

Instrumentation

- Definition
- javaagent
- ClassFileTransformer
- Javassist

Instrumentation

Java programming language API for tools to instrument Java programming language applications.

Tools use `java.lang.instrument` to modify the class file that defines a class – generally, by inserting into the byte-code of methods additional byte-code which will perform the instrumentation.

Can be used to add code for annotated types and methods.

javaagent

javaagent is a JVM “plugin”, a specially crafted .jar file, that utilizes the Instrumentation API

Javaagent is a java application, which is started in JMV of your application before your application.

Start method

```
public class Agent {  
    public static void premain(String args, Instrumentation instrumentation) {  
        System.out.println("Classes loaded: " + instrumentation.getAllLoadedClasses().length);  
        instrumentation.addTransformer(new MyClassTransformer());  
    }  
}
```

Maven configuration

```
<plugin>
  <artifactId>maven-assembly-plugin</artifactId>
  <configuration>
    <descriptorRefs>
      <descriptorRef>jar-with-dependencies</descriptorRef>
    </descriptorRefs>
    <appendAssemblyId>>false</appendAssemblyId>
    <archive>
      <index>>true</index>
      <manifest>
        <addClasspath>>true</addClasspath>
        <mainClass>space.harbour.l151.Main</mainClass>
      </manifest>
      <manifestEntries>
        <Premain-Class>space.harbour.l151.Agent</Premain-Class>
      </manifestEntries>
    </archive>
  </configuration>
  <executions>
    <execution>
      <id>make-assembly</id>
      <phase>package</phase>
      <goals>
        <goal>single</goal>
      </goals>
    </execution>
  </executions>
</plugin>
```

Usage

```
java -javaagent:agent.jar -jar app.jar
```

java.lang.instrument.ClassFileTransformer

```
byte[]  
transform( ClassLoader      loader,  
           String          className,  
           Class<?>        classBeingRedefined,  
           ProtectionDomain protectionDomain,  
           byte[]          classfileBuffer) throws IllegalClassFormatException;
```

loader

the defining loader of the class to be transformed, null for the bootstrap loader

className

the name of the class in the internal form of fully qualified class and interface names

classBeingRedefined

if this is triggered by a redefine or retransform, the class being redefined or retransformed if this is a class load, **null**

protectionDomain

the protection domain of the class being defined or redefined

classfileBuffer

the input byte buffer

Javassist

Javassist is a bytecode manipulation library.

Javassist enables Java programs to define a new class at runtime and to modify a class file when the JVM loads it.

Maven

```
<dependency>  
  <groupId>javassist</groupId>  
  <artifactId>javassist</artifactId>  
  <version>3.12.1.GA</version>  
</dependency>
```

How it works

