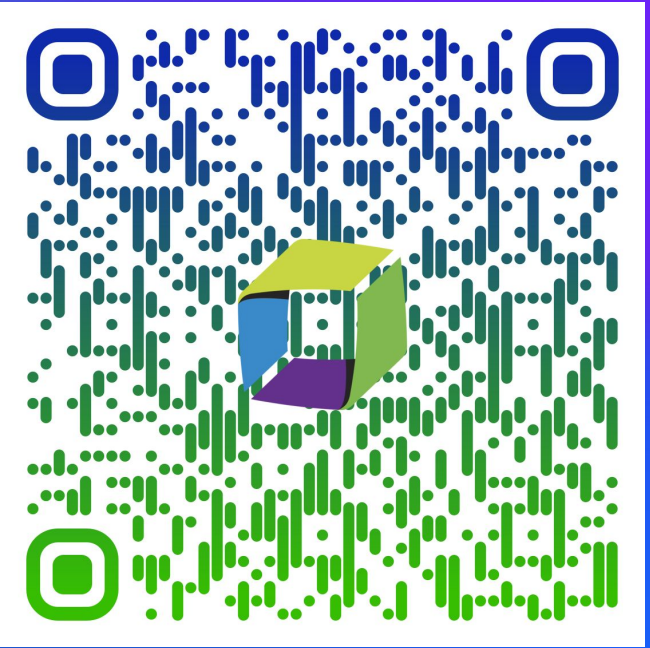
Who are we?

- Founded by 3 people in 2005 – now 4.000+ people (1.400+ engineers, 60+ offices globally)
- „A world where software works perfectly and securely“
- Automated, AI-powered, full-stack observability
- Target market: 15.000 biggest companies world-wide
- 70%+ of the Fortune 100 companies are our customers

dynatrace

Interested?



Senior Java Software Developer (m/f/x)

Senior Full Stack Developer with enthusiasm about Security (m/f/x)

careers.dynatrace.com

Profiling and code improvements using open-source tooling

Belk Stefan
Poulter Andrew

```java
package org.example;

public class Main {
    public static void main(String[] args) {
        int number = 5;
        System.out.println(String.format("%s + 2 = %s", number, addTwo(number)));
    }

    public static Integer addTwo(int i) {
        try {
            int two = 2;
            return i - two;
        } catch (Exception e) {
            throw new RuntimeException(e);
        }
    }
}
```

```java
package org.example;

public class Main {
    public static void main(String[] args) {
        int number = 5;
        System.out.println(String.format("%s + 2 = %s", number, addTwo(number)));
    }

    public static Integer addTwo(int i) {
        try {
            int two = 2;
            return i - two;
        } catch (Exception e) {
            throw new RuntimeException(e);
        }
    }
}
```

```java
package org.example;

public class Main {
    public static void main(String[] args) {
        int number = 5;
        System.out.println(String.format("%s + 2 = %s", number, addTwo(number)));
    }

    public static Integer addTwo(int i) {
        try {
            int two = 2;
            return i - two;
        } catch (Exception e) {
            throw new RuntimeException(e);
        }
    }
}
```

```java
package org.example;

public class Main {
    public static void main(String[] args) {
        int number = 5;
        System.out.println(String.format("%s + 2 = %s", number, addTwo(number)));
    }

    public static Integer addTwo(int i) {
        try {
            int two = 2;
            return i - two;
        } catch (Exception e) {
            throw new RuntimeException(e);
        }
    }
}
```

How do you debug a debugger?

Java Platform Debugger Architecture (JPDA)

Java Platform Debugger Architecture (JPDA)

- Java Debug Interface (JDI)

Java Platform Debugger Architecture (JPDA)

- Java Debug Interface (JDI)
- Java Debug Wire Protocol (JDWP)

Java Platform Debugger Architecture (JPDA)

- Java Debug Interface (JDI)
- Java Debug Wire Protocol (JDWP)
- Java Virtual Machine Tool Interface (JVM TI)

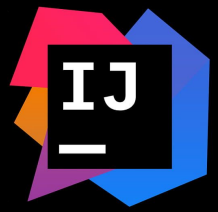
“You might want to use JVM TI if your tool has very specialized functionality, not available in JDWP/JDI, that can only be performed in the debuggee process and you have a lot of time on your hands.”

https://www.oracle.com/java/technologies/javase/java-platform-debugger-architecture.html

Debugger Frontend (JDI)

Communication layer

Debugged JVM (JVMTI)



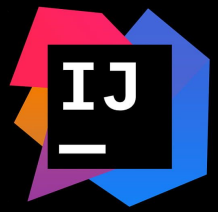
Now what?

```
System.out.println(“TEST”);
```

Debugger Frontend (JDI)

Communication layer

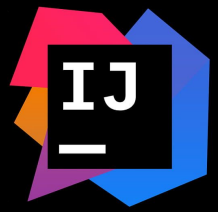
Debugged JVM (JVMTI)



Debugger Frontend (JDI)

Communication layer

Debugged JVM (JVMTI)

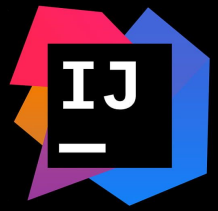


Debugger Frontend (JDI)

Intellij with patched JDI

Communication layer

Debugged JVM (JVMTI)



```
com.jetbrains.jdi.JDWPException.toJDIException(JDWPException.java:89)
com.jetbrains.jdi.StackFrameImpl.getSlotsValues(StackFrameImpl.java:302)
com.jetbrains.jdi.StackFrameImpl.getValues(StackFrameImpl.java:345)
com.intellij.debugger.jdi.StackFrameProxyImpl.getAllValues(StackFrameProxyImpl.java:450)
com.intellij.debugger.jdi.StackFrameProxyImpl.getValue(StackFrameProxyImpl.java:391)
com.intellij.debugger.ui.impl.watch.LocalVariableDescriptorImpl.calcValue(LocalVariableDescriptorImpl.java:73)
com.intellij.debugger.ui.impl.watch.ValueDescriptorImpl.setContext(ValueDescriptorImpl.java:214)
com.intellij.debugger.engine.JavaValue$1.contextAction(JavaValue.java:175)
com.intellij.debugger.engine.events.SuspendContextCommandImpl.action(SuspendContextCommandImpl.java:57)
com.intellij.debugger.engine.events.DebuggerCommandImpl.run(DebuggerCommandImpl.java:43)
com.intellij.debugger.engine.DebuggerManagerThreadImpl.processEvent(DebuggerManagerThreadImpl.java:155)
com.intellij.debugger.engine.DebuggerManagerThreadImpl.processEvent(DebuggerManagerThreadImpl.java:30)
com.intellij.debugger.impl.InvokeThread.lambda$run$0(InvokeThread.java:141)
com.intellij.openapi.progress.impl.CoreProgressManager.lambda$runProcess$2(CoreProgressManager.java:188)
com.intellij.openapi.progress.impl.CoreProgressManager.lambda$executeProcessUnderProgress$12(CoreProgressManager.java:608)
com.intellij.openapi.progress.impl.CoreProgressManager.registerIndicatorAndRun(CoreProgressManager.java:683)
com.intellij.openapi.progress.impl.CoreProgressManager.computeUnderProgress(CoreProgressManager.java:639)
com.intellij.openapi.progress.impl.CoreProgressManager.executeProcessUnderProgress(CoreProgressManager.java:607)
com.intellij.openapi.progress.impl.ProgressManagerImpl.executeProcessUnderProgress(ProgressManagerImpl.java:60)
com.intellij.openapi.progress.impl.CoreProgressManager.runProcess(CoreProgressManager.java:175)
com.intellij.debugger.impl.InvokeThread.lambda$run$1(InvokeThread.java:128)
com.intellij.openapi.project.DumbService.runWithAlternativeResolveEnabled(DumbService.jav:417)
com.intellij.debugger.impl.InvokeThread.run(InvokeThread.java:128)
com.intellij.debugger.impl.InvokeThread$WorkerThreadRequest.lambda$run$0(InvokeThread.java:52)
com.intellij.util.ConcurrencyUtil.runUnderThreadName(ConcurrencyUtil.java:212)
com.intellij.debugger.impl.InvokeThread$WorkerThreadRequest.run(InvokeThread.java:51)
com.intellij.openapi.application.impl.ApplicationImpl$1.run(ApplicationImpl.java:297)
java.base/java.util.concurrent.Executors$RunnableAdapter.call(Executors.java:539)
```

