Što kad se (s)ruši ili stane Java aplikacija?

Matija Tomašković | Evolva, CEO

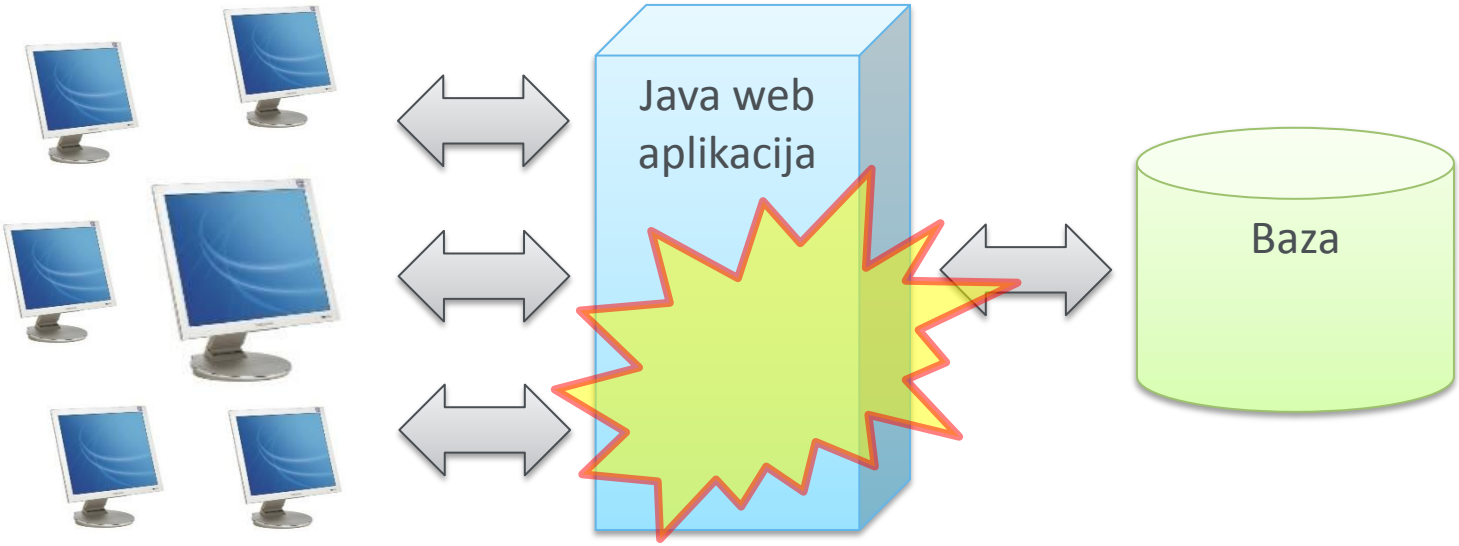
JavaCro'15

12.05.2015.

