

A photograph of a cemetery at sunset. The ground is covered in fallen brown leaves. Numerous tombstones of various shapes and sizes are scattered across the scene. In the background, a line of trees silhouettes against a bright, glowing sunset sky. The overall mood is somber and reflective.

Dead Classes

Juli 2025, Oli B.

Software rots

The Problem

Framework



Problems with Dead Classes



Problems with Dead Classes

waste



Problems with Dead Classes

waste

maintainability



Problems with Dead Classes

A large, spherical object made of dried, cracked mud sits on a dark, reflective surface. The sphere is covered in a network of deep, irregular cracks, giving it a textured, weathered appearance. The surface it sits on is dark and glossy, creating a clear reflection of the sphere below it. The background is a dark, gradient wash.

waste

maintainability

dependencies

Problems with Dead Classes



waste

refactoring costs

maintainability

dependencies

Problems with Dead Classes

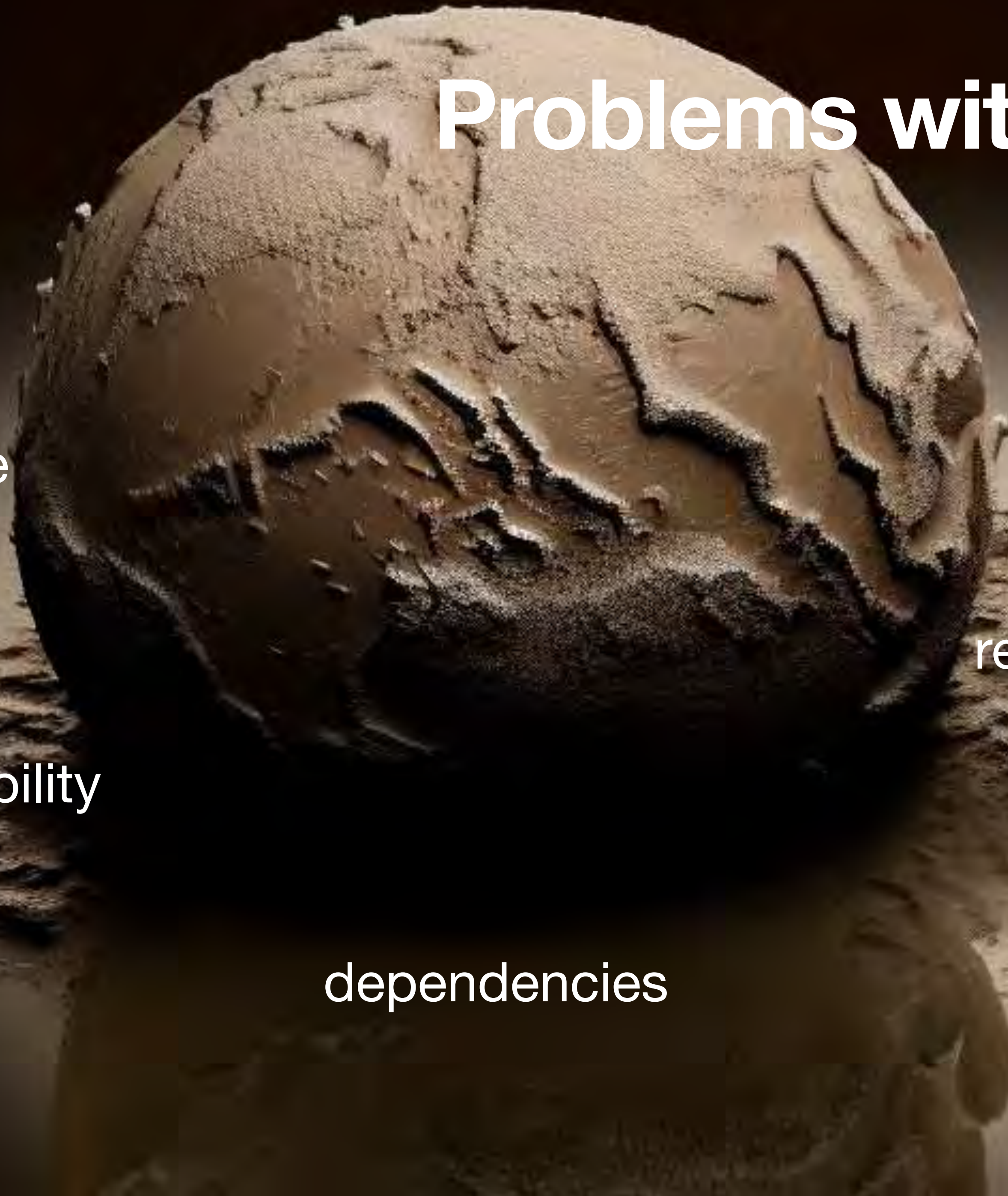
waste

technical debt

refactoring costs

maintainability

dependencies



Problems with Dead Classes



waste

technical debt

refactoring costs

maintainability

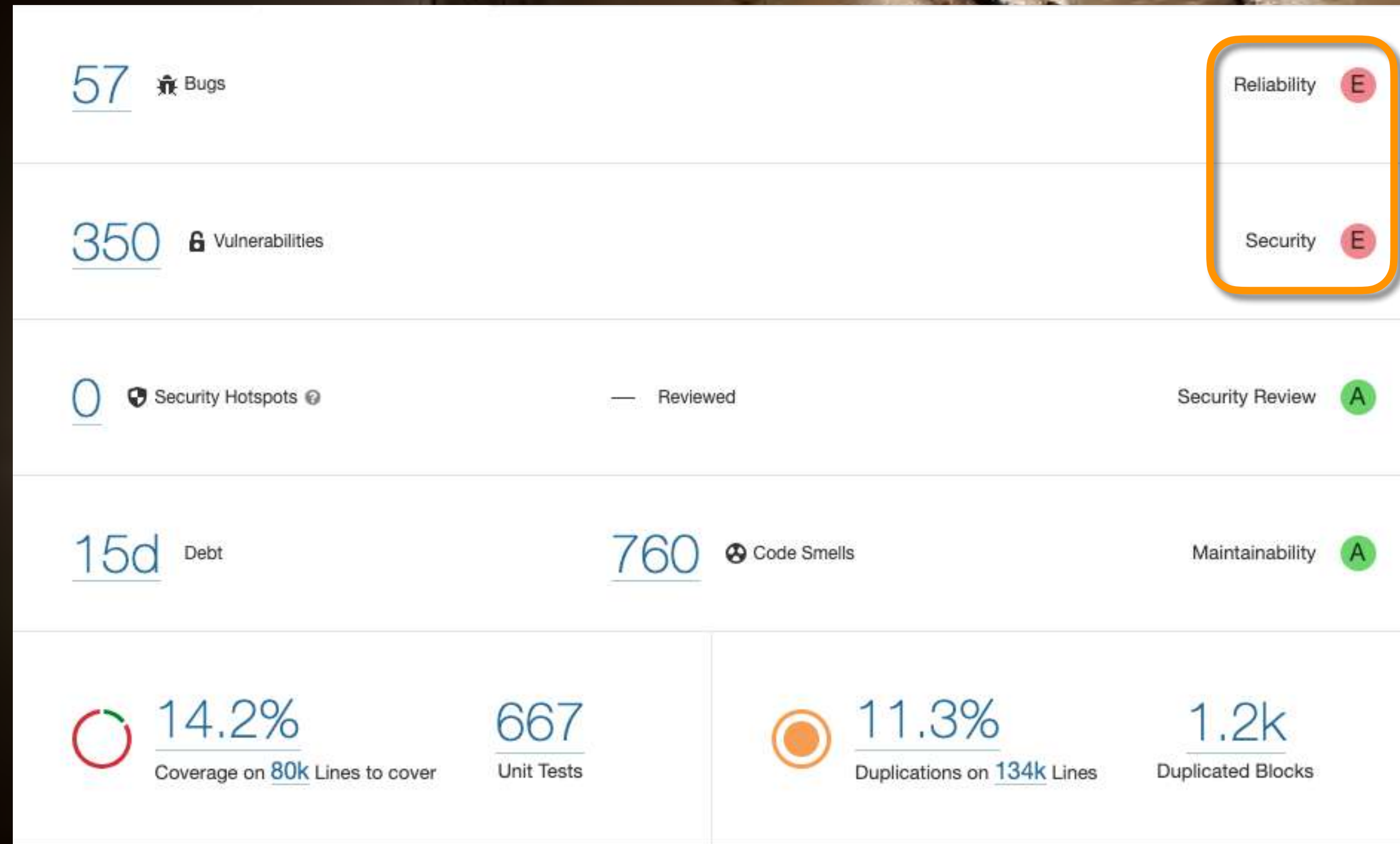
security

dependencies

Problems with Dead Classes

- SQL-Injection
- XML-Parsing
- Ressourcen-Leaks

security



Main Problem



How to Find Dead Classes

$$dc = ac - \sum_{t=0}^{\infty} lc$$

dc: dead classes

ac: all classes

lc: loaded classes

How to Get All Classes?

$$dc = ac - \sum_{t=0}^{\infty} lc$$

Get Classpath

```
String classpath = System.getProperty("java.class.path");
```

The screenshot shows the 'Java-Monitoring- und Managementkonsole' window. The title bar indicates 'pid: 38967 clazfish.sample.Bank'. The 'VM-Übersicht' tab is selected, displaying system and memory statistics. Below these, the 'VM-Argumente' and 'Classpath' are listed. The 'Classpath' is highlighted with a red box, showing a long list of JAR files and directories. The 'Library-Pfad' and 'Boot Classpath' are also visible at the bottom.

Betriebssystem: Mac OS X 15.1
Architektur: x86_64
Anzahl Prozessoren: 8
Festgeschriebener virtueller Speicher: 45.508.244 KB

Physischer Gesamtspeicher: 33.554.432 KB
Freier physischer Speicher: 2.796.456 KB
Gesamter Swap-Bereich: 2.097.152 KB
Freier Swap-Bereich: 1.759.232 KB

VM-Argumente: -javaagent:/Users/o Boehm/.m2/repository/de/aosd/clazfish/clazfish-agent/2.3-SNAPSHOT/clazfish-agent-2.3-SNAPSHOT.jar -javaagent:/Applications/opt/java/IntelliJ IDEA.app/Contents/lib/idea_rt.jar=57220:/Applications/opt/java/IntelliJ IDEA.app/Contents/bin -Dfile.encoding=ISO-8859-15

Classpath: /Users/o Boehm/work/github/en/ClazzFish/sample/target/classes:/Users/o Boehm/work/github/en/ClazzFish/jdbc/target/classes:/Users/o Boehm/work/github/en/ClazzFish/monitor/target/classes:/Users/o Boehm/.m2/repository/org/apache/commons/commons-lang3/3.12.0/commons-lang3-3.12.0.jar:/Users/o Boehm/.m2/repository/commons-io/commons-io/2.14.0/commons-io-2.14.0.jar:/Users/o Boehm/.m2/repository/io/github/classgraph/classgraph/4.8.154/classgraph-4.8.154.jar:/Users/o Boehm/.m2/repository/io/github/toolfactory/jvm-driver/9.4.2/jvm-driver-9.4.2.jar:/Users/o Boehm/.m2/repository/io/github/toolfactory/narcissus/1.0.7/narcissus-1.0.7.jar:/Users/o Boehm/.m2/repository/org/hsqldb/hsqldb/2.7.4/hsqldb-2.7.4.jar:/Users/o Boehm/.m2/repository/org/apache/logging/log4j/log4j-slf4j2-impl/2.22.1/log4j-slf4j2-impl-2.22.1.jar:/Users/o Boehm/.m2/repository/org/apache/logging/log4j/log4j-api/2.22.1/log4j-api-2.22.1.jar:/Users/o Boehm/.m2/repository/org/slf4j/slf4j-api/2.0.11/slf4j-api-2.0.11.jar:/Users/o Boehm/.m2/repository/org/apache/logging/log4j/log4j-core/2.22.1/log4j-core-2.22.1.jar

Library-Pfad: /Users/o Boehm/Library/Java/Extensions:/Library/Java/Extensions:/Network/Library/Java/Extensions:/System/Library/Java/Extensions:/usr/lib/java:.

