

# Fault Injection - Java

Sebastian Gerstenberg  
Fault Injection Seminar  
09.07.2015

# Agenda

---

1. Motivation
2. Jaca
3. Byteman
4. Javassist

## **Fault Injection - Java**

Sebastian  
Gerstenberg,  
07.09.2015  
Chart 2

# Agenda

---

- 1. Motivation**
2. Jaca
3. Byteman
4. Javassist

## **Fault Injection - Java**

Sebastian  
Gerstenberg,  
07.09.2015  
Chart **3**

# Motivation

## Code/Fault Injection in Managed Code

---

Changing and injecting bytecode at runtime

- Testing
  - Unit Tests
    - Tracing
    - Validation
    - Fault Injection
  - Hardening

**Fault Injection -  
Java**

Sebastian  
Gerstenberg,  
07.09.2015  
Chart 4

# Motivation

---

## Aspect Oriented Programming through Code Injection

- Dynamically add aspects
- Change of extend existing aspects
- Separation of concerns
- Code readability

**Fault Injection - Java**

Sebastian Gerstenberg,  
07.09.2015  
Chart 5

# Agenda

---

1. Motivation
- 2. Jaca**
3. Byteman
4. Javassist

## **Fault Injection - Java**

Sebastian  
Gerstenberg,  
07.09.2015  
Chart **6**

# Jaca

Developed 2001, University of Campinas

Graphical Interface for Fault Injection  
into compiled Java bytecode

Uses Javassist to modify bytecode



**Fault Injection -  
Java**

Sebastian  
Gerstenberg,  
07.09.2015  
Chart 7

# Jaca Features

---

Faults in:

- Method Parameter
- Method Return
- Field Access

Manipulation of primitive types

Definition of repetition patterns



**Fault Injection -  
Java**

Sebastian  
Gerstenberg,  
07.09.2015  
Chart 8



# Jaca Demo

---

## **Fault Injection - Java**

Sebastian  
Gerstenberg,  
07.09.2015  
Chart 9

# Agenda

---

1. Motivation
2. Jaca
- 3. Byteman**
4. Javassist

## **Fault Injection - Java**

Sebastian  
Gerstenberg,  
07.09.2015  
Chart **10**

# Byteman

---

Being developed by Jboss

Uses Javassist to inject Java source code into existing bytecode

Definition of Faults using Byteman rule-language

Rules can be defined at runtime

Agent can be added to running JVM

**Fault Injection -  
Java**

Sebastian  
Gerstenberg,  
07.09.2015  
Chart **11**

# Byteman - Rules

---

```
RULE injection test
CLASS de.hpi.fault_injection.Math
METHOD multiply(int, int)
AT ENTRY
IF true
DO $1 = 0;
ENDRULE
```