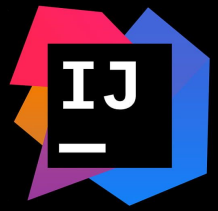
```
com.jetbrains.jdi.JDWPException.toJDIException(JDWPException.java:89)
com.jetbrains.jdi.StackFrameImpl.getSlotsValues(StackFrameImpl.java:302)
com.jetbrains.jdi.StackFrameImpl.getValues(StackFrameImpl.java:345)
com.intellij.debugger.jdi.StackFrameProxyImpl.getAllValues(StackFrameProxyImpl.java:450)
com.intellij.debugger.jdi.StackFrameProxyImpl.getValue(StackFrameProxyImpl.java:391)
com.intellij.debugger.ui.impl.watch.LocalVariableDescriptorImpl.calcValue(LocalVariableDescriptorImpl.java:73)
com.intellij.debugger.ui.impl.watch.ValueDescriptorImpl.setContext(ValueDescriptorImpl.java:214)
com.intellij.debugger.engine.JavaValue$1.contextAction(JavaValue.java:175)
com.intellij.debugger.engine.events.SuspendContextCommandImpl.action(SuspendContextCommandImpl.java:57)
com.intellij.debugger.engine.events.DebuggerCommandImpl.run(DebuggerCommandImpl.java:43)
com.intellij.debugger.engine.DebuggerManagerThreadImpl.processEvent(DebuggerManagerThreadImpl.java:155)
com.intellij.debugger.engine.DebuggerManagerThreadImpl.processEvent(DebuggerManagerThreadImpl.java:30)
com.intellij.debugger.impl.InvokeThread.lambda$run$0(InvokeThread.java:141)
com.intellij.openapi.progress.impl.CoreProgressManager.lambda$runProcess$2(CoreProgressManager.java:188)
com.intellij.openapi.progress.impl.CoreProgressManager.lambda$executeProcessUnderProgress$12(CoreProgressManager.java:608)
com.intellij.openapi.progress.impl.CoreProgressManager.registerIndicatorAndRun(CoreProgressManager.java:683)
com.intellij.openapi.progress.impl.CoreProgressManager.computeUnderProgress(CoreProgressManager.java:639)
com.intellij.openapi.progress.impl.CoreProgressManager.executeProcessUnderProgress(CoreProgressManager.java:607)
com.intellij.openapi.progress.impl.ProgressManagerImpl.executeProcessUnderProgress(ProgressManagerImpl.java:60)
com.intellij.openapi.progress.impl.CoreProgressManager.runProcess(CoreProgressManager.java:175)
com.intellij.debugger.impl.InvokeThread.lambda$run$1(InvokeThread.java:128)
com.intellij.openapi.project.DumbService.runWithAlternativeResolveEnabled(DumbService.jav:417)
com.intellij.debugger.impl.InvokeThread.run(InvokeThread.java:128)
com.intellij.debugger.impl.InvokeThread$WorkerThreadRequest.lambda$run$0(InvokeThread.java:52)
com.intellij.util.ConcurrencyUtil.runUnderThreadName(ConcurrencyUtil.java:212)
com.intellij.debugger.impl.InvokeThread$WorkerThreadRequest.run(InvokeThread.java:51)
com.intellij.openapi.application.impl.ApplicationImpl$1.run(ApplicationImpl.java:297)
java.base/java.util.concurrent.Executors$RunnableAdapter.call(Executors.java:539)
```

Debugger Frontend (JDI)

Intellij with patched JDI

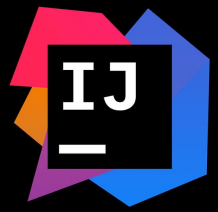
Communication layer

Debugged JVM (JVMTI)



Debugger Frontend (JDI)

~~Intellij with patched JDI~~

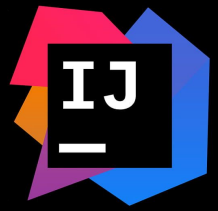


Communication layer

Debugged JVM (JVMTI)

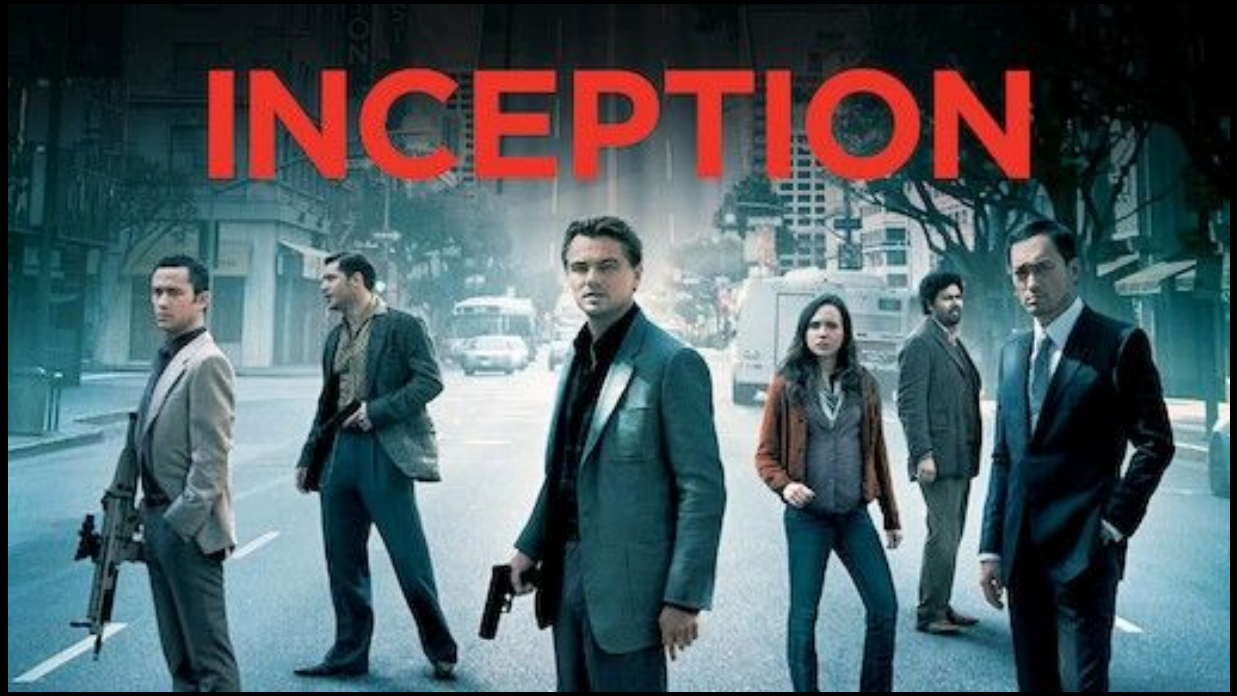
Debugger Frontend (JDI)

~~Intellij with patched JDI~~
Local build IDE in debugging mode



Communication layer

Debugged JVM (JVMTI)


INCEPTION

StackFrame Command Set (16)

GetValues Command (1)

Type	Name	Description
threadID	thread	The frame's thread.
frameID	frame	The frame ID.
int	slots	The number of values to get.
Repeated slots times:		
- int	slot	The local variable's index in the frame.
- byte	sigbyte	A tag identifying the type of the variable

https://docs.oracle.com/en/java/javase/17/docs/specs/jdwp/jdwp-protocol.html#JDWP_StackFrame_GetValues

Tag Constants

Name	Value	Description
ARRAY	91	'[' - an array object (objectID size).
BYTE	66	'B' - a byte value (1 byte).
CHAR	67	'C' - a character value (2 bytes).
OBJECT	76	'L' - an object (objectID size).
FLOAT	70	'F' - a float value (4 bytes).
DOUBLE	68	'D' - a double value (8 bytes).
INT	73	'I' - an int value (4 bytes).