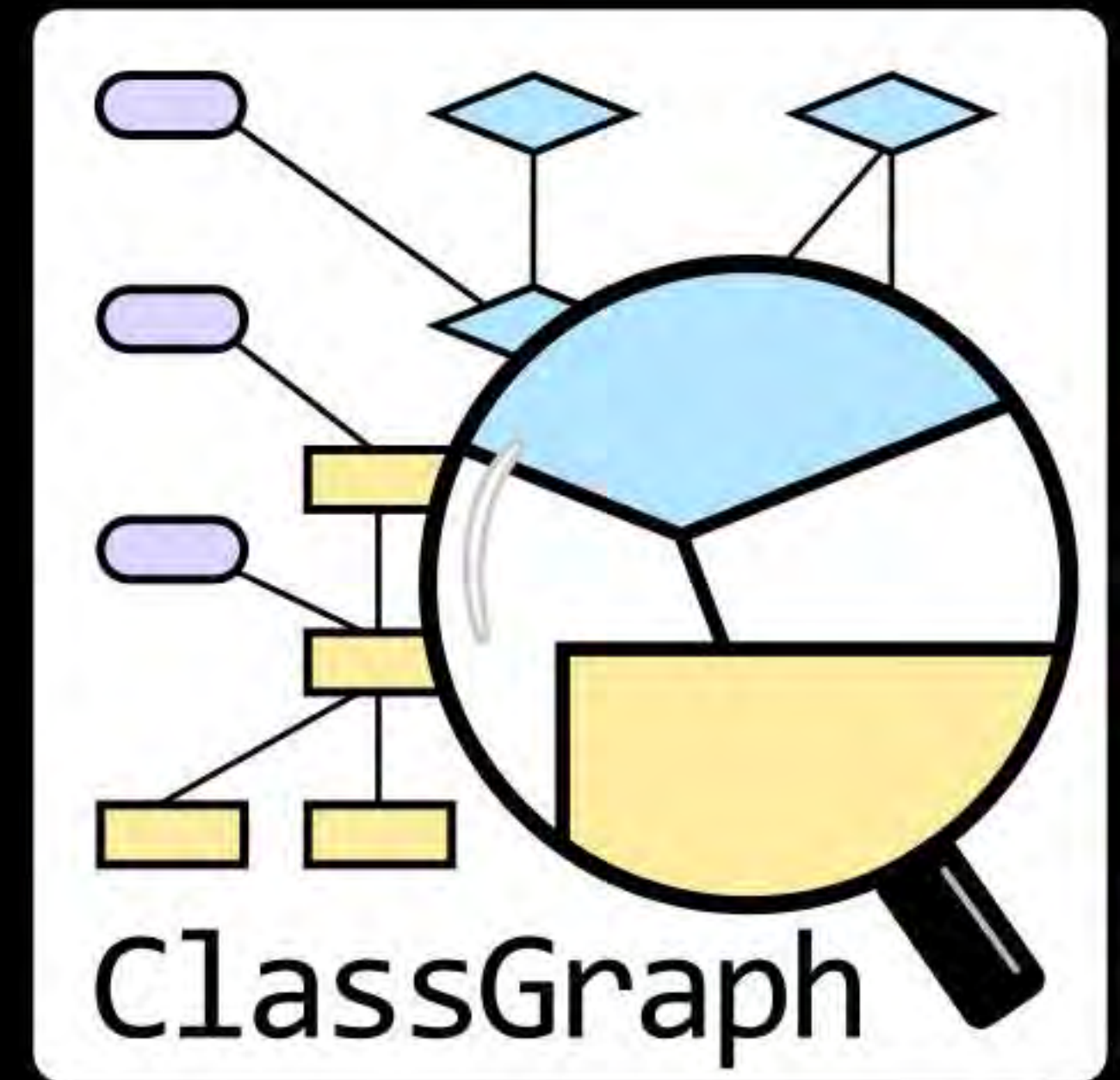
Boot Classpath: Nicht verfügbar

Get All Classes

- file:/...
 - find .class files in directory
- jar:file:/...
 - simple JAR: find .class files in JAR (ZIP)
 - nested JARs (JAR, WAR, EAR):
 - find simple JARs, inspect it

Libs for Classpathes

- <https://github.com/classgraph>
- `com.google.common.reflect.ClassPath` (Guava)



How to Get Loaded Classes?

ClassLoader Internals

```
▼ this.classLoader = {ClassLoaders$AppClassLoader@4142}
  > Ⓡ assertionLock: Object = {Object@4166}
    Ⓡ classAssertionStatus: Map = null
  ▼ Ⓡ classes: ArrayList = {ArrayList@4162} size = 1005
    > 0 = {@799} "interface clazzfish.agent.ClasspathAgentMBean" ... Navigate
    > 1 = {@800} "class clazzfish.agent.ClasspathAgent" ... Navigate
    > 2 = {@1044} "class clazzfish.agent.logging.SingleLineFormatter" ... Navigate
    > 3 = {@1179} "class clazzfish.agent.logging.StdoutHandler" ... Navigate
    > 4 = {@1633} "class com.intellij.rt.junit.JUnitStarter" ... Navigate
    > 5 = {@1634} "class com.intellij.rt.execution.junit.RepeatCount" ... Navigate
    > 6 = {@1635} "interface com.intellij.rt.junit.IdeaTestRunner" ... Navigate
    > 7 = {@1637} "class com.intellij.junit5.JUnit5IdeaTestRunner" ... Navigate
    > 8 = {@1640} "interface org.junit.platform.engine.TestDescriptor" ... Navigate
```

Ask the Classloader

```
/**
 * Returns the class with the given binary name if this
 * loader has been recorded by the Java virtual machine as an initiating
 * loader of a class with that binary name. Otherwise
 * {@code null} is returned.
 *
 * @param name
 *         The binary name of the class
 *
 * @return The {@code Class} object, or {@code null} if the class has
 *         not been loaded
 *
 * @since 1.1
 */
protected final Class<?> findLoadedClass(String name) {
    if (!checkName(name))
        return null;
    return findLoadedClass0(name);
}

private final native Class<?> findLoadedClass0(String name);
```

Ask the Classloader

SystemClassLoader

WebClassLoader

Which Classloader?

ContextClassLoader

AppClassLoader

Ask the GC

jmap -histo:all <pid>

```
oboehm@PLTCM7002$ jmap -histo:all 9668
```

num	#instances	#bytes	class name (module)
1:	182752	15764168	[B (java.base@17.0.6)
2:	17269	8513224	[I (java.base@17.0.6)
3:	135138	3243312	java.lang.String (java.base@17.0.6)
4:	14721	1177680	java.net.URI (java.base@17.0.6)
5:	32078	1026496	java.util.concurrent.ConcurrentHashMap\$Node (java.base@17.0.6)
6:	21455	858200	java.util.TreeMap\$Entry (java.base@17.0.6)
...			
1372:	1	16	sun.util.resources.LocaleData (java.base@17.0.6)
1373:	1	16	sun.util.resources.LocaleData\$LocaleDataStrategy (java.base@17.0.6)
1374:	1	16	sun.util.resources.cldr.provider.CLDRLocaleDataMetaInfo (jdk.localedata@17.0.6)
Total	627763	42881928	

```
oboehm@PLTCM7002$
```

Ask DiagnosticCommand

The screenshot displays the Java Monitoring & Management Console window. The title bar reads "Java Monitoring & Management Console". Below the title bar is a menu bar with "Connection", "Window", and "Help". The main content area is divided into two panes. The left pane shows a tree view of the MBean hierarchy, with "DiagnosticCommand" selected under "com.sun.management". The right pane shows the "Operation invocation" section, which includes a text input field containing "gcClassHistogram" and a "String[]" dropdown. Below this is the "MBeanOperationInfo" section, which contains a table with the following data:

Name	Value
Operation:	
Name	gcClassHistogram
Description	Provide statistics about the Java heap usage.
Impact	ACTION_INFO
ReturnType	java.lang.String
Parameter-0:	
Name	arguments
Description	Array of Diagnostic Commands Arguments and Options
Type	[Ljava.lang.String;

Below the "MBeanOperationInfo" section is the "Descriptor" section, which contains a table with the following data:

Name	Value
Operation:	
dcmd.arguments	{-all={dcmd.arg.description=Inspect all objects, including unreach...
dcmd.description	Provide statistics about the Java heap usage.
dcmd.enabled	true
dcmd.help	GC.class_histogram Provide statistics about the Java heap usage. I...
dcmd.name	GC.class_histogram
dcmd.permissionAction	
dcmd.permissionClass	java.lang.management.ManagementPermission
dcmd.permissionName	monitor
dcmd.vmlImpact	High: Depends on Java heap size and content.

Ask DiagnosticCommand II

```
public static List<Class<?>> getLoadedClassesFromGC() {
    String mbeanName = "com.sun.management:type=DiagnosticCommand";
    try {
        Object classHistogram = ManagementFactory.getPlatformMBeanServer().invoke(
            new ObjectName(mbeanName),
            "gcClassHistogram",
            new Object[]{new String[]{"-all"}},
            new String[]{"[Ljava.lang.String;"});
        return parseClassHistogramm(classHistogram.toString());
    } catch (JMException ex) {
        log.warn("Cannot call 'getLoadedClasses(..)' from MBean \"{}}\"", mbeanName, ex);
        return new ArrayList<>();
    }
}
```



Use JVM Log

java -Xlog:class+load=info ...

```
[0.021s][info][class,load] java.lang.Object source: shared objects file
[0.021s][info][class,load] java.io.Serializable source: shared objects file
[0.022s][info][class,load] java.lang.Comparable source: shared objects file
[0.022s][info][class,load] java.lang.CharSequence source: shared objects file
[0.022s][info][class,load] java.lang.constant.Constable source: shared objects file
[0.022s][info][class,load] java.lang.constant.ConstantDesc source: shared objects file
[0.022s][info][class,load] java.lang.String source: shared objects file
...
```

Ask DiagnosticCommand III

The screenshot displays the Java Monitoring & Management Console for the process `pid: 57349 com.intellij.idea.Main`. The `MBeans` tab is selected, showing a tree of MBeans on the left and details for the `vmClassHierarchy` operation on the right.

Left Panel (MBeans Tree):

- `com.sun.management`
 - D diagnosticCommand**
 - Operations
 - `compilerCodeHeapAn`
 - `compilerCodecache`
 - `compilerCodelist`
 - `compilerDirectivesAdc`
 - `compilerDirectivesCle`
 - `compilerDirectivesPrin`
 - `compilerDirectivesRen`
 - `compilerQueue`
 - `gcClassHistogram`
 - `gcFinalizerInfo`
 - `gcHeapInfo`
 - `gcRun`
 - `gcRunFinalization`
 - `help`
 - `jfrCheck`
 - `jfrConfigure`
 - `jfrDump`
 - `jfrStart`
 - `jfrStop`
 - `jfrView`
 - `jvmtiAgentLoad`
 - `jvmtiDataDump`
 - `threadDumpToFile`
 - `threadPrint`
 - `vmCds`
 - vmClassHierarchy**

Right Panel (Operation invocation):

Operation invocation: `java.lang.String` `vmClassHierarchy` (arguments: `String[]`)

MBeanOperationInfo:

Name	Value
Operation:	
Name	<code>vmClassHierarchy</code>
Description	Print a list of all loaded classes, indented to show the class hierarchy. The na...
Impact	<code>ACTION_INFO</code>
ReturnType	<code>java.lang.String</code>
Parameter-0:	
Name	<code>arguments</code>
Description	Array of Diagnostic Commands Arguments and Options
Type	<code>[Ljava.lang.String;</code>

Descriptor:

Name	Value
Operation:	
<code>dcmd.arguments</code>	<code>{-i={dcmd.arg.description=Inherited interfaces should be printed., dcmd.ar...</code>
<code>dcmd.description</code>	Print a list of all loaded classes, indented to show the class hierarchy. The na...
<code>dcmd.enabled</code>	<code>true</code>
<code>dcmd.help</code>	<code>VM.class_hierarchy</code> Print a list of all loaded classes, indented to show the clas...
<code>dcmd.name</code>	<code>VM.class_hierarchy</code>
<code>dcmd.permissionAction</code>	
<code>dcmd.permissionClass</code>	<code>java.lang.management.ManagementPermission</code>
<code>dcmd.permissionName</code>	<code>monitor</code>
<code>dcmd.vmlImpact</code>	Medium: Depends on number of loaded classes.