## **Fault Injection - Java**

Sebastian  
Gerstenberg,  
07.09.2015  
Chart **12**

# Byteman - Demo

---

## **Fault Injection - Java**

Sebastian  
Gerstenberg,  
07.09.2015  
Chart **13**

# Agenda

---

1. Motivation
2. Jaca
3. Byteman
4. **Javassist**

## **Fault Injection - Java**

Sebastian  
Gerstenberg,  
07.09.2015  
Chart **14**

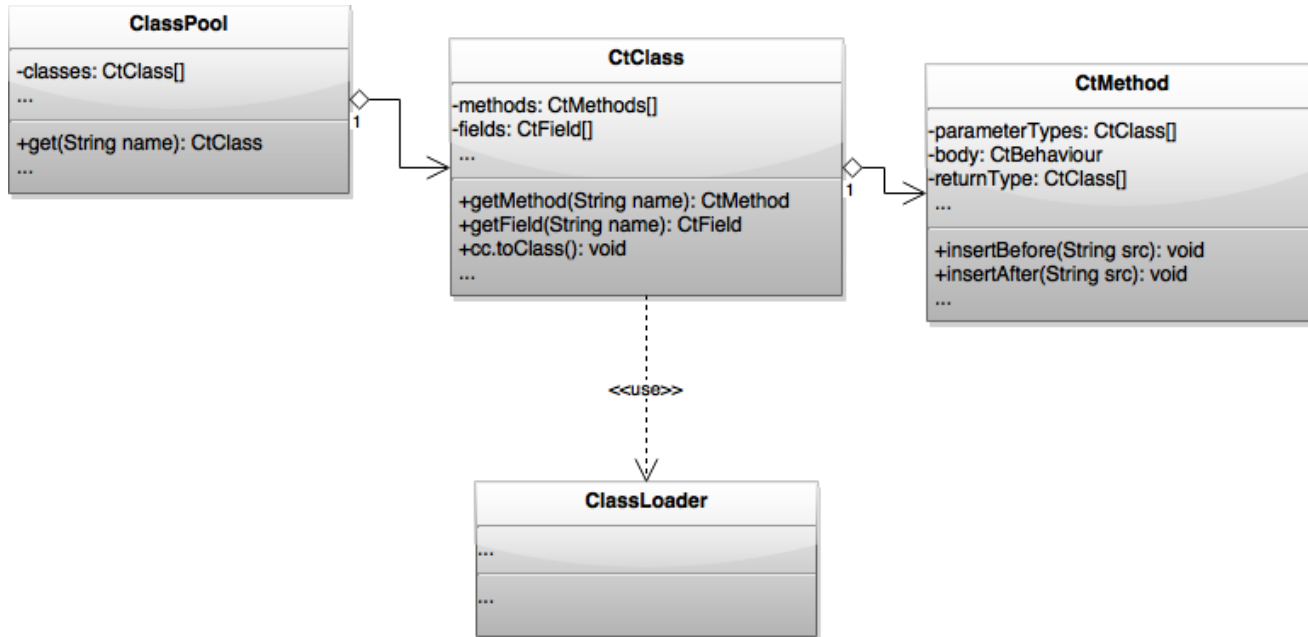
## Modify Java Bytecode

1. Directly modify bytecode
2. Write source code, compile on-the-fly, inject into existing bytecode

## Manual injection into Java Classloader

**Fault Injection -  
Java**

Sebastian  
Gerstenberg,  
07.09.2015  
Chart **15**



## Fault Injection - Java

Sebastian Gerstenberg,  
07.09.2015  
Chart 16



```
ClassPool pool = ClassPool.getDefault();

// create metamodel for class Math
CtClass cc = pool.get("de.hpi.fault_injection.Math");
// retrieve model for method get
CtMethod multiplyMethod = cc.getDeclaredMethod("multiply");

// modify: args[0] = 0
multiplyMethod.insertBefore("$1 = 0;");

// update the class file *injection*
cc.writeFile();

// tell classloader to directly load injected class
cc.toClass();
```



```
public int multiply(int y, int arg1) {
    0  iconst_0;
    1  istore_1;          /* x */
    /* L6 */
    2  iload_1;          /* x */
    3  iload_2;          /* y */
    4  imul;
    5  ireturn;
}

public static int multiplyCorrect(int x, int y) {
    /* L10 */
    0  iload_0;          /* x */
    1  iload_1;          /* y */
    2  imul;
    3  ireturn;
}
```

Java

Sebastian  
Gerstenberg,  
07.09.2015  
Chart 17

Thank you  
for your attention!

Speaker  
Job Description  
Institute

# Quellen

---

- <https://developer.jboss.org/> Wiki, How to
- <http://jboss-javassist.github.io/javassist/>
- <http://docs.oracle.com/javase/specs/jvms/se7/html/>
- <http://www.ic.unicamp.br/~eliane/JACA.html>
- Jaca: a reflective fault injection tool based on patterns, Dependable Systems and Networks, 2002. DSN 2002. Proceedings. International Conference on

## **Fault Injection - Java**

Sebastian  
Gerstenberg,  
07.09.2015  
Chart **19**