- Specification based: SDL
- Model based: UML
- Path oriented: control flow
- Random generation: random test distribution
- Goal oriented: agents
- Intelligent test case generation: computation
- Template based: flexible

Table of Contents

- Introduction
- Background
- **Automatic Testing Platform**
- ATP4Romulus
- Conclusions & Future Work

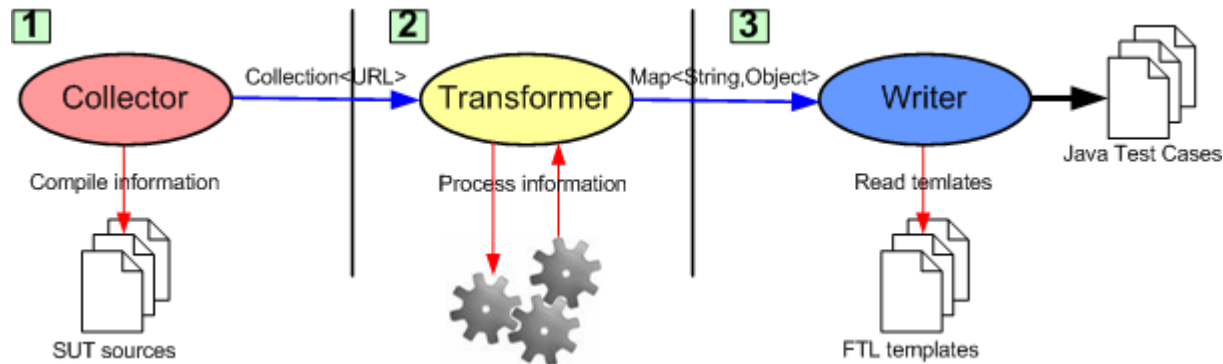
Automatic Testing Platform (I)

- Template based: FreeMarker
- ATP4Java:
 - Generators: Entities in charge of the **automatic** test case generation
 - CSV: Input file for **semi-automatic** test case generation

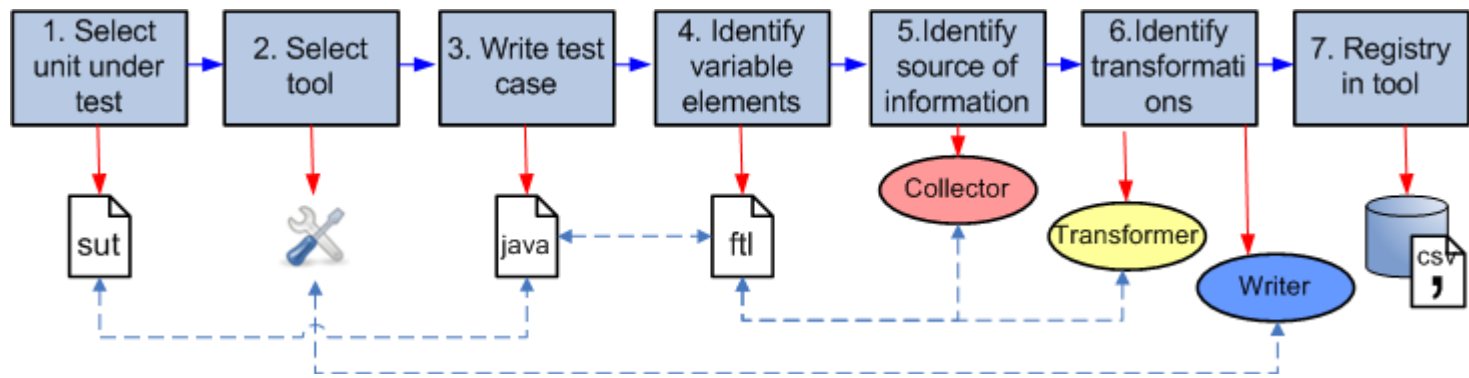


Automatic Testing Platform (II)

- Generators



- Extending ATP4Java



Automatic Testing Platform (III)

- Semi-automatic generation: CSV

```
new,Store,1000  
new,Box,3,2,2  
run,$1,fits,$2  
assert,$3
```

```
new,java.lang.Integer,1000  
new,Store,$1  
new,Box,3,2,2  
run,$2,insert,$3  
run,$2,getCapacity  
assert,$5,!=,$1
```

```
run,StaticStore,getCapacity  
assert,$1,<,1000
```



```
@Test  
public void testCSV_1() throws Exception {  
    Store store1 = new Store(1000);  
    Box box2 = new Box(3,2,2);  
    boolean boolean3 = store1.fits(box2);  
    Assert.assertTrue(boolean3);  
}
```

```
@Test  
public void testCSV_2() throws Exception {  
    Integer integer1 = new Integer(1000);  
    Store store2 = new Store(integer1);  
    Box box3 = new Box(3,2,2);  
    store2.insert(box3);  
    Integer integer5 = store2.getCapacity();  
    Assert.assertTrue(!EqualsBuilder.  
        reflectionEquals(integer5, integer1));  
}
```

```
@Test  
public void testCSV_3() throws Exception {  
    int int1 = StaticStore.getCapacity();  
    Assert.assertTrue(int1 < 1000);  
}
```

Table of Contents

- Introduction
- Background
- Automatic Testing Platform
- **ATP4Romulus**
- Conclusions & Future Work

ATP4Romulus (I)

- Extension of ATP4Java for Romulus framework
- Set of specific generators
- Functional and structural test case generation
 - Domain tests (DDD approach)
 - Functional test (CRUD, I18N, View, ...)
- <http://www.ict-romulus.eu/web/atp4romulus>



Table of Contents

- Introduction
- Background
- Automatic Testing Platform
- ATP4Romulus
- **Conclusions & Future Work**

Conclusions & Future Work

- Open source testing suite released for automatic unit test case generation: **ATP**
 - Focused on agile web development
 - Flexible (template-based)
 - Scalable (by Generators)
- To come: extending ATP to integration and system testing levels (mocks, web testing)

Thank you

Boni García

bgarcia@dit.upm.es