Ask DiagnosticCommand IV

```
private static final String DIAGNOSTIC_COMMAND = "com.sun.management:type=DiagnosticCommand";

public static Set<Class<?>> getLoadedClassesFromVmClassHierarchy() {
    try {
        Object classHierarchy = ManagementFactory.getPlatformMBeanServer().invoke(
            new ObjectName(DIAGNOSTIC_COMMAND),
            "vmClassHierarchy",
            new Object[]{new String[]{" "}},
            new String[]{"[Ljava.lang.String;"});
        return parseClassHierarchy(classHierarchy.toString());
    } catch (JMException ex) {
        log.warn("Cannot call 'vmClassHierarchy(..)' from MBean \"{ }\"", DIAGNOSTIC_COMMAND, ex);
        return new HashSet<>();
    }
}
```

The Official Way...

```
package java.lang.instrument;
```

```
public interface Instrumentation {
```

```
...
```

```
/**
```

```
 * Returns an array of all classes currently loaded by the JVM.
```

```
 * The returned array includes all classes and interfaces, including
```

```
 * {@linkplain Class#isHidden hidden classes or interfaces}, and array classes
```

```
 * of all types.
```

```
 *
```

```
 * @return an array containing all the classes loaded by the JVM, zero-length if there are none
```

```
 */
```

```
@SuppressWarnings("rawtypes")
```

```
Class[]
```

```
getAllLoadedClasses();
```

```
...
```

```
}
```

Hooks

agentmain, premain

```
public class ClasspathAgent implements ClasspathAgentMBean {
```

```
    private static Instrumentation instrumentation;
```

```
    public static void agentmain(final String agentArgs, final Instrumentation inst){  
        premain(agentArgs, inst);  
    }
```

```
    public static void premain(final String agentArgs, final Instrumentation inst){  
        instrumentation = inst;  
    }
```

```
    public Class<?>[] getLoadedClasses() {  
        return instrumentation.getAllLoadedClasses();  
    }
```

```
    ...
```

```
Manifest-Version: 1.0  
Created-By: Maven Jar Plugin 3.2.0  
Build-Jdk-Spec: 21  
Agent-Class:  
clazzfish.agent.ClasspathAgent  
Premain-Class:  
clazzfish.agent.ClasspathAgent
```

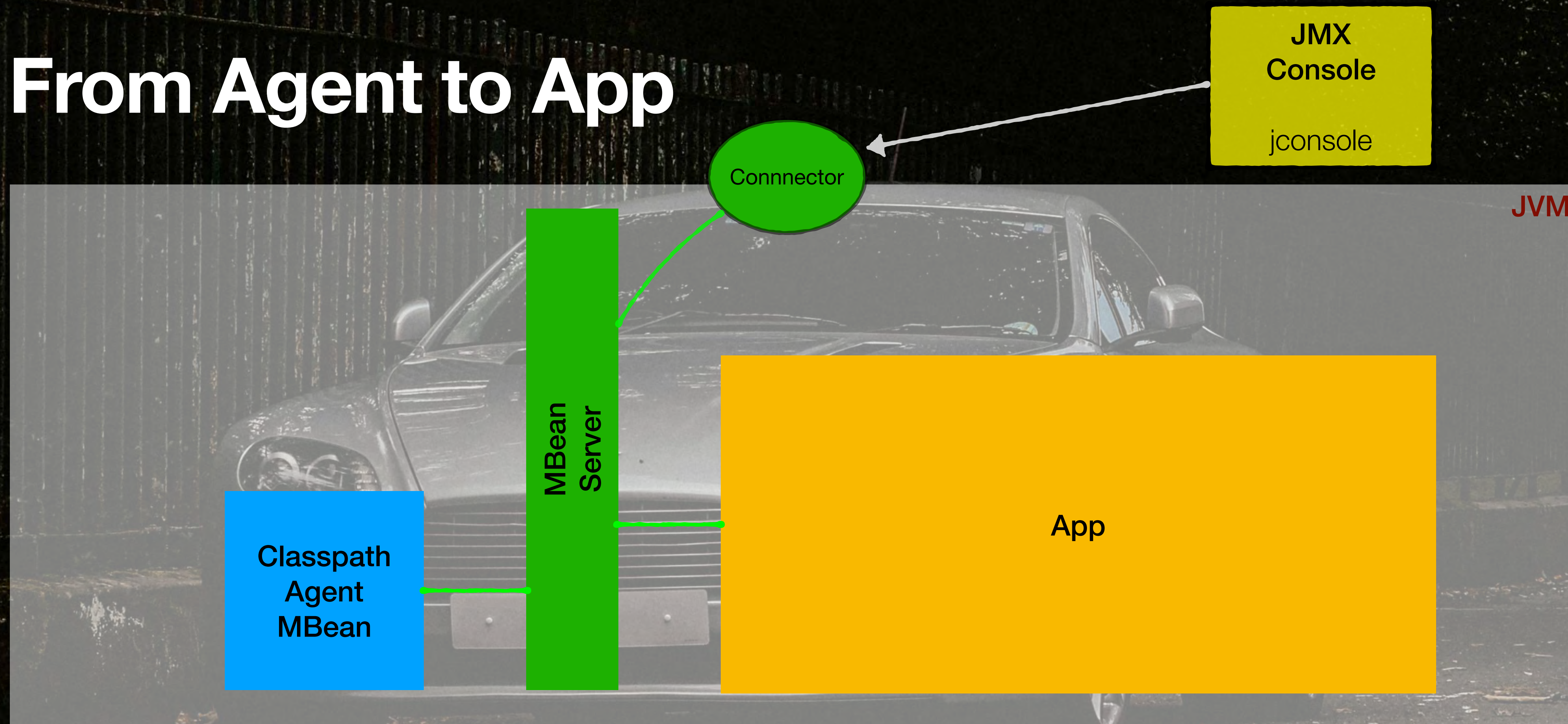
META-INF/MANIFEST.MF

From Agent to App



```
java -javaagent:agent007.jar ...
```

From Agent to App



```
java -javaagent:agent007.jar ...
```

Agent Calling...

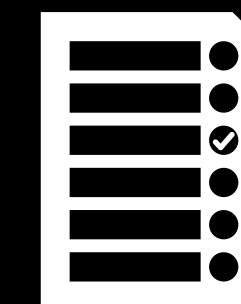
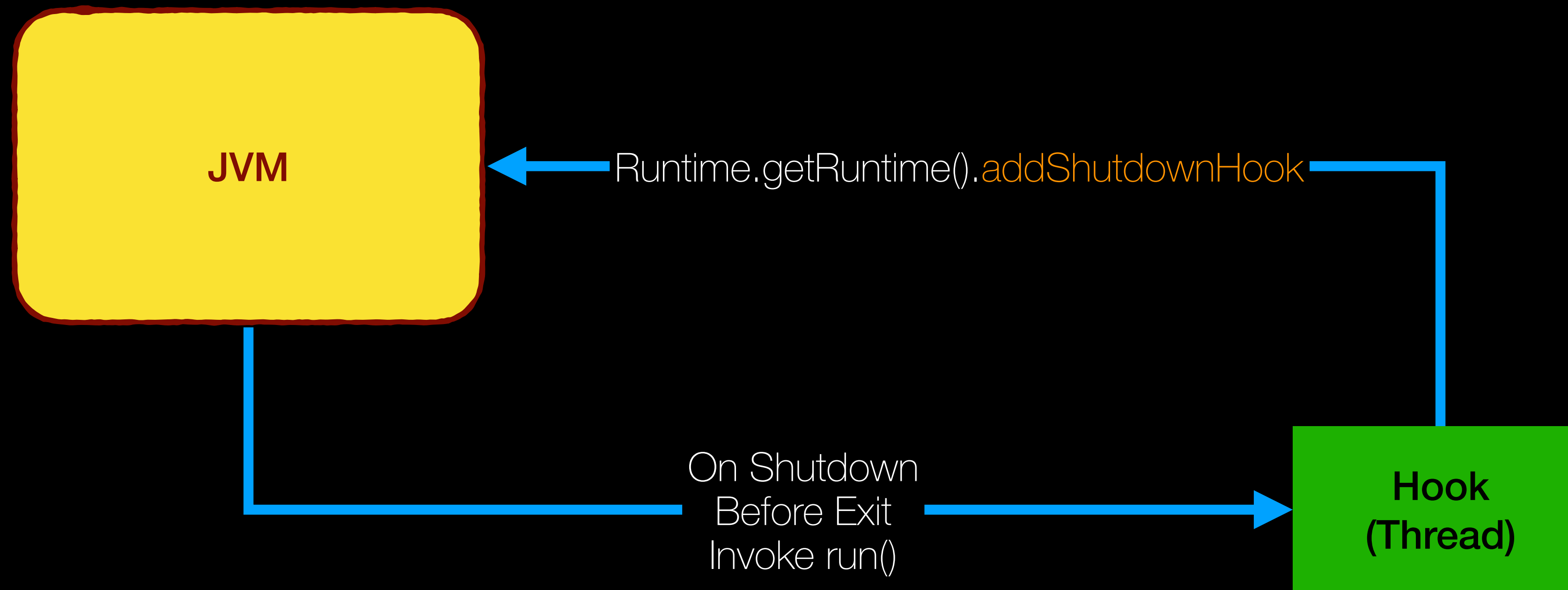
```
public List<Class<?>> getLoadedClassListFromAgent() {  
    Class<?>[] classes = (Class<?>[]) MBEAN_SERVER.invoke(AGENT_MBEAN, "getLoadedClasses",  
        new Object[0], new String[0]);  
    return Arrays.asList(classes);  
}
```

How to Get Loaded Classes?

$$\sum_{t=0}^{\infty} lc$$

How to Get Loaded Classes?

t1

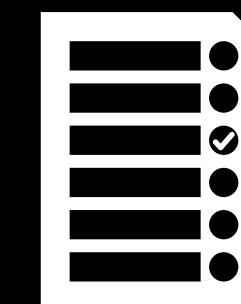
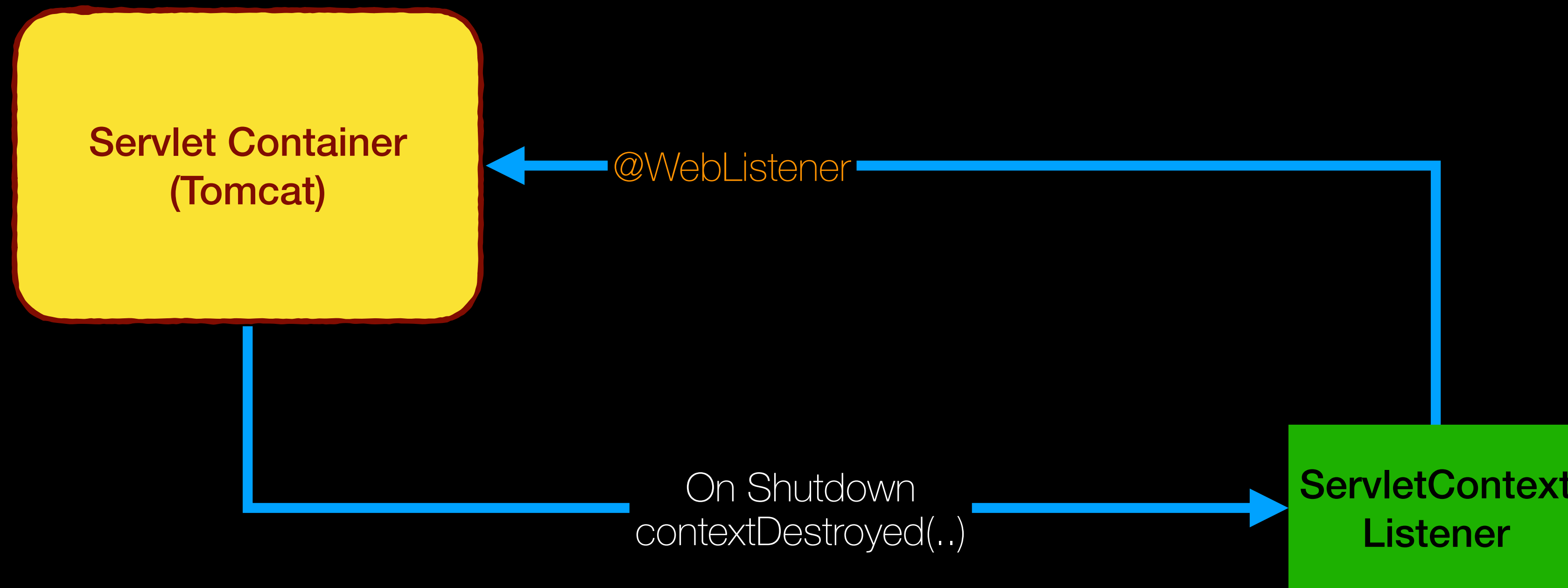


t0

t1

How to Get Loaded Classes?