Tools to use:

Tools to use:

- ASM Bytecode Viewer (intellij plugin)

Tools to use:

- ASM Bytecode Viewer (intellij plugin)
- jclasslib Bytecode Viewer (intellij plugin)

Tools to use:

- ASM Bytecode Viewer (intellij plugin)
- jclasslib Bytecode Viewer (intellij plugin)
- javap (part of the JDK)

```
public class org.example.Main
  minor version: 0
  major version: 61
  flags: (0x0021) ACC_PUBLIC, ACC_SUPER
  this_class: #22                         // org/example/Main
  super_class: #2                         // java/lang/Object
  interfaces: 0, fields: 0, methods: 3, attributes: 1
```

```
public class org.example.Main
  minor version: 0
  major version: 61
  flags: (0x0021) ACC_PUBLIC, ACC_SUPER
  this_class: #22                         // org/example/Main
  super_class: #2                         // java/lang/Object
  interfaces: 0, fields: 0, methods: 3, attributes: 1
```

```
public class org.example.Main
  minor version: 0
  major version: 61
  flags: (0x0021) ACC_PUBLIC, ACC_SUPER
  this_class: #22                         // org/example/Main
  super_class: #2                         // java/lang/Object
  interfaces: 0, fields: 0, methods: 3, attributes: 1
```

```
public class org.example.Main
  minor version: 0
  major version: 61
  flags: (0x0021) ACC_PUBLIC, ACC_SUPER
  this_class: #22                         // org/example/Main
  super_class: #2                         // java/lang/Object
  interfaces: 0, fields: 0, methods: 3, attributes: 1
```

```
Constant pool:
   #1 = Methodref          #2.#3          // java/lang/Object."<init>":()V
   #2 = Class              #4             // java/lang/Object
   #3 = NameAndType        #5:#6          // "<init>":()V
   #4 = Utf8               java/lang/Object
   #5 = Utf8               <init>
   #6 = Utf8               ()V
   #7 = Fieldref           #8.#9          // java/lang/System.out:Ljava/io/PrintStream;
   #8 = Class              #10            // java/lang/System
   #9 = NameAndType        #11:#12        // out:Ljava/io/PrintStream;
  #10 = Utf8               java/lang/System
  #11 = Utf8               out
  #12 = Utf8               Ljava/io/PrintStream;
  #13 = String             #14            // %s + 2 = %s
  #14 = Utf8               %s + 2 = %s
  #15 = Methodref          #16.#17        // java/lang/Integer.valueOf:(I)Ljava/lang/Integer;
  #16 = Class              #18            // java/lang/Integer
  #17 = NameAndType        #19:#20        // valueOf:(I)Ljava/lang/Integer;
  #18 = Utf8               java/lang/Integer
  #19 = Utf8               valueOf
  #20 = Utf8               (I)Ljava/lang/Integer;
  #21 = Methodref          #22.#23        // org/example/Main.addTwo:(I)Ljava/lang/Integer;
  #22 = Class              #24            // org/example/Main
  #23 = NameAndType        #25:#20        // addTwo:(I)Ljava/lang/Integer;
  #24 = Utf8               org/example/Main
  #25 = Utf8               addTwo
  #26 = Methodref          #27.#28        // java/lang/String.format:(Ljava/lang/String;[Ljava/lang/Object;)Ljava/lang/String;
```

```
public org.example.Main();
  descriptor: ()V
  flags: (0x0001) ACC_PUBLIC
  Code:
    stack=1, locals=1, args_size=1
       0: aload_0
       1: invokespecial #1                  // Method java/lang/Object."<init>":()V
       4: return
    LineNumberTable:
      line 3: 0
    LocalVariableTable:
      Start  Length  Slot  Name   Signature
          0       5     0  this   Lorg/example/Main;
```

```
public org.example.Main();
  descriptor: ()V
  flags: (0x0001) ACC_PUBLIC
  Code:
    stack=1, locals=1, args_size=1
       0: aload_0
       1: invokespecial #1                  // Method java/lang/Object."<init>":()V
       4: return
    LineNumberTable:
      line 3: 0
    LocalVariableTable:
      Start  Length  Slot  Name   Signature
          0       5     0  this   Lorg/example/Main;
```

```
public org.example.Main();
  descriptor: ()V
  flags: (0x0001) ACC_PUBLIC
  Code:
    stack=1, locals=1, args_size=1
       0: aload_0
       1: invokespecial #1                  // Method java/lang/Object."<init>":()V
       4: return
    LineNumberTable:
      line 3: 0
    LocalVariableTable:
      Start  Length  Slot  Name   Signature
          0       5     0  this   Lorg/example/Main;
```

```
public org.example.Main();
  descriptor: ()V
  flags: (0x0001) ACC_PUBLIC
  Code:
    stack=1, locals=1, args_size=1
       0: aload_0
       1: invokespecial #1                  // Method java/lang/Object."<init>":()V
       4: return
    LineNumberTable:
      line 3: 0
    LocalVariableTable:
      Start  Length  Slot  Name   Signature
          0       5     0  this   Lorg/example/Main;
```

```
public static void main(java.lang.String[]);
  descriptor: ([Ljava/lang/String;)V
  flags: (0x0009) ACC_PUBLIC, ACC_STATIC
  Code:
    stack=6, locals=2, args_size=1
       0: iconst_5
       1: istore_1
       2: getstatic     #7                  // Field java/lang/System.out:Ljava/io/PrintStream;
       5: ldc           #13                 // String %s + 2 = %s
       7: iconst_2
       8: anewarray     #2                  // class java/lang/Object
      11: dup
      12: iconst_0
      13: iload_1
      14: invokestatic  #15                 // Method java/lang/Integer.valueOf:(I)Ljava/lang/Integer;
      17: aastore
      18: dup
      19: iconst_1
      20: iload_1
      21: invokestatic  #21                 // Method addTwo:(I)Ljava/lang/Integer;
      24: aastore
      25: invokestatic  #26                 // Method java/lang/String.format:.........
      28: invokevirtual #32                 // Method java/io/PrintStream.println:(Ljava/lang/String;)V
      31: return
    LineNumberTable:
      line 5: 0
      line 6: 2
      line 7: 31
    LocalVariableTable:
      Start  Length  Slot  Name   Signature
          0      32     0  args   [Ljava/lang/String;
          2      30     1 number   I
```

```
public static void main(java.lang.String[]);
  descriptor: ([Ljava/lang/String;)V
  flags: (0x0009) ACC_PUBLIC, ACC_STATIC
  Code:
    stack=6, locals=2, args_size=1
       0: iconst_5
       1: istore_1
       2: getstatic     #7                  // Field java/lang/System.out:Ljava/io/PrintStream;
       5: ldc           #13                 // String %s + 2 = %s
       7: iconst_2
       8: anewarray     #2                  // class java/lang/Object
      11: dup
      12: iconst_0
      13: iload_1
      14: invokestatic  #15                 // Method java/lang/Integer.valueOf:(I)Ljava/lang/Integer;
      17: aastore
      18: dup
      19: iconst_1
      20: iload_1
      21: invokestatic  #21                 // Method addTwo:(I)Ljava/lang/Integer;
      24: aastore
      25: invokestatic  #26                 // Method java/lang/String.format:.........
      28: invokevirtual #32                 // Method java/io/PrintStream.println:(Ljava/lang/String;)V
      31: return
    LineNumberTable:
      line 5: 0
      line 6: 2
      line 7: 31
    LocalVariableTable:
      Start  Length  Slot  Name   Signature
          0      32     0  args   [Ljava/lang/String;
          2      30     1 number   I
```

```
public static void main(java.lang.String[]);
  descriptor: ([Ljava/lang/String;)V
  flags: (0x0009) ACC_PUBLIC, ACC_STATIC
  Code:
    stack=6, locals=2, args_size=1
       0: iconst_5
       1: istore_1
       2: getstatic     #7                  // Field java/lang/System.out:Ljava/io/PrintStream;
       5: ldc           #13                 // String %s + 2 = %s
       7: iconst_2
       8: anewarray     #2                  // class java/lang/Object
      11: dup
      12: iconst_0
      13: iload_1
      14: invokestatic  #15                 // Method java/lang/Integer.valueOf:(I)Ljava/lang/Integer;
      17: aastore
      18: dup
      19: iconst_1
      20: iload_1
      21: invokestatic  #21                 // Method addTwo:(I)Ljava/lang/Integer;
      24: aastore
      25: invokestatic  #26                 // Method java/lang/String.format:.........
      28: invokevirtual #32                 // Method java/io/PrintStream.println:(Ljava/lang/String;)V
      31: return
    LineNumberTable:
      line 5: 0
      line 6: 2
      line 7: 31
    LocalVariableTable:
      Start  Length  Slot  Name   Signature
          0      32     0  args   [Ljava/lang/String;
          2      30     1 number   I
```

```
Constant pool:
   #1 = Methodref          #2.#3          // java/lang/Object."<init>":()V
   #2 = Class              #4             // java/lang/Object
   #3 = NameAndType        #5:#6          // "<init>":()V
   #4 = Utf8               java/lang/Object
   #5 = Utf8               <init>
   #6 = Utf8               ()V
   #7 = Fieldref           #8.#9          // java/lang/System.out:Ljava/io/PrintStream;
   #8 = Class              #10            // java/lang/System
   #9 = NameAndType        #11:#12        // out:Ljava/io/PrintStream;
  #10 = Utf8               java/lang/System
  #11 = Utf8               out
  #12 = Utf8               Ljava/io/PrintStream;
  #13 = String             #14            // %s + 2 = %s
  #14 = Utf8               %s + 2 = %s
  #15 = Methodref          #16.#17        // java/lang/Integer.valueOf:(I)Ljava/lang/Integer;
  #16 = Class              #18            // java/lang/Integer
  #17 = NameAndType        #19:#20        // valueOf:(I)Ljava/lang/Integer;