POSLOVNE WEB APLIKACIJE

EVOLVA

Java web aplikacije u enterprise okruženju



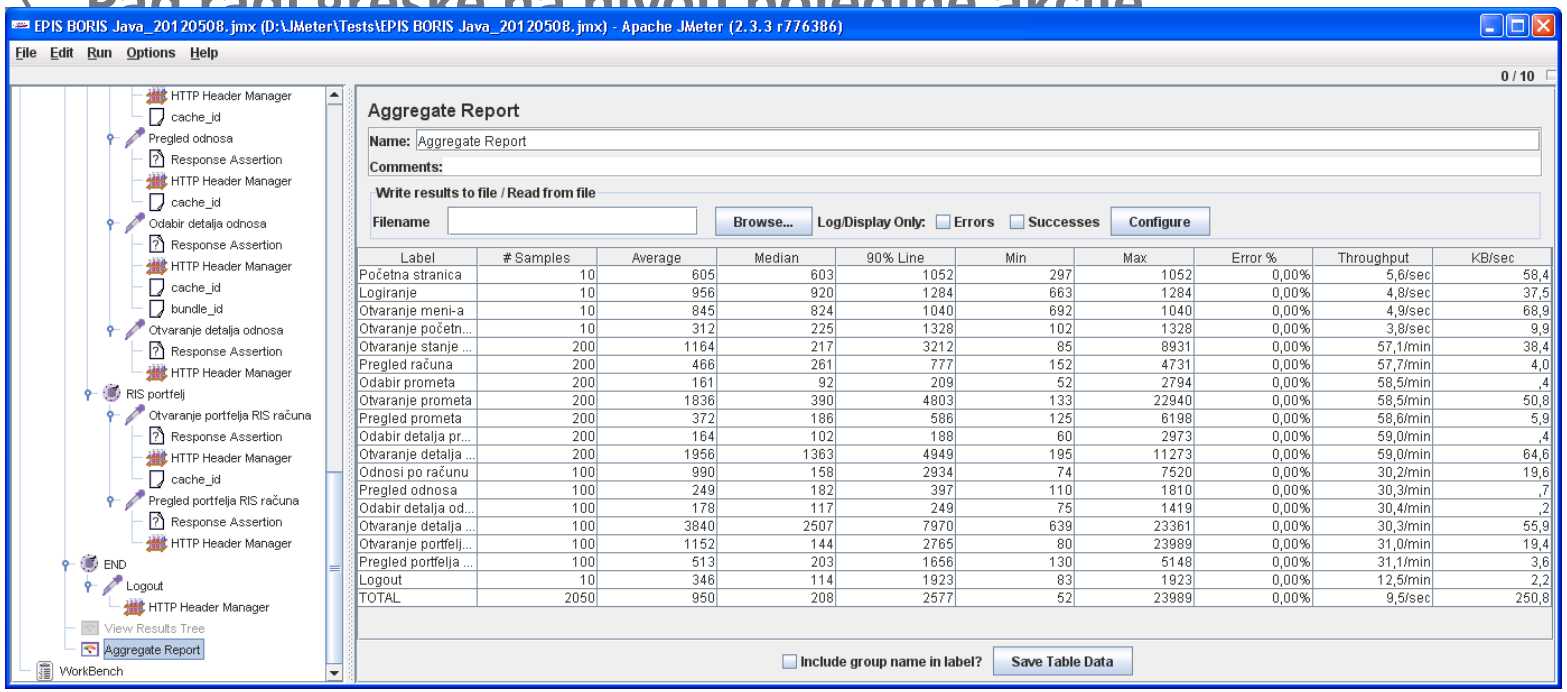
- **enterprise okruženje** - veliki broj korisnika
- **rušenje/zastoj aplikacije** – moguće ozbiljne **posljedice**

Najčešći razlozi pada Java web aplikacija

- **Pad radi greške na nivou pojedine akcije**
  - česti uzrok: nedostatak granica
  - prevencija: short term scenariji / unit testovi
- **Pad radi prevelikog opterećenja**
  - prevencija: peak test
- **Pad radi dugoročne anomalije**
  - prevencija : long-term testovi

Najčešći razlozi pada Java web aplikacija

- Pad radi greške na nivou pojedine akcije



Label	# Samples	Average	Median	90% Line	Min	Max	Error %	Throughput	KB/sec
Početna stranica	10	605	603	1052	297	1052	0,00%	5,6/sec	58,4
Logiranje	10	956	920	1284	663	1284	0,00%	4,8/sec	37,5
Otvaranje meni-a	10	845	824	1040	692	1040	0,00%	4,9/sec	68,9
Otvaranje početn...	10	312	225	1328	102	1328	0,00%	3,8/sec	9,9
Otvaranje stanje ...	200	1164	217	3212	85	8931	0,00%	57,1/min	38,4
Pregled računa	200	466	261	777	152	4731	0,00%	57,7/min	4,0
Odabir prometa	200	161	92	209	52	2794	0,00%	58,5/min	,4
Otvaranje prometa	200	1836	390	4803	133	22940	0,00%	58,5/min	50,8
Pregled prometa	200	372	186	586	125	6198	0,00%	58,6/min	5,9
Odabir detalja pr...	200	164	102	188	60	2973	0,00%	59,0/min	,4
Otvaranje detalja ...	200	1956	1363	4949	195	11273	0,00%	59,0/min	64,6
Odnosi po računu	100	990	158	2934	74	7520	0,00%	30,2/min	19,6
Pregled odnosa	100	249	182	397	110	1810	0,00%	30,3/min	,7
Odabir detalja od...	100	178	117	249	75	1419	0,00%	30,4/min	,2
Otvaranje detalja ...	100	3840	2507	7970	639	23361	0,00%	30,3/min	55,9
Otvaranje portfelj...	100	1152	144	2765	80	23989	0,00%	31,0/min	19,4
Pregled portfelja ...	100	513	203	1656	130	5148	0,00%	31,1/min	3,6
Logout	10	346	114	1923	83	1923	0,00%	12,5/min	2,2
TOTAL	2050	950	208	2577	52	23989	0,00%	9,5/sec	250,8

Klasifikacija pada sustava: rušenje ili zastoj?