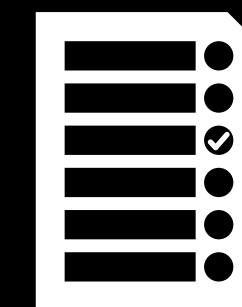
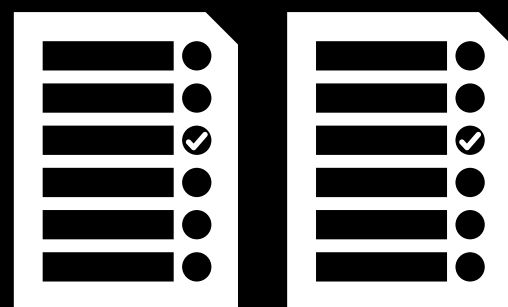
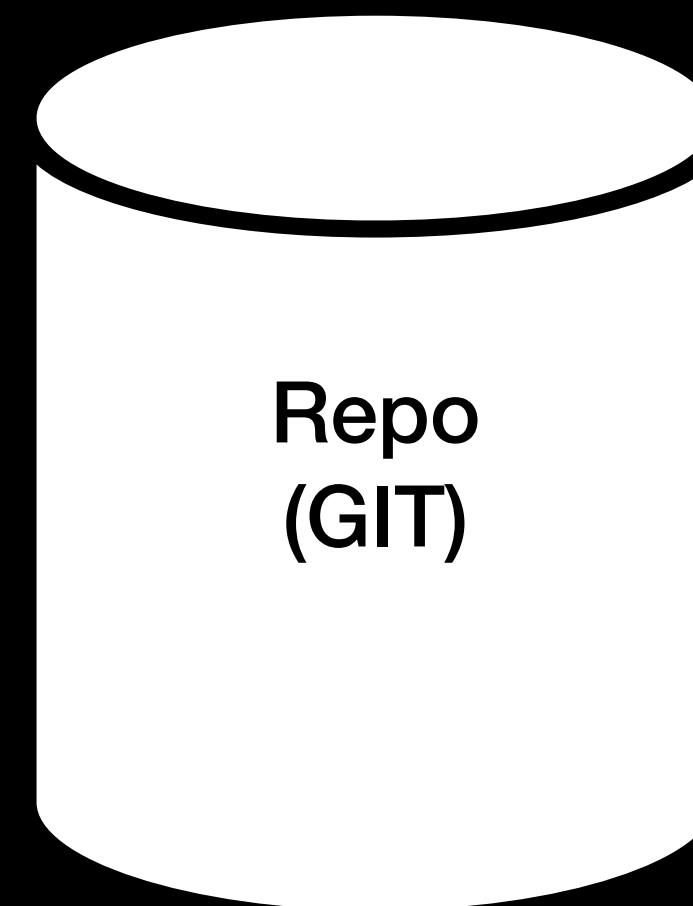
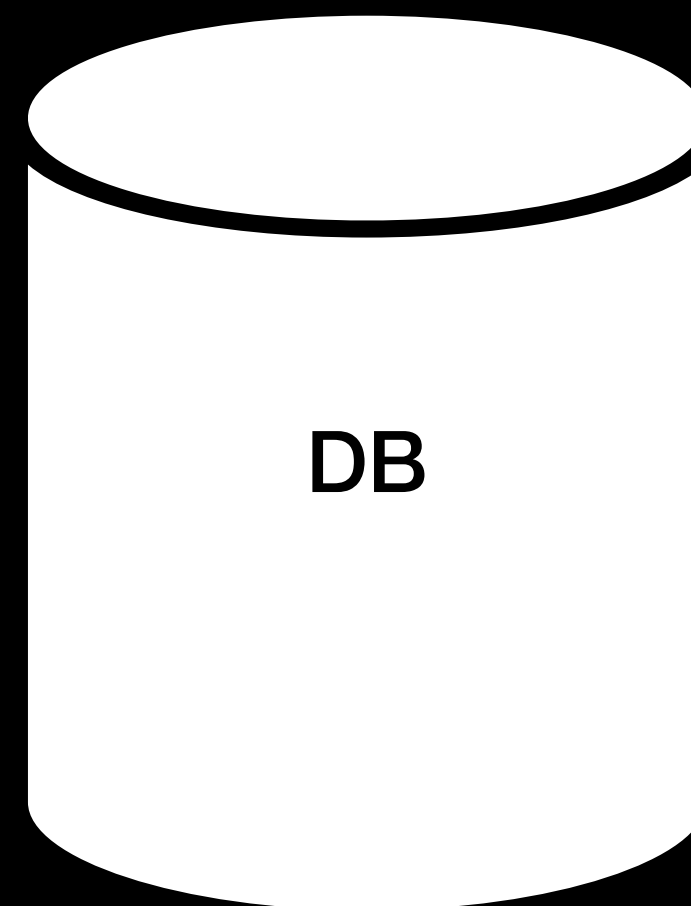
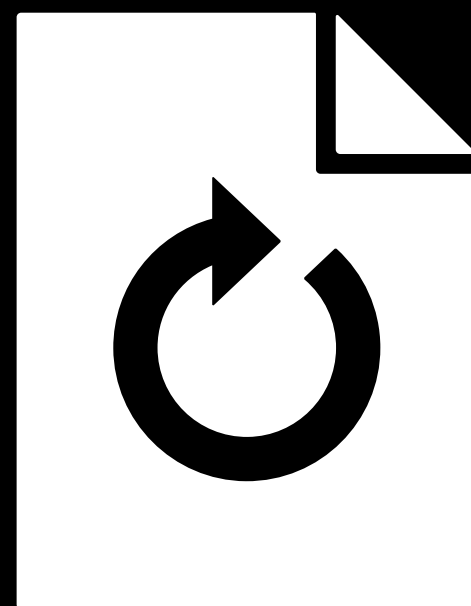
t1



t0

t1

$$\sum_{t=0}^{\infty} lc$$



t0 t1 t2

tn

Kalender 2025

Januar							
KW	Mo	Di	Mi	Do	Fr	Sa	So
1			1	2	3	4	5
2	6	7	8	9	10	11	12
3	13	14	15	16	17	18	19
4	20	21	22	23	24	25	26
5	27	28	29	30	31		

Februar							
KW	Mo	Di	Mi	Do	Fr	Sa	So
5						1	2
6	3	4	5	6	7	8	9
7	10	11	12	13	14	15	16
8	17	18	19	20	21	22	23
9	24	25	26	27	28		

März							
KW	Mo	Di	Mi	Do	Fr	Sa	So
9						1	2
10	3	4	5	6	7	8	9
11	10	11	12	13	14	15	16
12	17	18	19	20	21	22	23
13	24	25	26	27	28	29	30
14	31						

April							
KW	Mo	Di	Mi	Do	Fr	Sa	So
14		1	2	3	4	5	6
15	7	8	9	10	11	12	13
16	14	15	16	17	18	19	20
17	21	22	23	24	25	26	27
18	28	29	30				

Mai							
KW	Mo	Di	Mi	Do	Fr	Sa	So
18				1	2	3	4
19	5	6	7	8	9	10	11
20	12	13	14	15	16	17	18
21	19	20	21	22	23	24	25
22	26	27	28	29	30	31	

Juni							
KW	Mo	Di	Mi	Do	Fr	Sa	So
22							1
23	2	3	4	5	6	7	8
24	9	10	11	12	13	14	15
25	16	17	18	19	20	21	22
26	23	24	25	26	27	28	29
27	30						

Juli							
KW	Mo	Di	Mi	Do	Fr	Sa	So
27		1	2	3	4	5	6
28	7	8	9	10	11	12	13
29	14	15	16	17	18	19	20
30	21	22	23	24	25	26	27
31	28	29	30	31			

August							
KW	Mo	Di	Mi	Do	Fr	Sa	So
31					1	2	3
32	4	5	6	7	8	9	10
33	11	12	13	14	15	16	17
34	18	19	20	21	22	23	24
35	25	26	27	28	29	30	31

September							
KW	Mo	Di	Mi	Do	Fr	Sa	So
36	1	2	3	4	5	6	7
37	8	9	10	11	12	13	14
38	15	16	17	18	19	20	21
39	22	23	24	25	26	27	28
40	29	30					

Oktober							
KW	Mo	Di	Mi	Do	Fr	Sa	So
40			1	2	3	4	5
41	6	7	8	9	10	11	12
42	13	14	15	16	17	18	19
43	20	21	22	23	24	25	26
44	27	28	29	30	31		