  #18 = Utf8               java/lang/Integer
  #19 = Utf8               valueOf
  #20 = Utf8               (I)Ljava/lang/Integer;
  #21 = Methodref          #22.#23        // org/example/Main.addTwo:(I)Ljava/lang/Integer;
  #22 = Class              #24            // org/example/Main
  #23 = NameAndType        #25:#20        // addTwo:(I)Ljava/lang/Integer;
  #24 = Utf8               org/example/Main
  #25 = Utf8               addTwo
  #26 = Methodref          #27.#28        // java/lang/String.format:(Ljava/lang/String;[Ljava/lang/Object;)Ljava/lang/String;
```

```
Constant pool:
   #1 = Methodref          #2.#3          // java/lang/Object."<init>":()V
   #2 = Class              #4             // java/lang/Object
   #3 = NameAndType        #5:#6          // "<init>":()V
   #4 = Utf8               java/lang/Object
   #5 = Utf8               <init>
   #6 = Utf8               ()V
   #7 = Fieldref           #8.#9          // java/lang/System.out:Ljava/io/PrintStream;
   #8 = Class              #10            // java/lang/System
   #9 = NameAndType        #11:#12        // out:Ljava/io/PrintStream;
  #10 = Utf8               java/lang/System
  #11 = Utf8               out
  #12 = Utf8               Ljava/io/PrintStream;
  #13 = String             #14            // %s + 2 = %s
  #14 = Utf8               %s + 2 = %s
  #15 = Methodref          #16.#17        // java/lang/Integer.valueOf:(I)Ljava/lang/Integer;
  #16 = Class              #18            // java/lang/Integer
  #17 = NameAndType        #19:#20        // valueOf:(I)Ljava/lang/Integer;
  #18 = Utf8               java/lang/Integer
  #19 = Utf8               valueOf
  #20 = Utf8               (I)Ljava/lang/Integer;
  #21 = Methodref          #22.#23        // org/example/Main.addTwo:(I)Ljava/lang/Integer;
  #22 = Class              #24            // org/example/Main
  #23 = NameAndType        #25:#20        // addTwo:(I)Ljava/lang/Integer;
  #24 = Utf8               org/example/Main
  #25 = Utf8               addTwo
  #26 = Methodref          #27.#28        // java/lang/String.format:(Ljava/lang/String;[Ljava/lang/Object;)Ljava/lang/String;
```

```
public static void main(java.lang.String[]);
  descriptor: ([Ljava/lang/String;)V
  flags: (0x0009) ACC_PUBLIC, ACC_STATIC
  Code:
    stack=6, locals=2, args_size=1
       0: iconst_5
       1: istore_1
       2: getstatic     #7                  // Field java/lang/System.out:Ljava/io/PrintStream;
       5: ldc           #13                 // String %s + 2 = %s
       7: iconst_2
       8: anewarray     #2                  // class java/lang/Object
      11: dup
      12: iconst_0
      13: iload_1
      14: invokestatic  #15                 // Method java/lang/Integer.valueOf:(I)Ljava/lang/Integer;
      17: aastore
      18: dup
      19: iconst_1
      20: iload_1
      21: invokestatic  #21                 // Method addTwo:(I)Ljava/lang/Integer;
      24: aastore
      25: invokestatic  #26                 // Method java/lang/String.format:.........
      28: invokevirtual #32                 // Method java/io/PrintStream.println:(Ljava/lang/String;)V
      31: return
    LineNumberTable:
      line 5: 0
      line 6: 2
      line 7: 31
    LocalVariableTable:
      Start  Length  Slot  Name   Signature
          0      32     0  args   [Ljava/lang/String;
          2      30     1 number   I
```

```
public static void main(java.lang.String[]);
  descriptor: ([Ljava/lang/String;)V
  flags: (0x0009) ACC_PUBLIC, ACC_STATIC
  Code:
    stack=6, locals=2, args_size=1
       0: iconst_5
       1: istore_1
       2: getstatic     #7                  // Field java/lang/System.out:Ljava/io/PrintStream;
       5: ldc           #13                 // String %s + 2 = %s
       7: iconst_2
       8: anewarray     #2                  // class java/lang/Object
      11: dup
      12: iconst_0
      13: iload_1
      14: invokestatic  #15                 // Method java/lang/Integer.valueOf:(I)Ljava/lang/Integer;
      17: aastore
      18: dup
      19: iconst_1
      20: iload_1
      21: invokestatic  #21                 // Method addTwo:(I)Ljava/lang/Integer;
      24: aastore
      25: invokestatic  #26                 // Method java/lang/String.format:.........
      28: invokevirtual #32                 // Method java/io/PrintStream.println:(Ljava/lang/String;)V
      31: return
    LineNumberTable:
      line 5: 0
      line 6: 2
      line 7: 31
    LocalVariableTable:
      Start  Length  Slot  Name   Signature
          0      32     0  args   [Ljava/lang/String;
          2      30     1 number   I
```

```
public static void main(java.lang.String[]);
  descriptor: ([Ljava/lang/String;)V
  flags: (0x0009) ACC_PUBLIC, ACC_STATIC
  Code:
    stack=6, locals=2, args_size=1
       0: iconst_5
       1: istore_1
       2: getstatic     #7                  // Field java/lang/System.out:Ljava/io/PrintStream;
       5: ldc           #13                 // String %s + 2 = %s
       7: iconst_2
       8: anewarray     #2                  // class java/lang/Object
      11: dup
      12: iconst_0
      13: iload_1
      14: invokestatic  #15                 // Method java/lang/Integer.valueOf:(I)Ljava/lang/Integer;
      17: aastore
      18: dup
      19: iconst_1
      20: iload_1
      21: invokestatic  #21                 // Method addTwo:(I)Ljava/lang/Integer;
      24: aastore
      25: invokestatic  #26                 // Method java/lang/String.format:.........
      28: invokevirtual #32                 // Method java/io/PrintStream.println:(Ljava/lang/String;)V
      31: return
    LineNumberTable:
      line 5: 0
      line 6: 2
      line 7: 31
    LocalVariableTable:
      Start  Length  Slot  Name   Signature
          0      32     0  args   [Ljava/lang/String;
          2      30     1 number   I
```

```
public static java.lang.Integer addTwo(int);
  descriptor: (I)Ljava/lang/Integer;
  flags: (0x0009) ACC_PUBLIC, ACC_STATIC
  Code:
    stack=3, locals=2, args_size=1
       0: iconst_2
       1: istore_1
       2: iload_0
       3: iload_1
       4: isub
       5: invokestatic  #15                 // Method java/lang/Integer.valueOf:(I)Ljava/lang/Integer;
       8: areturn
       9: astore_1
      10: new           #40                 // class java/lang/RuntimeException
      13: dup
      14: aload_1
      15: invokespecial #42                 // Method java/lang/RuntimeException."<init>":(Ljava/lang/Throwable;)V
      18: athrow
    Exception table:
       from    to  target type
           0     8     9   Class java/lang/Exception
    LineNumberTable:
      line 11: 0
      line 12: 2
      line 13: 9
      line 14: 10
    LocalVariableTable:
      Start  Length  Slot  Name   Signature
          2       7     1   two    I
         10       9     1     e    Ljava/lang/Exception;
          0      19     0     i    I
    StackMapTable: number_of_entries = 1
      frame_type = 73 /* same_locals_1_stack_item */
        stack = [ class java/lang/Exception ]
```

```
public static java.lang.Integer addTwo(int);
  descriptor: (I)Ljava/lang/Integer;
  flags: (0x0009) ACC_PUBLIC, ACC_STATIC
  Code:
    stack=3, locals=2, args_size=1
       0: iconst_2
       1: istore_1
       2: iload_0
       3: iload_1
       4: isub
       5: invokestatic  #15                 // Method java/lang/Integer.valueOf:(I)Ljava/lang/Integer;
       8: areturn
       9: astore_1
      10: new           #40                 // class java/lang/RuntimeException
      13: dup
      14: aload_1
      15: invokespecial #42                 // Method java/lang/RuntimeException."<init>":(Ljava/lang/Throwable;)V
      18: athrow
    Exception table:
       from    to  target type
           0     8     9   Class java/lang/Exception
    LineNumberTable:
      line 11: 0
      line 12: 2
      line 13: 9
      line 14: 10
    LocalVariableTable:
      Start  Length  Slot  Name   Signature
          2       7     1   two    I
         10       9     1     e    Ljava/lang/Exception;
          0      19     0     i    I
    StackMapTable: number_of_entries = 1
      frame_type = 73 /* same_locals_1_stack_item */
        stack = [ class java/lang/Exception ]
```

```
public static java.lang.Integer addTwo(int);
  descriptor: (I)Ljava/lang/Integer;
  flags: (0x0009) ACC_PUBLIC, ACC_STATIC
  Code:
    stack=3, locals=2, args_size=1
       0: iconst_2
       1: istore_1
       2: iload_0
       3: iload_1
       4: isub
       5: invokestatic  #15                 // Method java/lang/Integer.valueOf:(I)Ljava/lang/Integer;
       8: areturn
       9: astore_1
      10: new           #40                 // class java/lang/RuntimeException
      13: dup
      14: aload_1
      15: invokespecial #42                 // Method java/lang/RuntimeException."<init>":(Ljava/lang/Throwable;)V
      18: athrow
    Exception table:
       from    to  target type
           0     8     9   Class java/lang/Exception
    LineNumberTable:
      line 11: 0
      line 12: 2
      line 13: 9
      line 14: 10
    LocalVariableTable:
      Start  Length  Slot  Name   Signature
          2       7     1   two    I
         10       9     1     e    Ljava/lang/Exception;
          0      19     0     i    I
    StackMapTable: number_of_entries = 1
      frame_type = 73 /* same_locals_1_stack_item */
        stack = [ class java/lang/Exception ]
```