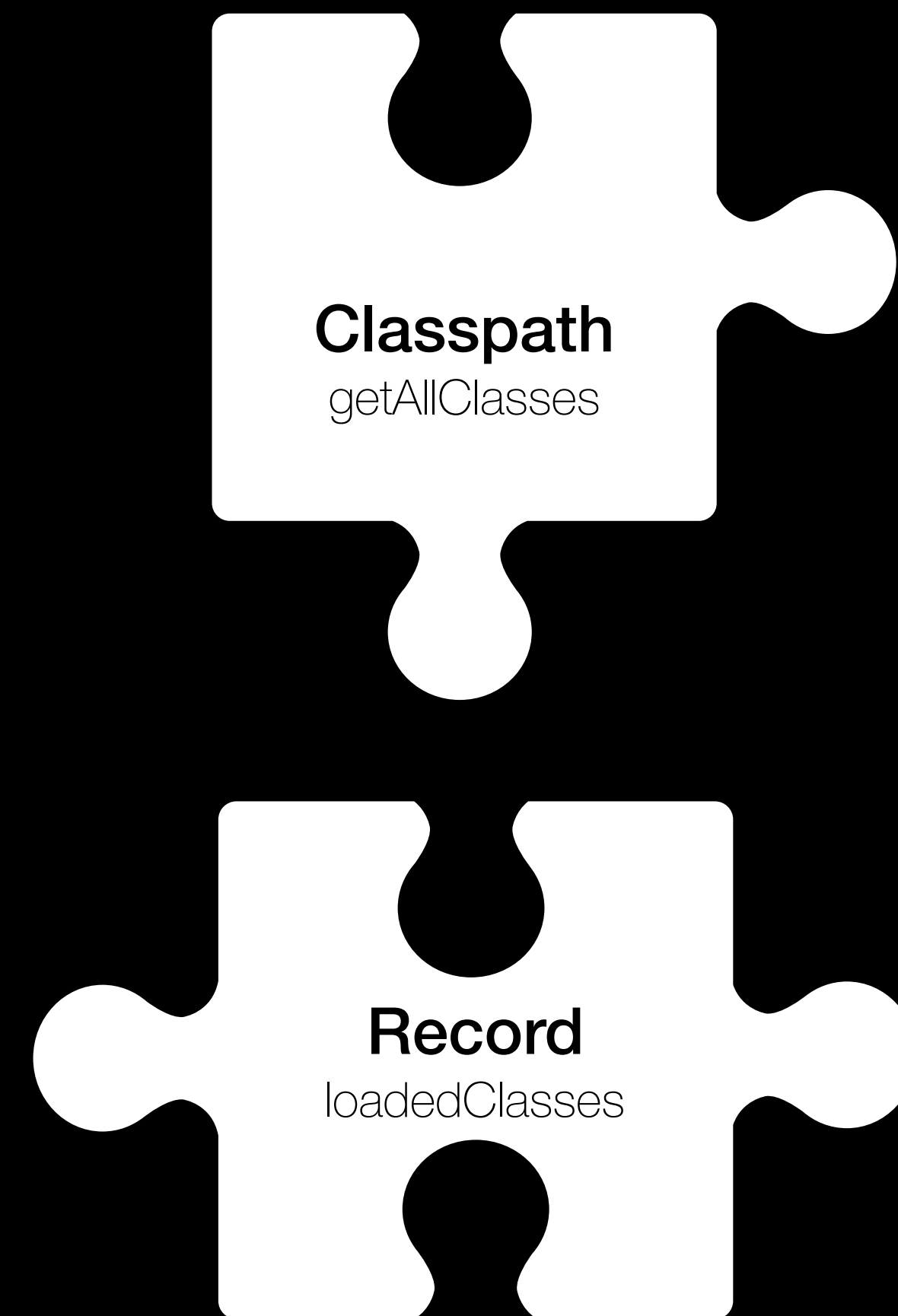
November							
KW	Mo	Di	Mi	Do	Fr	Sa	So
44						1	2
45	3	4	5	6	7	8	9
46	10	11	12	13	14	15	16
47	17	18	19	20	21	22	23
48	24	25	26	27	28	29	30

Dezember							
KW	Mo	Di	Mi	Do	Fr	Sa	So
49	1	2	3	4	5	6	7
50	8	9	10	11	12	13	14
51	15	16	17	18	19	20	21
52	22	23	24	25	26	27	28
1	29	30	31				

Gesetzliche Feiertage 2025 (bundesweit)

1. Januar	Neujahr	1. Mai	Tag der Arbeit	3. Oktober	Tag der Deutschen Einheit
18. April	Karfreitag	29. Mai	Christi Himmelfahrt	25. Dezember	1. Weihnachtstag
21. April	Ostermontag	9. Juni	Pfingstmontag	26. Dezember	2. Weihnachtstag

All Pieces Together



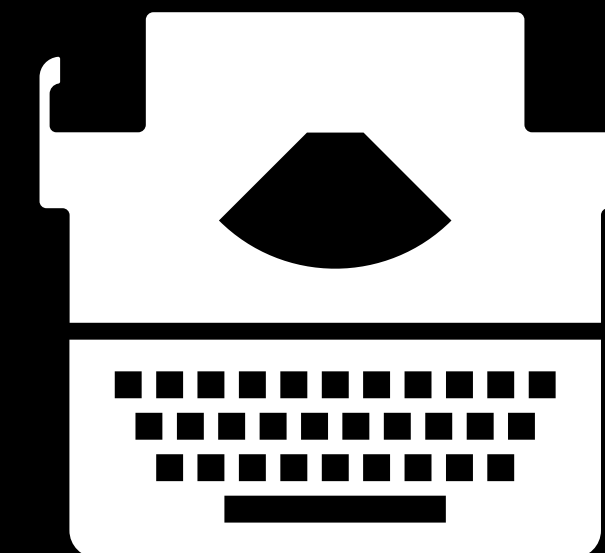
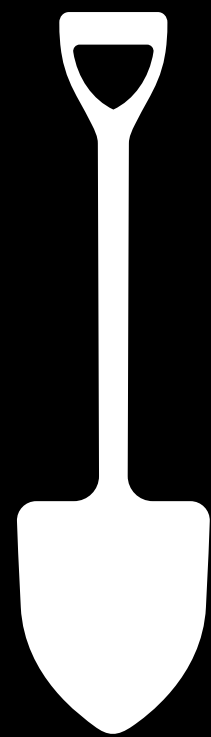
ClazzFish 🐟

DIY

ClasspathMonitor

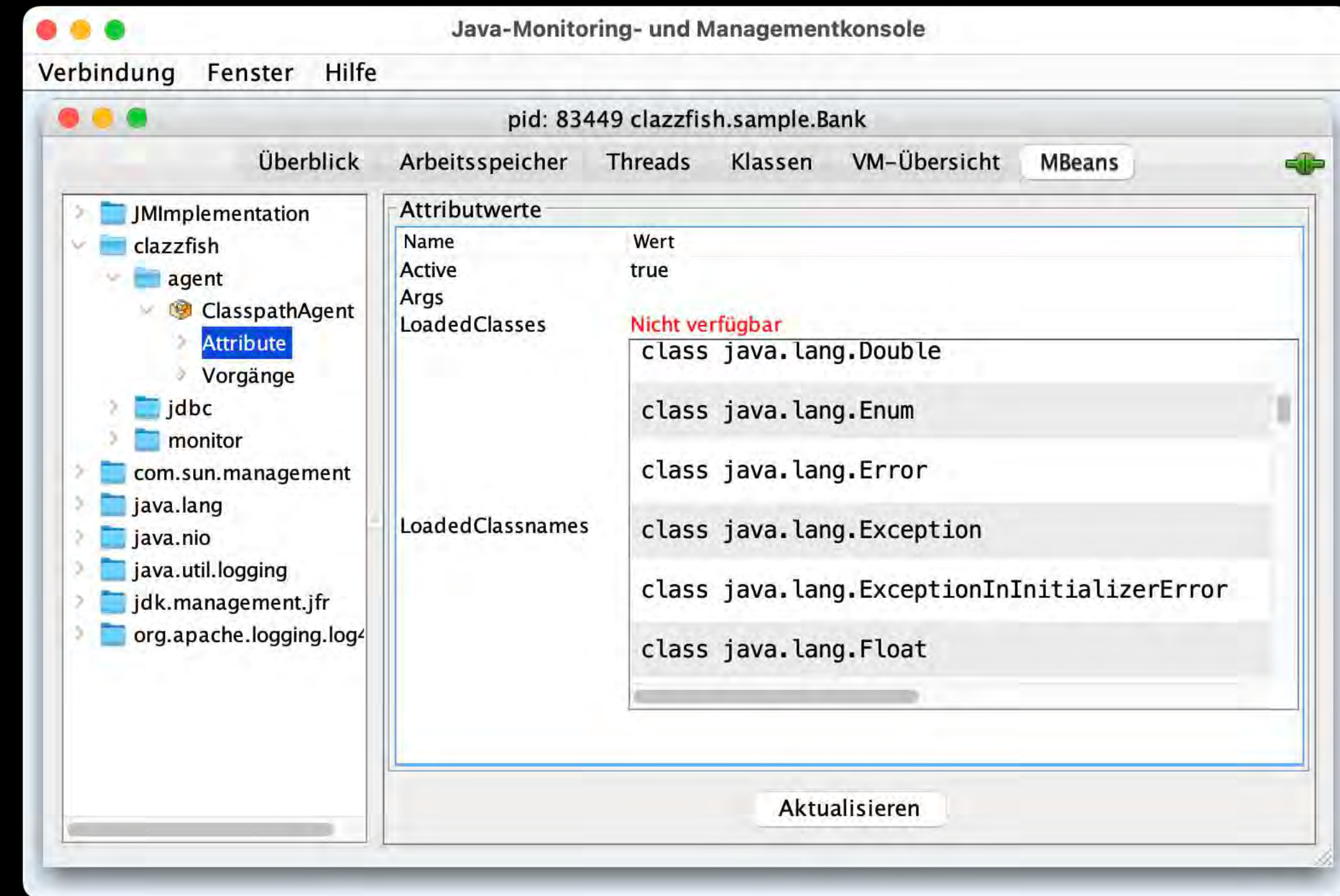


ClasspathAgent

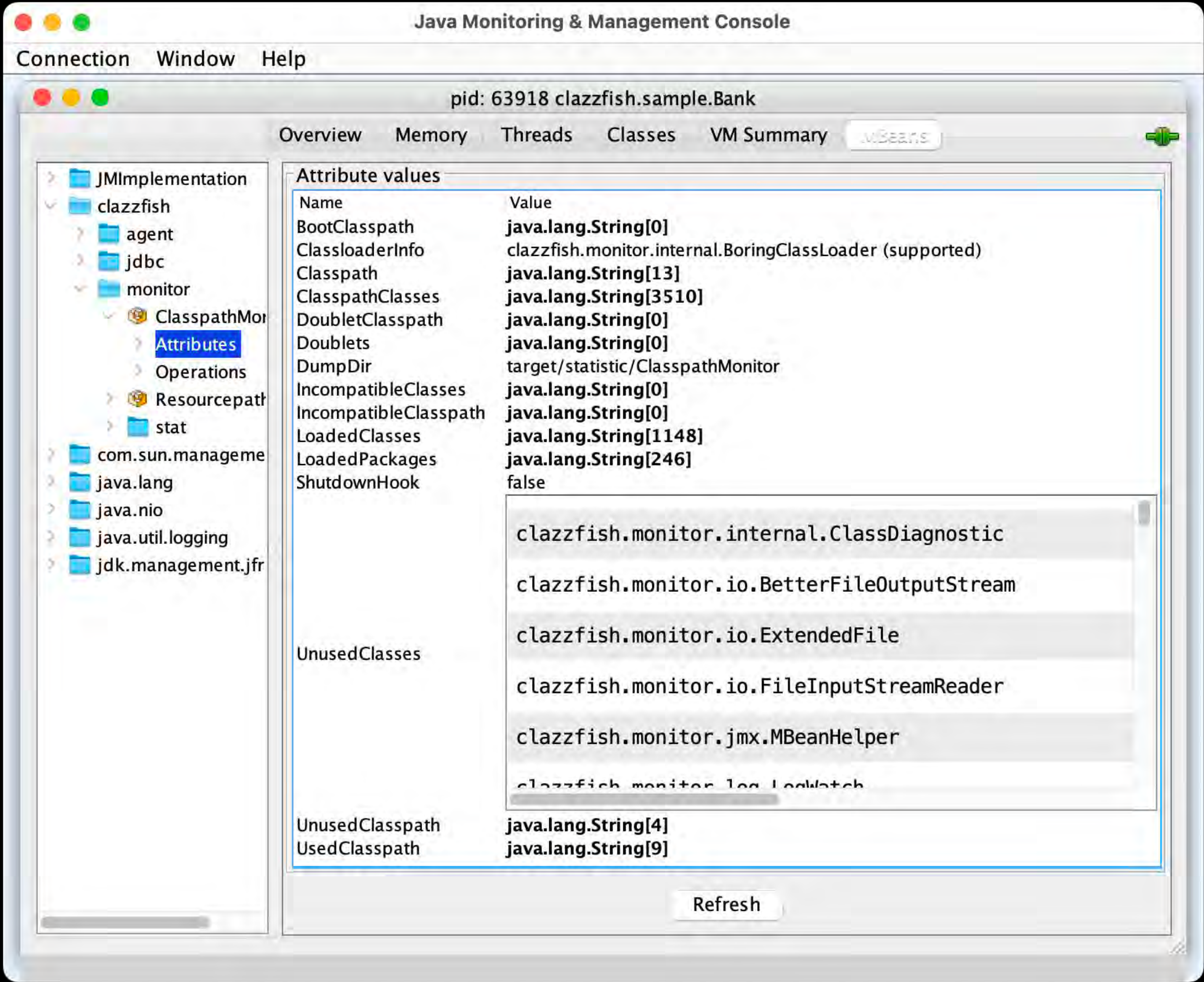


ClazzStatistic

ClasspathAgent



ClasspathMonitor



ClasspathMonitor



ClazzStatistic

Classpath	Classname	Count
file:/Users/o Boehm/work/github/en/ClazzFish/monitor/target/classes	clazzfish.monitor.AbstractMonitor	6
file:/Users/o Boehm/work/github/en/ClazzFish/monitor/target/classes	clazzfish.monitor.AbstractMonitorMBean	6
file:/Users/o Boehm/work/github/en/ClazzFish/monitor/target/classes	clazzfish.monitor.ClassloaderType	5
file:/Users/o Boehm/work/github/en/ClazzFish/monitor/target/classes	clazzfish.monitor.ClasspathMonitor	6
file:/Users/o Boehm/work/github/en/ClazzFish/monitor/target/classes	clazzfish.monitor.ClasspathMonitorMBean	6
file:/Users/o Boehm/work/github/en/ClazzFish/monitor/target/classes	clazzfish.monitor.ResourcepathMonitor	2
file:/Users/o Boehm/work/github/en/ClazzFish/monitor/target/classes	clazzfish.monitor.ResourcepathMonitorMBean	2
file:/Users/o Boehm/work/github/en/ClazzFish/monitor/target/classes	clazzfish.monitor.Starter	0
file:/Users/o Boehm/work/github/en/ClazzFish/monitor/target/classes	clazzfish.monitor.exception.ClassloaderException	6
file:/Users/o Boehm/work/github/en/ClazzFish/monitor/target/classes	clazzfish.monitor.exception.NotFoundException	6



ClazzStatistic



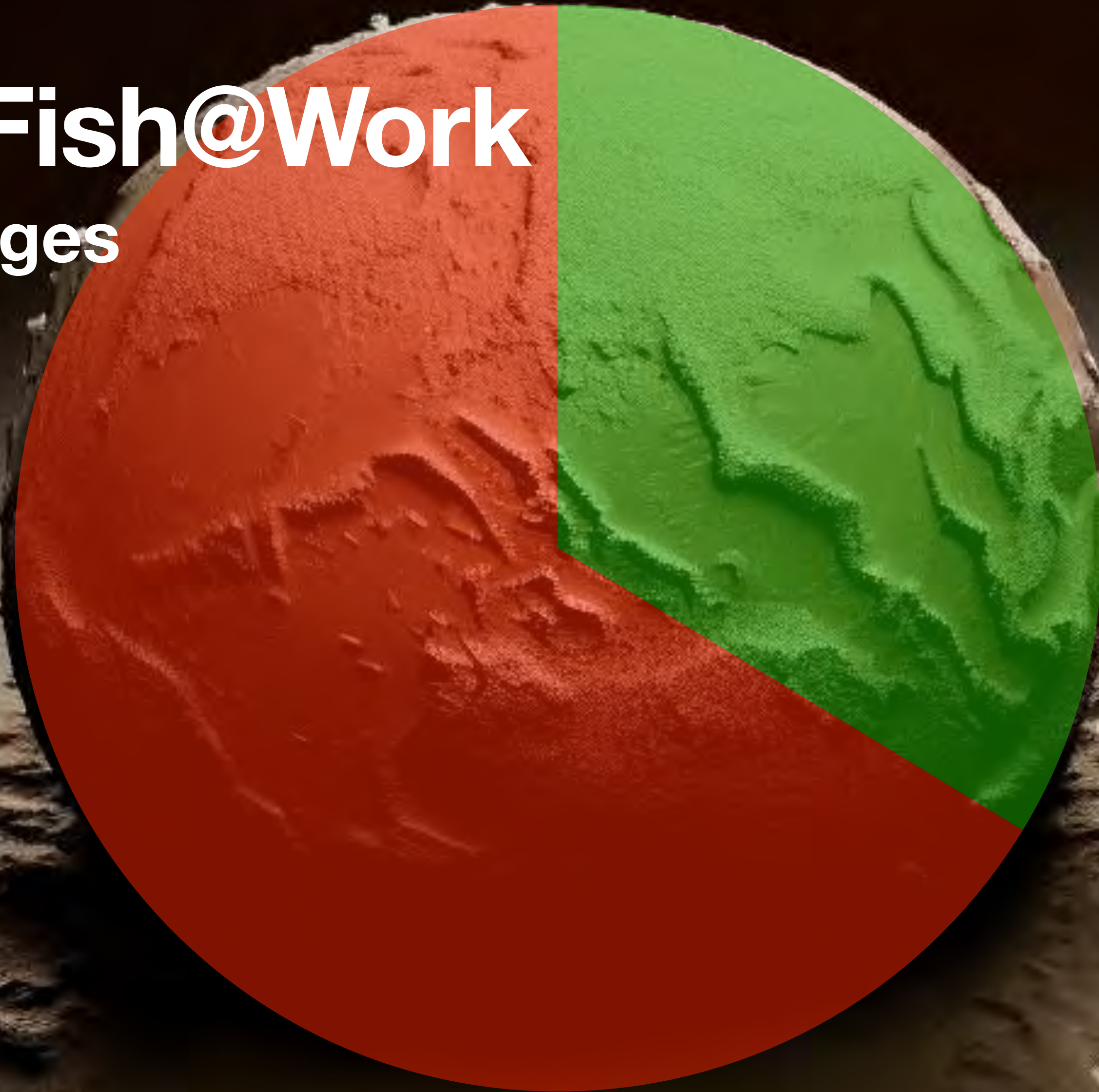
Demo?

ClazzFish@Work

swing package

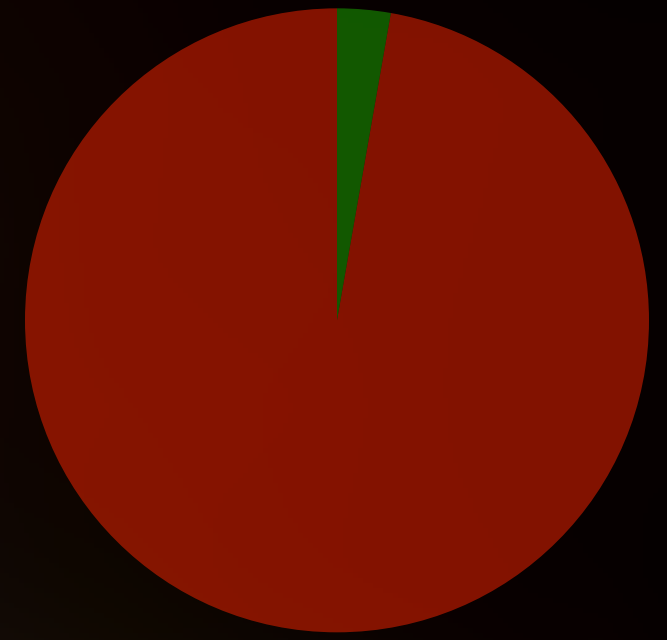
- simple approach
- no graphics
- manual evaluation

ClazzFish@Work
all packages



ClazzFish@Work

with dependencies



1/2025

Maintainability Overview ⓘ

(Only showing data for the first 500 files)

Color: Maintainability Rating

Size: Code Smells

☐ A

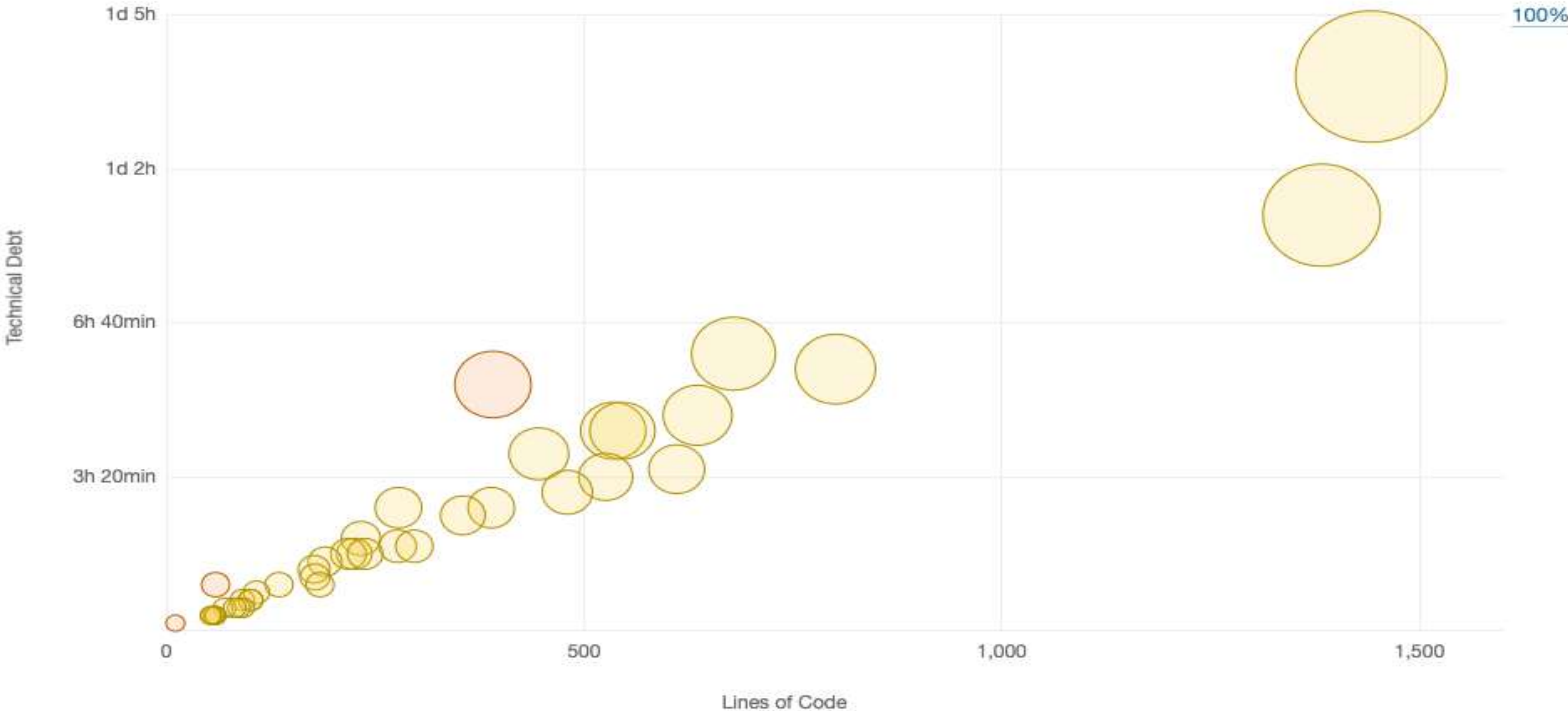
☐ B

☒ C

☒ D

☒ E

[See the data presented on this chart as a list](#)



(Only showing data for the first 500 files)