Try

```
public static java.lang.Integer addTwo(int);
                                  ng/Integer;
                                  .IC, ACC_STATIC

    stack=3, locals=2, args_size=1
       0: iconst_2
       1: istore_1
       2: iload_0
       3: iload_1
       4: isub
       5: invokestatic  #15                 // Method java/lang/Integer.valueOf:(I)Ljava/lang/Integer;
       8: areturn
       9: astore_1
      10: new           #40                 // class java/lang/RuntimeException
      13: dup
      14: aload_1
      15: invokespecial #42                 // Method java/lang/RuntimeException."<init>":(Ljava/lang/Throwable;)V
      18: athrow
    Exception table:
       from    to  target type
           0     8     9   Class java/lang/Exception
    LineNumberTable:
      line 11: 0
      line 12: 2
      line 13: 9
      line 14: 10
    LocalVariableTable:
      Start  Length  Slot  Name   Signature
          2       7     1   two    I
         10       9     1     e    Ljava/lang/Exception;
          0      19     0     i    I
    StackMapTable: number_of_entries = 1
      frame_type = 73 /* same_locals_1_stack_item */
        stack = [ class java/lang/Exception ]
```

```
public static java.lang.Integer addTwo(int);
  descriptor: (I)Ljava/lang/Integer;
  flags: (0x0009) ACC_PUBLIC, ACC_STATIC
  Code:
    stack=3, locals=2, args_size=1
       0: iconst_2
       1: istore_1
```