Color: Maintainability Rating

Size: Code Smells

☐ A

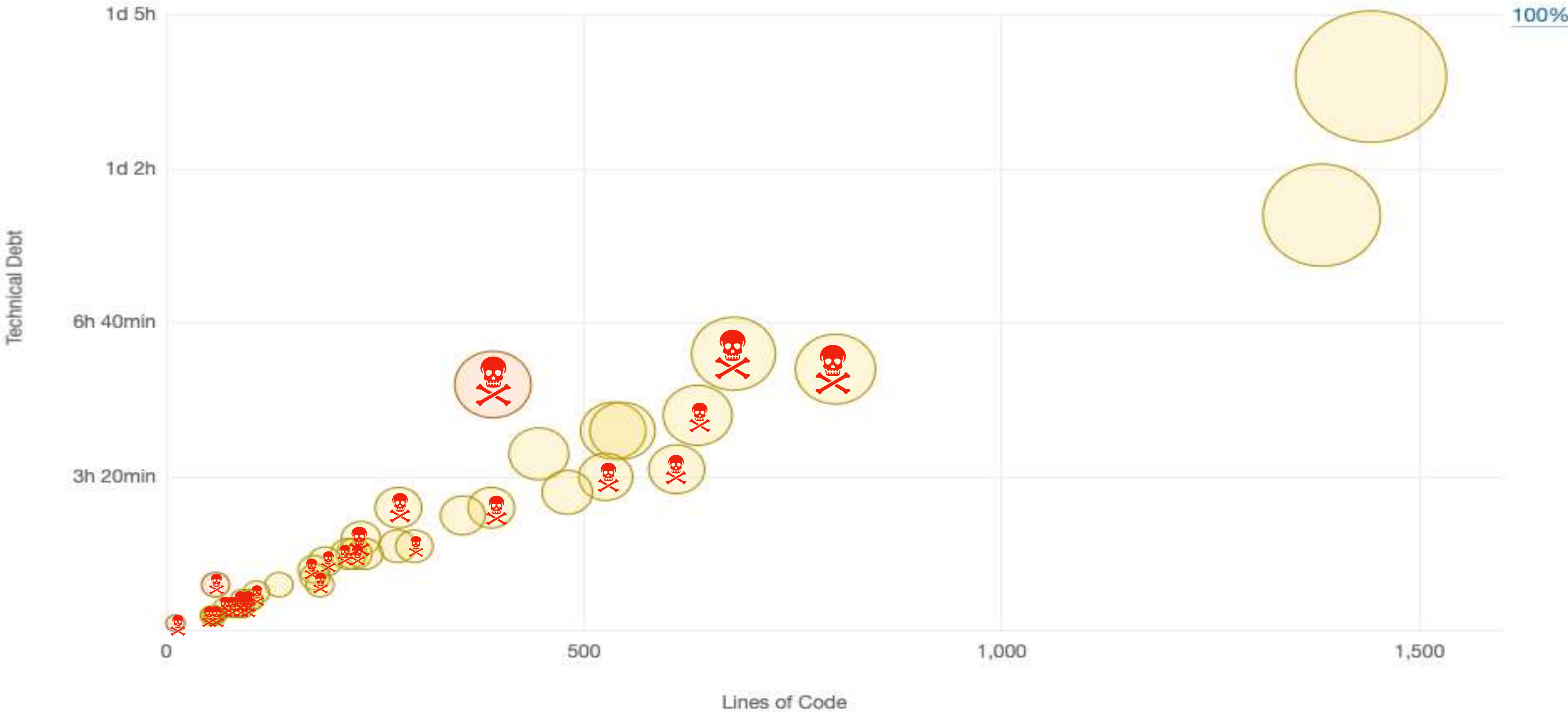
☐ B

☒ C

☒ D

☒ E

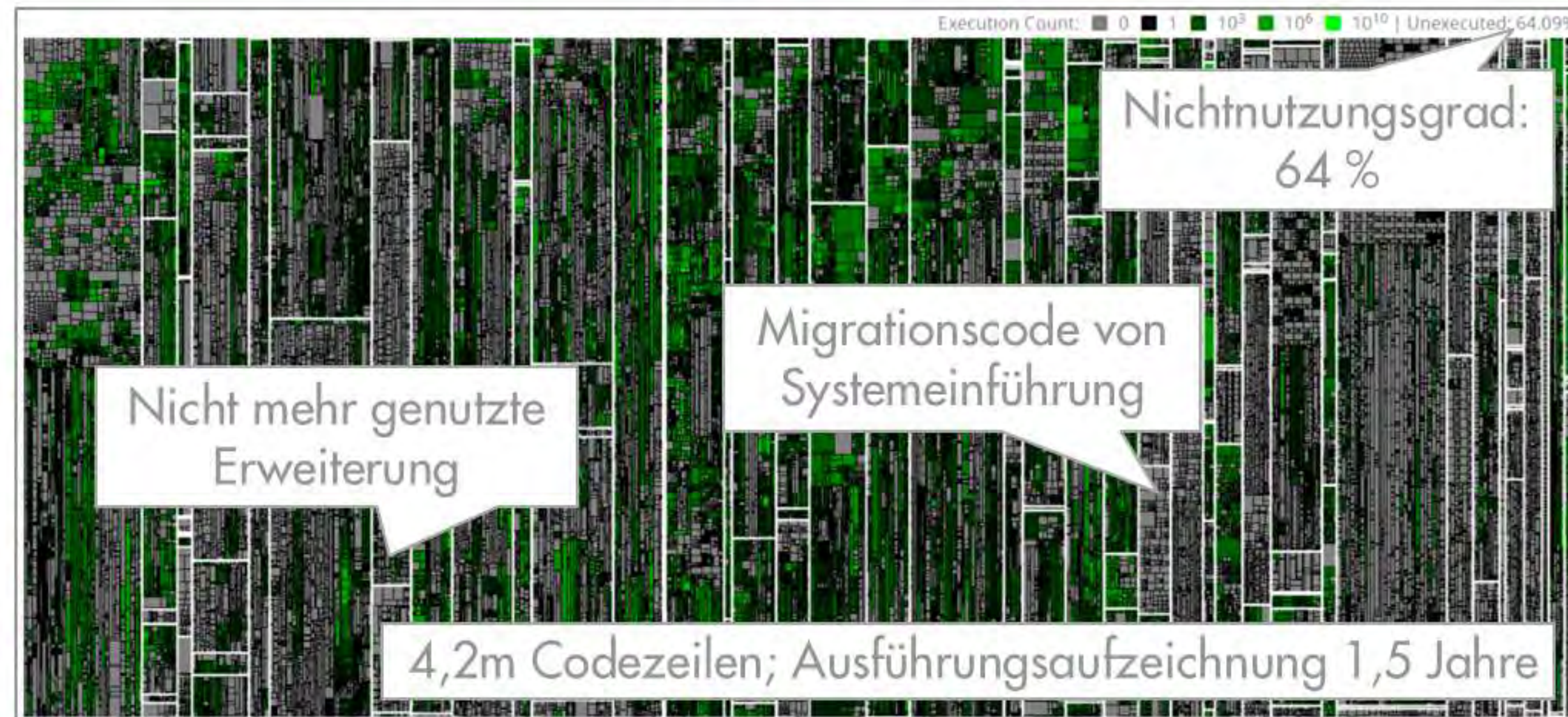
[See the data presented on this chart as a list](#)



Alternatives?

16.10.2025
JUGS Meetup

Nutzlosen Code mit Teamscale identifizieren



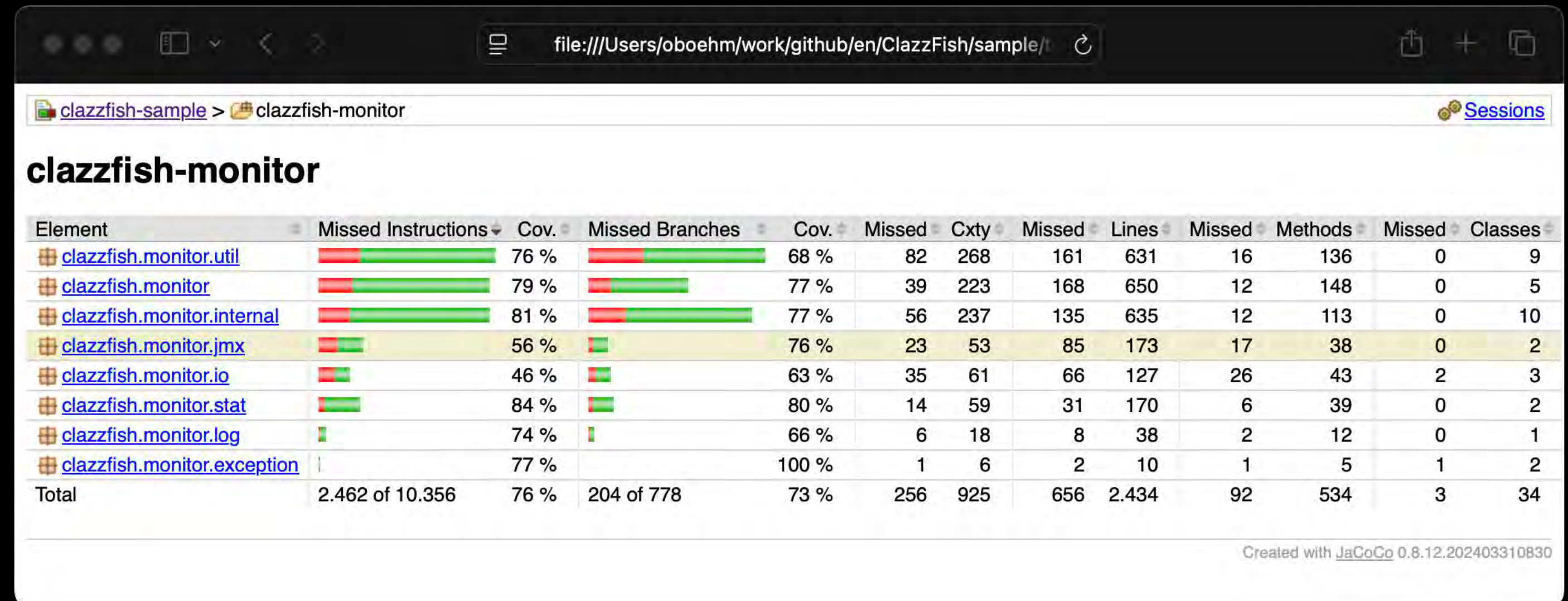
teamscale.com

Alternatives?

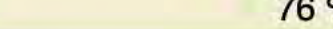
- <https://teamscale.com/>
- <https://github.com/parttimenerd/dead-code-agent>

Alternatives?

- <https://teamscale.com/>
- <https://github.com/parttimenerd/dead-code-agent>
- JaCoCo

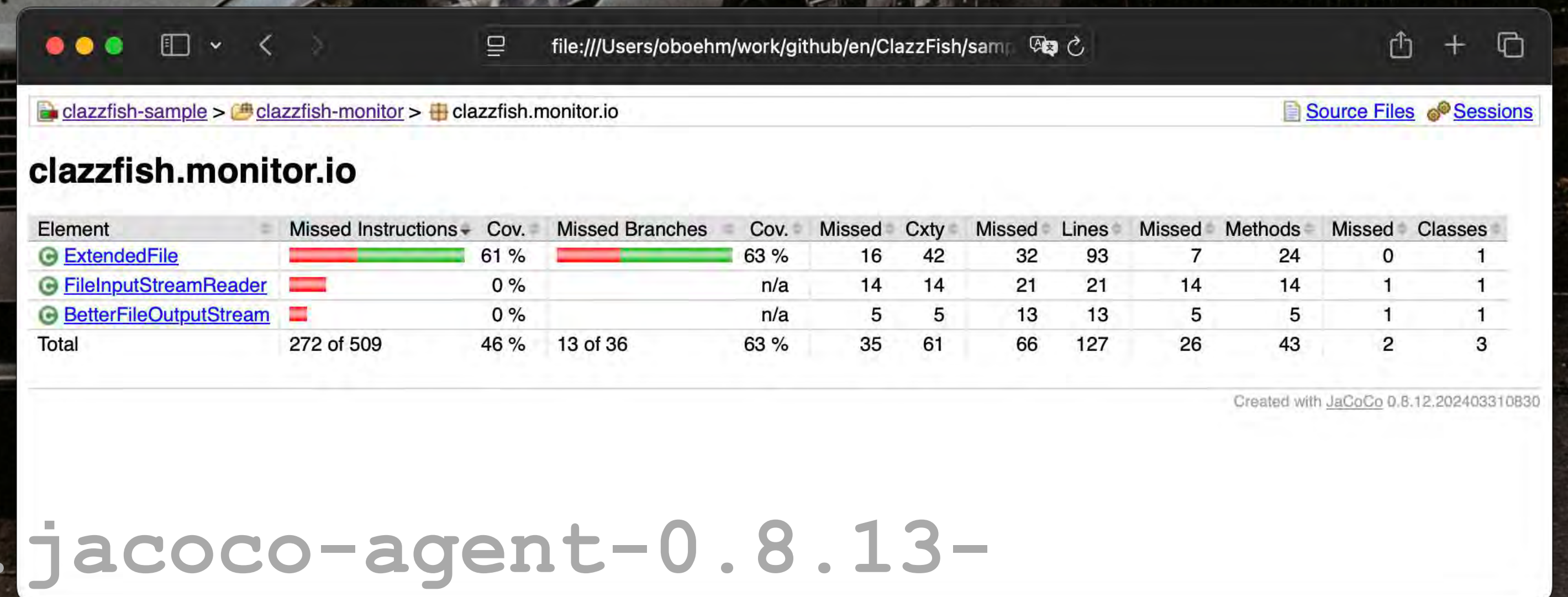


The screenshot shows a JaCoCo coverage report for the 'clazzfisch-monitor' project. The report is displayed in a web browser window with the file path 'file:///Users/oboehm/work/github/en/ClazzFish/sample/'. The report title is 'clazzfisch-monitor'. The table below provides a detailed breakdown of coverage metrics for various elements of the project.

Element	Missed Instructions	Cov.	Missed Branches	Cov.	Missed	Cxty	Missed	Lines	Missed	Methods	Missed	Classes
clazzfisch.monitor.util		76 %		68 %	82	268	161	631	16	136	0	9
clazzfisch.monitor		79 %		77 %	39	223	168	650	12	148	0	5
clazzfisch.monitor.internal		81 %		77 %	56	237	135	635	12	113	0	10
clazzfisch.monitor.jmx		56 %		76 %	23	53	85	173	17	38	0	2
clazzfisch.monitor.io		46 %		63 %	35	61	66	127	26	43	2	3
clazzfisch.monitor.stat		84 %		80 %	14	59	31	170	6	39	0	2
clazzfisch.monitor.log		74 %		66 %	6	18	8	38	2	12	0	1
clazzfisch.monitor.exception		77 %		100 %	1	6	2	10	1	5	1	2
Total	2.462 of 10.356	76 %	204 of 778	73 %	256	925	656	2.434	92	534	3	34

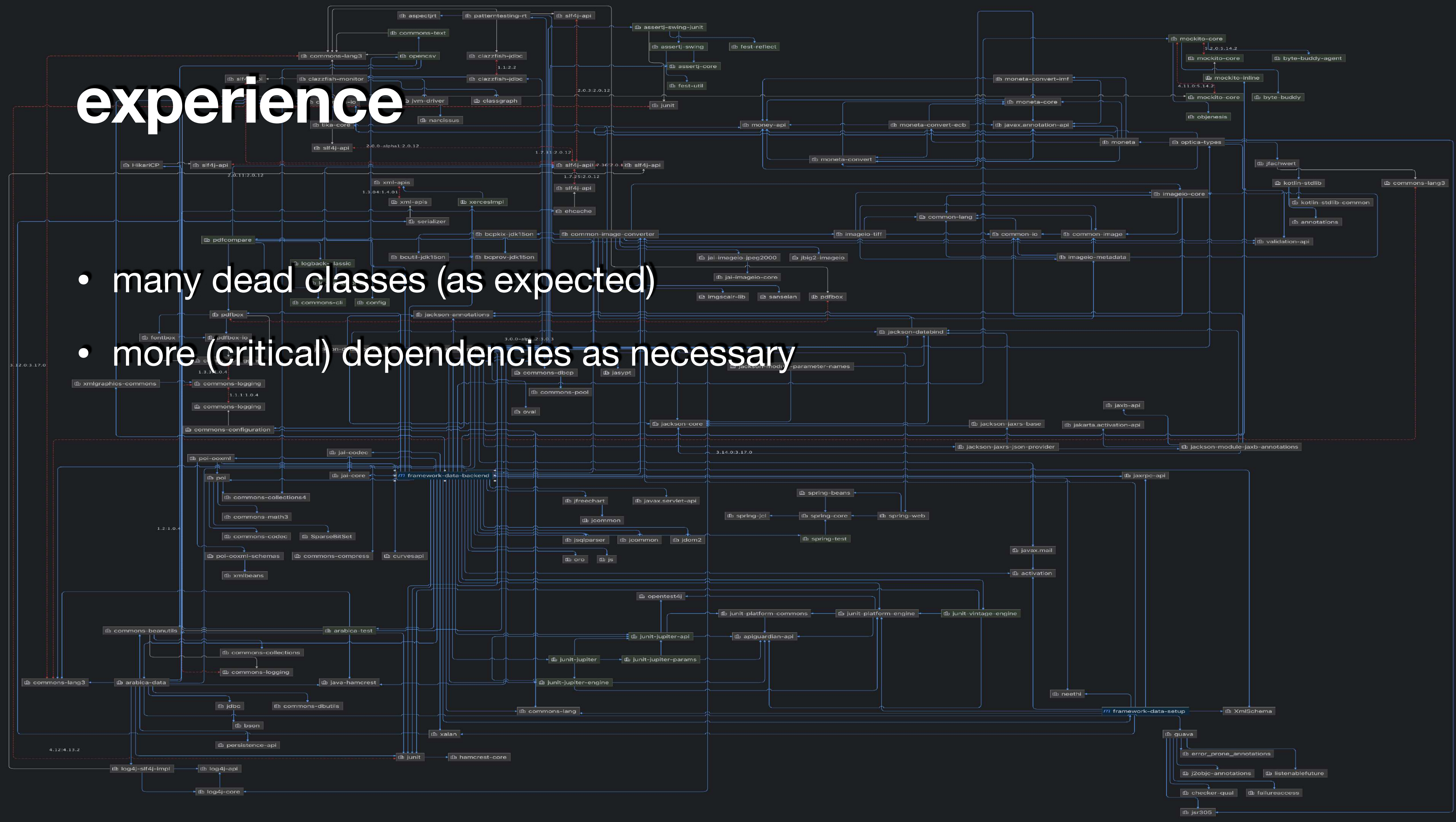
Created with JaCoCo 0.8.12.202403310830

JaCoCo

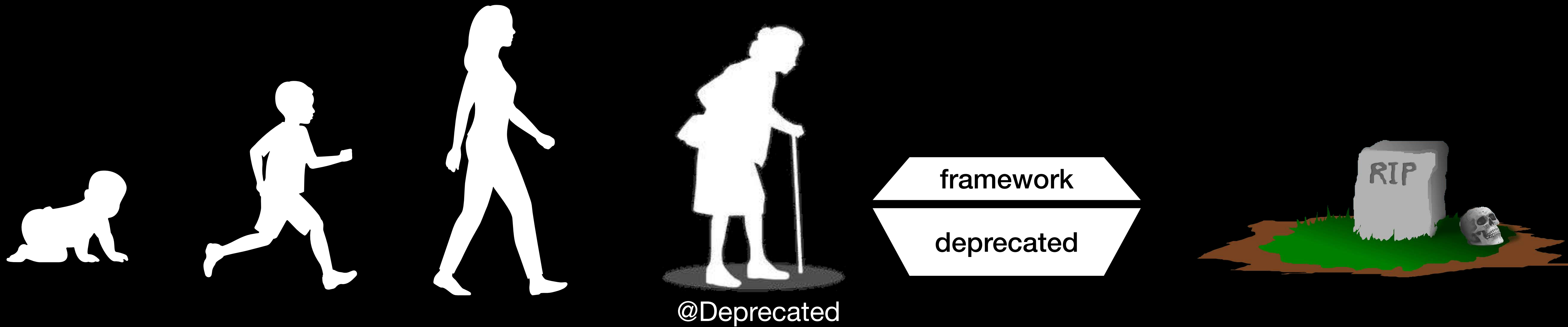


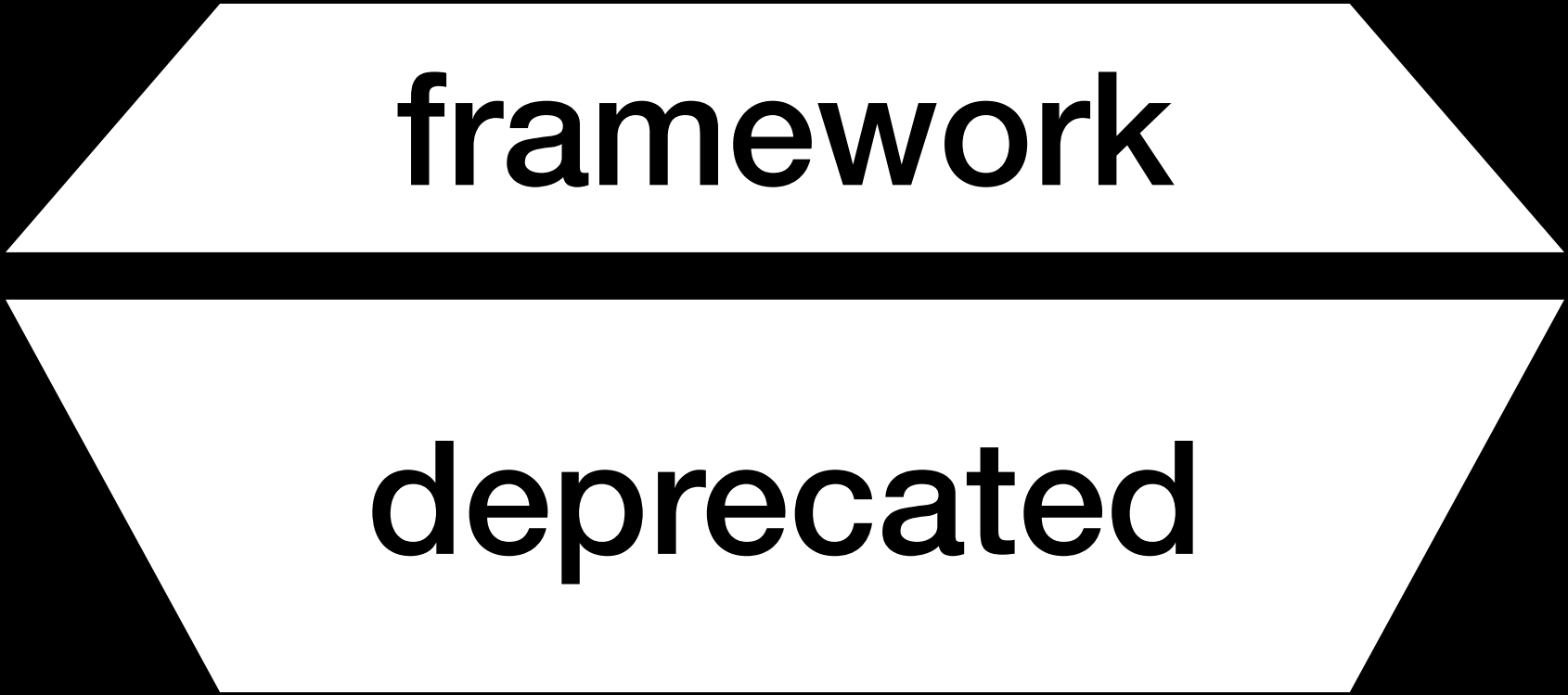
```
java -javaagent:org.jacoco-agent-0.8.13-  
runtime.jar=destfile=log/jacoco.exec,append=true...
```

- many dead classes (as expected)
- more (critical) dependencies as necessary



The Classes Lifecycle





framework

deprecated

Next Steps

- assess dead code
- mark dead classes as `@Deprecated`
- move them to framework-deprecated
- reduce dependencies
- clean framework-deprecated



Software rots



*Perfection is not achieved
when there is noting to add,
but when there is nothing more to leave out.*

(Antoine de Saint-Exupéry)



<https://github.com/oboekm/ClazzFish>



oboekm (Juli-2025)

25.-26.9.2025
JUG Saxony Day

*Perfection is not achieved
when there is noting to add,
but when there is nothing more to leave out.*

(Antoine de Saint-Exupéry)



<https://github.com/oboekm/ClazzFish>



oboekm (Juli-2025)