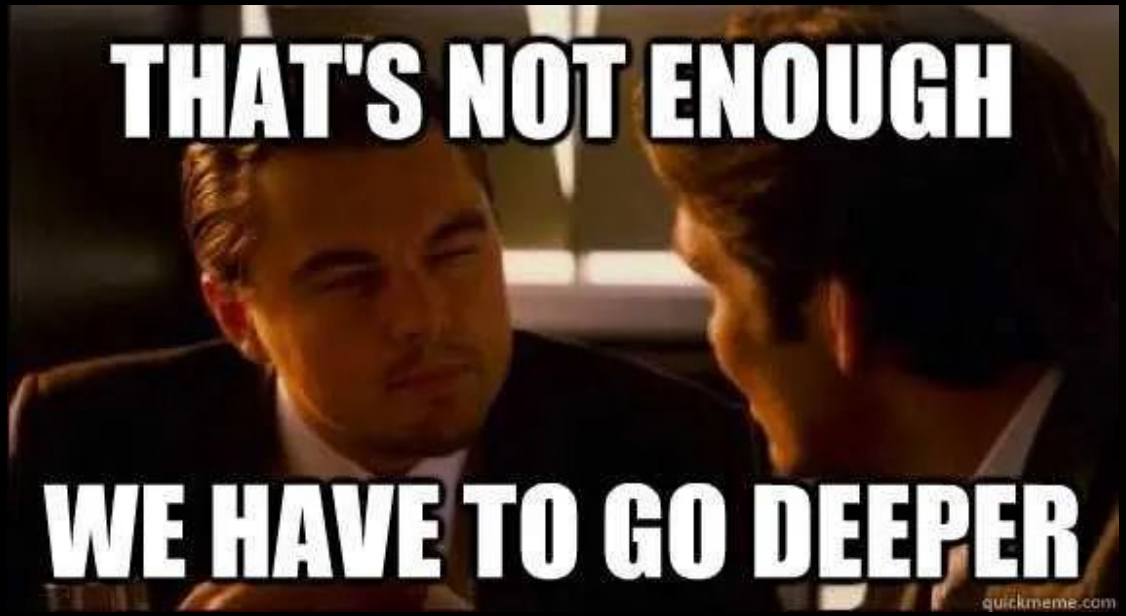
catch

```
       5: invokestatic  #15                 // Method java/lang/Integer.valueOf:(I)Ljava/lang/Integer;
       8: areturn
       9: astore_1
      10: new           #40                 // class java/lang/RuntimeException
      13: dup
      14: aload_1
      15: invokespecial #42                 // Method java/lang/RuntimeException."<init>":(Ljava/lang/Throwable;)V
      18: athrow
    Exception table:
       from    to  target type
           0     8     9   Class java/lang/Exception
    LineNumberTable:
      line 11: 0
      line 12: 2
      line 13: 9
      line 14: 10
    LocalVariableTable:
      Start  Length  Slot  Name   Signature
          2       7     1   two    I
         10       9     1     e    Ljava/lang/Exception;
          0      19     0     i    I
    StackMapTable: number_of_entries = 1
      frame_type = 73 /* same_locals_1_stack_item */
        stack = [ class java/lang/Exception ]
```

```
public static java.lang.Integer addTwo(int);
  descriptor: (I)Ljava/lang/Integer;
  flags: (0x0009) ACC_PUBLIC, ACC_STATIC
  Code:
    stack=3, locals=2, args_size=1
       0: iconst_2
       1: istore_1
       2: iload_0
       3: iload_1
       4: isub
       5: invokestatic  #15                 // Method java/lang/Integer.valueOf:(I)Ljava/lang/Integer;
       8: areturn
       9: astore_1
      10: new           #40                 // class java/lang/RuntimeException
      13: dup
      14: aload_1
      15: invokespecial #42                 // Method java/lang/RuntimeException."<init>":(Ljava/lang/Throwable;)V
      18: athrow
    Exception table:
       from    to  target type
           0     8     9   Class java/lang/Exception
    LineNumberTable:
      line 11: 0
      line 12: 2
      line 13: 9
      line 14: 10
    LocalVariableTable:
      Start  Length  Slot  Name   Signature
          2       7     1   two   I
         10       9     1     e   Ljava/lang/Exception;
          0      19     0     i   I
    StackMapTable: number_of_entries = 1
      frame_type = 73 /* same_locals_1_stack_item */
        stack = [ class java/lang/Exception ]
```

```
public static java.lang.Integer addTwo(int);
  descriptor: (I)Ljava/lang/Integer;
  flags: (0x0009) ACC_PUBLIC, ACC_STATIC
  Code:
    stack=3, locals=2, args_size=1
       0: iconst_2
       1: istore_1
       2: iload_0
       3: iload_1
       4: isub
       5: invokestatic  #15                 // Method java/lang/Integer.valueOf:(I)Ljava/lang/Integer;
       8: areturn
       9: astore_1
      10: new           #40                 // class java/lang/RuntimeException
      13: dup
      14: aload_1
      15: invokespecial #42                 // Method java/lang/RuntimeException."<init>":(Ljava/lang/Throwable;)V
      18: athrow
    Exception table:
       from    to  target type
           0     8     9   Class java/lang/Exception
    LineNumberTable:
      line 11: 0
      line 12: 2
      line 13: 9
      line 14: 10
    LocalVariableTable:
      Start  Length  Slot  Name   Signature
          2       7     1   two    I
         10       9     1     e    Ljava/lang/Exception;
          0      19     0     i    I
    StackMapTable: number_of_entries = 1
      frame_type = 73 /* same_locals_1_stack_item */
        stack = [ class java/lang/Exception ]
```

```
public static java.lang.Integer addTwo(int);
  descriptor: (I)Ljava/lang/Integer;
  flags: (0x0009) ACC_PUBLIC, ACC_STATIC
  Code:
    stack=3, locals=2, args_size=1
       0: iconst_2
       1: istore_1
       2: iload_0
       3: iload_1
       4: isub
       5: invokestatic  #15                 // Method java/lang/Integer.valueOf:(I)Ljava/lang/Integer;
       8: areturn
       9: astore_1
      10: new           #40                 // class java/lang/RuntimeException
      13: dup
      14: aload_1
      15: invokespecial #42                 // Method java/lang/RuntimeException."<init>":(Ljava/lang/Throwable;)V
      18: athrow
    Exception table:
       from    to  target type
           0     8     9   Class java/lang/Exception
    LineNumberTable:
      line 11: 0
      line 12: 2
      line 13: 9
      line 14: 10
    LocalVariableTable:
      Start  Length  Slot  Name   Signature
          2       7     1   two   I
         10       9     1     e   Ljava/lang/Exception;
          0      19     0     i   I
    StackMapTable: number_of_entries = 1
      frame_type = 73 /* same_locals_1_stack_item */
        stack = [ class java/lang/Exception ]
```


THAT'S NOT ENOUGH
WE HAVE TO GO DEEPER
quickmeme.com

Java Platform Debugger Architecture (JPDA)

- Java Debug Interface (JDI)
- Java Debug Wire Protocol (JDWP)
- Java Virtual Machine Tool Interface (JVM TI)

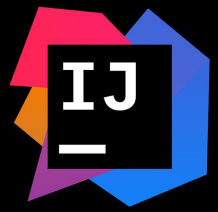
Java Platform Debugger Architecture (JPDA)

- Java Debug Interface (JDI)
- Java Debug Wire Protocol (JDWP)
- Java Virtual Machine Tool Interface (JVM TI)

Debugger Frontend (JDI)

Communication layer

Debugged JVM (JVMTI)



Debugger Frontend (JDI)

Communication layer

Debugged JVM (JVMTI)

Debugger Frontend (JDI)

Communication layer

Debugged JVM (JVMTI)

Patched openJDK
More logs from JVMTI

```
enum BasicType {
// The values T_BOOLEAN..T_LONG (4..11) are derived from the JVMS.
  T_BOOLEAN     = JVM_T_BOOLEAN,    //  4
  T_CHAR        = JVM_T_CHAR,       //  5
  T_FLOAT       = JVM_T_FLOAT,      //  6
  T_DOUBLE      = JVM_T_DOUBLE,     //  7
  T_BYTE        = JVM_T_BYTE,       //  8
  T_SHORT       = JVM_T_SHORT,      //  9
  T_INT         = JVM_T_INT,        // 10
  T_LONG        = JVM_T_LONG,       // 11
  T_OBJECT      = 12,
  T_ARRAY       = 13,
  ......
};
```

```
enum BasicType {
// The values T_BOOLEAN..T_LONG (4..11) are derived from the JVMS.
  T_BOOLEAN     = JVM_T_BOOLEAN,    //  4
  T_CHAR        = JVM_T_CHAR,       //  5
  T_FLOAT       = JVM_T_FLOAT,      //  6
  T_DOUBLE      = JVM_T_DOUBLE,     //  7
  T_BYTE        = JVM_T_BYTE,       //  8
  T_SHORT       = JVM_T_SHORT,      //  9
  T_INT         = JVM_T_INT,        // 10
  T_LONG        = JVM_T_LONG,       // 11
  T_OBJECT      = 12,
  T_ARRAY       = 13,
  ......
};
```

```
public static java.lang.Integer addTwo(int);
  descriptor: (I)Ljava/lang/Integer;
  flags: (0x0009) ACC_PUBLIC, ACC_STATIC
  Code:
    stack=3, locals=2, args_size=1
       0: iconst_2
       1: istore_1
       2: iload_0
       3: iload_1
       4: isub
       5: invokestatic  #15                 // Method java/lang/Integer.valueOf:(I)Ljava/lang/Integer;
       8: areturn
       9: astore_1
      10: new           #40                 // class java/lang/RuntimeException
      13: dup
      14: aload_1
      15: invokespecial #42                 // Method java/lang/RuntimeException."<init>":(Ljava/lang/Throwable;)V
      18: athrow
    Exception table:
       from    to  target type
           0     8     9   Class java/lang/Exception
    LineNumberTable:
      line 11: 0
      line 12: 2
      line 13: 9
      line 14: 10
    LocalVariableTable:
      Start  Length  Slot  Name   Signature
          2       7     1   two    I
         10       9     1     e    Ljava/lang/Exception;
          0      19     0     i    I
    StackMapTable: number_of_entries = 1
      frame_type = 73 /* same_locals_1_stack_item */
        stack = [ class java/lang/Exception ]
```

```
public static java.lang.Integer addTwo(int);
  descriptor: (I)Ljava/lang/Integer;
  flags: (0x0009) ACC_PUBLIC, ACC_STATIC
  Code:
    stack=3, locals=2, args_size=1
       0: iconst_2
       1: istore_1
       2: iload_0
       3: iload_1
       4: isub
       5: invokestatic  #15                 // Method java/lang/Integer.valueOf:(I)Ljava/lang/Integer;
       8: areturn
       9: astore_1
      10: new           #40                 // class java/lang/RuntimeException
      13: dup
      14: aload_1
      15: invokespecial #42                 // Method java/lang/RuntimeException."<init>":(Ljava/lang/Throwable;)V
      18: athrow
    Exception table:
       from    to  target type
           0     8     9   Class java/lang/Exception
    LineNumberTable:
      line 11: 0
      line 12: 2
      line 13: 9
      line 14: 10
    LocalVariableTable:
      Start  Length  Slot  Name   Signature
          2       7     1   two    I
         10       9     1     e    Ljava/lang/Exception;
          0      19     0     i    I
    StackMapTable: number_of_entries = 1
      frame_type = 73 /* same_locals_1_stack_item */
        stack = [ class java/lang/Exception ]
```

Chapter 4. The class File Format

Chapter 4. The class File Format

- The value of start_pc must be a valid index into the code array of this Code attribute and must be the index of the opcode of an instruction.

Chapter 4. The class File Format

- The value of start_pc must be a valid index into the code array of this Code attribute and must be the index of the opcode of an instruction.

- The value of start_pc + length must either be a valid index into the code array of this Code attribute and be the index of the opcode of an instruction, or it must be the first index beyond the end of that code array.

```
public static java.lang.Integer addTwo(int);
  descriptor: (I)Ljava/lang/Integer;
  flags: (0x0009) ACC_PUBLIC, ACC_STATIC
  Code:
    stack=3, locals=2, args_size=1
       0: iconst_2
       1: istore_1
       2: iload_0
       3: iload_1
       4: isub
       5: invokestatic  #15                 // Method java/lang/Integer.valueOf:(I)Ljava/lang/Integer;
       8: areturn
       9: astore_1
      10: new           #40                 // class java/lang/RuntimeException
      13: dup
      14: aload_1
      15: invokespecial #42                 // Method java/lang/RuntimeException."<init>":(Ljava/lang/Throwable;)V
      18: athrow
    Exception table:
       from    to  target type
           0     8     9   Class java/lang/Exception
    LineNumberTable:
      line 11: 0
      line 12: 2
      line 13: 9
      line 14: 10
    LocalVariableTable:
      Start  Length  Slot  Name   Signature
          2       7     1   two    I
         10       9     1      e   Ljava/lang/Exception;
          0      19     0      i   I
    StackMapTable: number_of_entries = 1
      frame_type = 73 /* same_locals_1_stack_item */
        stack = [ class java/lang/Exception ]
```

```
public static java.lang.Integer addTwo(int);
  descriptor: (I)Ljava/lang/Integer;
  flags: (0x0009) ACC_PUBLIC, ACC_STATIC
  Code:
    stack=3, locals=2, args_size=1
       0: iconst_2
       1: istore_1
       2: iload_0
       3: iload_1
       4: isub
       5: invokestatic  #15                 // Method java/lang/Integer.valueOf:(I)Ljava/lang/Integer;
       8: areturn
       9: astore_1
      10: new           #40                 // class java/lang/RuntimeException
      13: dup
      14: aload_1
      15: invokespecial #42                 // Method java/lang/RuntimeException."<init>":(Ljava/lang/Throwable;)V
      18: athrow
    Exception table:
       from    to  target type
           0     8     9   Class java/lang/Exception
    LineNumberTable:
      line 11: 0
      line 12: 2
      line 13: 9
      line 14: 10
    LocalVariableTable:
      Start  Length  Slot  Name   Signature
          2       7     1   two    I
         10       9     1      e   Ljava/lang/Exception;
          0      19     0      i   I
    StackMapTable: number_of_entries = 1
      frame_type = 73 /* same_locals_1_stack_item */
        stack = [ class java/lang/Exception ]
```

```
public static java.lang.Integer addTwo(int);
  descriptor: (I)Ljava/lang/Integer;
  flags: (0x0009) ACC_PUBLIC, ACC_STATIC
  Code:
    stack=3, locals=2, args_size=1
       0: iconst_2
       1: istore_1
       2: iload_0
       3: iload_1
       4: isub
       5: invokestatic  #15                 // Method java/lang/Integer.valueOf:(I)Ljava/lang/Integer;
       8: areturn
       9: astore_1
      10: new           #40                 // class java/lang/RuntimeException
      13: dup
      14: aload_1
      15: invokespecial #42                 // Method java/lang/RuntimeException."<init>":(Ljava/lang/Throwable;)V
      18: athrow
    Exception table:
       from    to  target type
           0     8     9   Class java/lang/Exception
    LineNumberTable:
      line 11: 0
      line 12: 2
      line 13: 9
      line 14: 10
    LocalVariableTable:
      Start  Length  Slot  Name   Signature
          2       7     1   two    I
         10       9     1     e    Ljava/lang/Exception;
          0      19     0     i    I
    StackMapTable: number_of_entries = 1
      frame_type = 73 /* same_locals_1_stack_item */
        stack = [ class java/lang/Exception ]
```

```
public static java.lang.Integer addTwo(int);
  descriptor: (I)Ljava/lang/Integer;
  flags: (0x0009) ACC_PUBLIC, ACC_STATIC
  Code:
    stack=3, locals=2, args_size=1
       0: iconst_2
       1: istore_1
       2: iload_0
       3: iload_1
       4: isub
       5: invokestatic  #15                 // Method java/lang/Integer.valueOf:(I)Ljava/lang/Integer;
       8: areturn
       9: astore_1
      10: new           #40                 // class java/lang/RuntimeException
      13: dup
      14: aload_1
      15: invokespecial #42                 // Method java/lang/RuntimeException."<init>":(Ljava/lang/Throwable;)V
      18: athrow
    Exception table:
       from    to  target type
           0     8     9   Class java/lang/Exception
    LineNumberTable:
      line 11: 0
      line 12: 2
      line 13: 9
      line 14: 10
    LocalVariableTable:
      Start  Length  Slot  Name   Signature
          2       7     1   two    I
         10       9     1      e   Ljava/lang/Exception;
          0      19     0      i   I
    StackMapTable: number_of_entries = 1
      frame_type = 73 /* same_locals_1_stack_item */
        stack = [ class java/lang/Exception ]
```

```
public static java.lang.Integer addTwo(int);
  descriptor: (I)Ljava/lang/Integer;
  flags: (0x0009) ACC_PUBLIC, ACC_STATIC
  Code:
    stack=3, locals=2, args_size=1
       0: iconst_2
       1: istore_1
       2: iload_0
       3: iload_1
       4: isub
       5: invokestatic  #15                 // Method java/lang/Integer.valueOf:(I)Ljava/lang/Integer;
       8: areturn
       9: astore_1
      10: new           #40                 // class java/lang/RuntimeException
      13: dup
      14: aload_1
      15: invokespecial #42                 // Method java/lang/RuntimeException."<init>":(Ljava/lang/Throwable;)V
      18: athrow
    Exception table:
       from    to  target type
           0     8     9   Class java/lang/Exception
    LineNumberTable:
      line 11: 0
      line 12: 2
      line 13: 9
      line 14: 10
    LocalVariableTable:
      Start  Length  Slot  Name   Signature
          2       7     1   two    I
         10       9     1     e    Ljava/lang/Exception;
          0      19     0     i    I
    StackMapTable: number_of_entries = 1
      frame_type = 73 /* same_locals_1_stack_item */
        stack = [ class java/lang/Exception ]
```

```
public static java.lang.Integer addTwo(int);
  descriptor: (I)Ljava/lang/Integer;
  flags: (0x0009) ACC_PUBLIC, ACC_STATIC
  Code:
    stack=3, locals=2, args_size=1
       0: iconst_2
       1: istore_1
       2: iload_0
       3: iload_1
       4: isub
       5: invokestatic  #15                 // Method java/lang/Integer.valueOf:(I)Ljava/lang/Integer;
       8: areturn
       9: astore_1
      10: new           #40                 // class java/lang/RuntimeException
      13: dup
      14: aload_1
      15: invokespecial #42                 // Method java/lang/RuntimeException."<init>":(Ljava/lang/Throwable;)V
      18: athrow
    Exception table:
       from    to  target type
           0     8     9   Class java/lang/Exception
    LineNumberTable:
      line 11: 0
      line 12: 2
      line 13: 9
      line 14: 10
    LocalVariableTable:
      Start  Length  Slot  Name   Signature
          2       7     1   two    I
         10       9     1     e    Ljava/lang/Exception;
          0      19     0     i    I
    StackMapTable: number_of_entries = 1
      frame_type = 73 /* same_locals_1_stack_item */
        stack = [ class java/lang/Exception ]
```

Truths are Illusions

-Friedrich Nietzsche

```
public static java.lang.Integer addTwo(int);
  descriptor: (I)Ljava/lang/Integer;
  flags: (0x0009) ACC_PUBLIC, ACC_STATIC
  Code:
    stack=3, locals=2, args_size=1
       0: iconst_2
       1: istore_1
       2: iload_0
       3: iload_1
       4: isub
       5: invokestatic  #15                 // Method java/lang/Integer.valueOf:(I)Ljava/lang/Integer;
       8: areturn
       9: astore_1
      10: new           #40                 // class java/lang/RuntimeException
      13: dup
      14: aload_1
      15: invokespecial #42                 // Method java/lang/RuntimeException."<init>":(Ljava/lang/Throwable;)V
      18: athrow
    Exception table:
       from    to  target type
           0     8     9   Class java/lang/Exception
    LineNumberTable:
      line 11: 0
      line 12: 2
      line 13: 9
      line 14: 10
    LocalVariableTable:
      Start  Length  Slot  Name   Signature
          2       7     1   two    I
         10       9     1     e    Ljava/lang/Exception;
          0      19     0     i    I
    StackMapTable: number_of_entries = 1
      frame_type = 73 /* same_locals_1_stack_item */
        stack = [ class java/lang/Exception ]
```

Java agents

Java agents

- java.lang.instrument

Java agents

- java.lang.instrument

- Provides services that allow Java programming language agents to instrument programs running on the JVM.

```
public static java.lang.Integer addTwo(int);
  descriptor: (I)Ljava/lang/Integer;
  flags: (0x0009) ACC_PUBLIC, ACC_STATIC
  Code:
    stack=3, locals=2, args_size=1
       0: iconst_2
       1: istore_1
       2: iload_0
       3: iload_1
       4: iadd
       5: invokestatic  #15
       8: areturn
       9: astore_1
      10: new           #40
      13: dup
      14: aload_1
      15: invokespecial #42
      18: athrow
    Exception table:
       from    to  target type
           0     8     9   Class java/lang/Exception
    LineNumberTable:
      line 11: 0
      line 12: 2
      line 13: 9
      line 14: 10
    LocalVariableTable:
      Start  Length  Slot  Name   Signature
          2       7     1   two    I
         10       9     1     e    Ljava/lang/Exception;
          0      19     0     i    I
    StackMapTable: number_of_entries = 1
      frame_type = 73 /* same_locals_1_stack_item */
        stack = [ class java/lang/Exception ]
```

```
public static java.lang.Integer addTwo(int);
  descriptor: (I)Ljava/lang/Integer;
  flags: (0x0009) ACC_PUBLIC, ACC_STATIC
  Code:
    stack=3, locals=2, args_size=1
       0: iconst_2
       1: istore_1
       2: iload_0
       3: iload_1
       4: iadd
       5: invokestatic  #15
       8: areturn
       9: astore_1
      10: new           #40
      13: dup
      14: aload_1
      15: invokespecial #42
      18: athrow
    Exception table:
       from    to  target type
           0     8     9   Class java/lang/Exception
    LineNumberTable:
      line 11: 0
      line 12: 2
      line 13: 9
      line 14: 10
    LocalVariableTable:
      Start  Length  Slot  Name   Signature
          0       9     1   two    I
          0      19     1     e    Ljava/lang/Exception;
          0      19     0     i    I
    StackMapTable: number_of_entries = 1
      frame_type = 73 /* same_locals_1_stack_item */
        stack = [ class java/lang/Exception ]
```

```
public static java.lang.Integer addTwo(int);
  descriptor: (I)Ljava/lang/Integer;
  flags: (0x0009) ACC_PUBLIC, ACC_STATIC
  Code:
    stack=3, locals=2, args_size=1
       0: iconst_2
       1: istore_1
       2: iload_0
       3: iload_1
       4: iadd
       5: invokestatic  #15
       8: areturn
       9: astore_1
      10: new           #40
      13: dup
      14: aload_1
      15: invokespecial #42
      18: athrow
    Exception table:
       from    to  target type
           0     8     9   Class java/lang/Exception
    LineNumberTable:
      line 11: 0
      line 12: 2
      line 13: 9
      line 14: 10
    LocalVariableTable:
      Start  Length  Slot  Name   Signature
          2       7     1   two    I
         10       9     1     e    Ljava/lang/Exception;
          0      19     0     i    I
    StackMapTable: number_of_entries = 1
      frame_type = 73 /* same_locals_1_stack_item */
        stack = [ class java/lang/Exception ]
```

```
public static java.lang.Integer addTwo(int);
  descriptor: (I)Ljava/lang/Integer;
  flags: (0x0009) ACC_PUBLIC, ACC_STATIC
  Code:
    stack=3, locals=2, args_size=1
       0: iconst_2
       1: istore_1
       2: iload_0
       3: iload_1
       4: iadd
       5: invokestatic  #15
       8: areturn
       9: astore_1
      10: new           #40
      13: dup
      14: aload_1
      15: invokespecial #42
      18: athrow
    Exception table:
       from    to  target type
           0     8     9   Class java/lang/Exception
    LineNumberTable:
      line 11: 0
      line 12: 2
      line 13: 9
      line 14: 10
    LocalVariableTable:
      Start  Length  Slot  Name   Signature
          0       9     1   two    I
          0      19     1     e    Ljava/lang/Exception;
          0      19     0     i    I
    StackMapTable: number_of_entries = 1
      frame_type = 73 /* same_locals_1_stack_item */
        stack = [ class java/lang/Exception ]
```