RUŠENJE

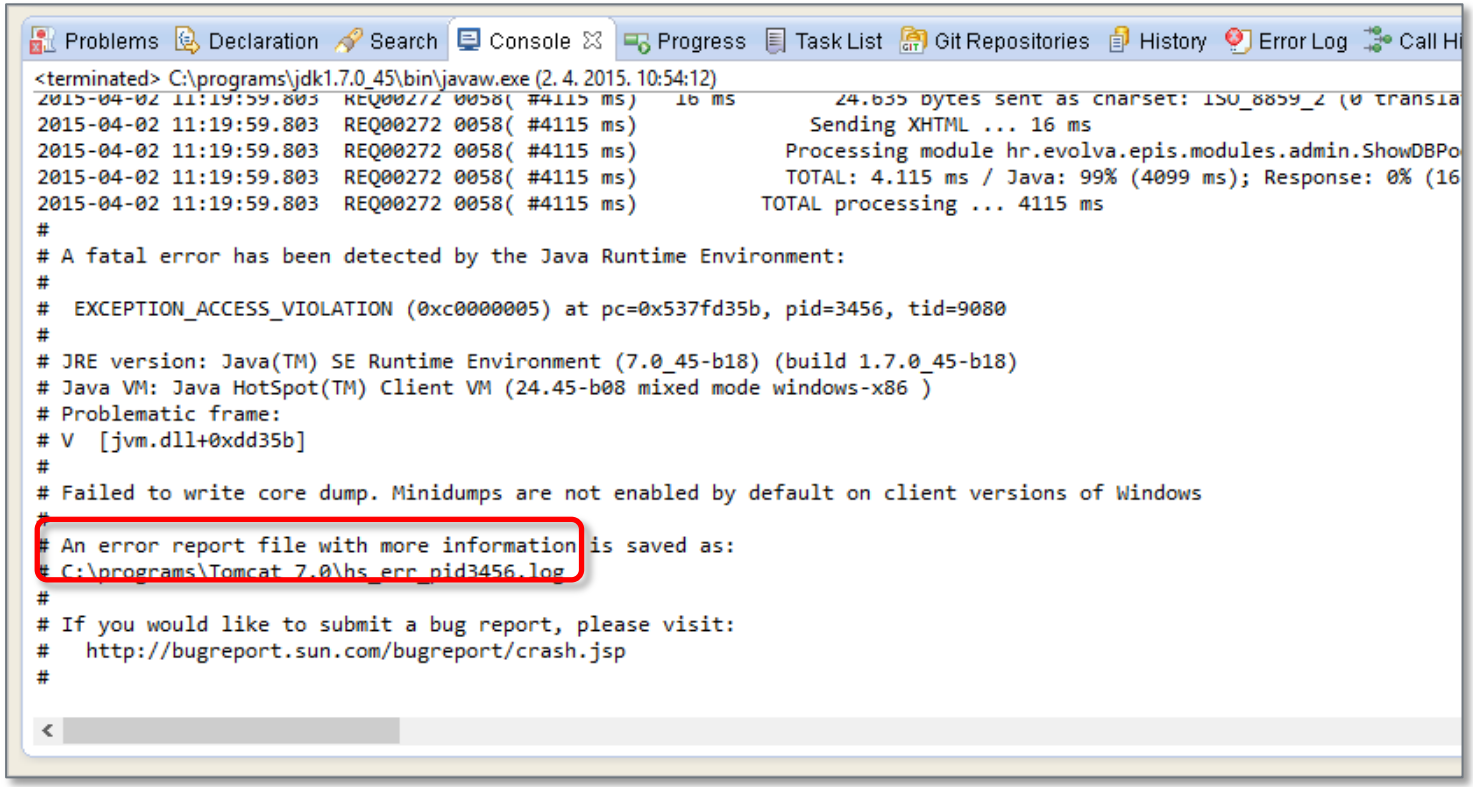
- JVM crash

ZASTOJI

- Zauzeće resursa
  - Memorija
  - Thread pool
  - Disk
  - Baza: connection pools, statements, ...
- Loša sinhronizacija threadova
  - puno blocked threadova
  - thread deadlock
- Mrtva petlja

JVM crash - analiza

Situacija: JVM crash



- hs_err_pid*.log datoteku kreira JVM kod rušenja
- na vrhu je razlog

Situacija: JVM crash

Gdje je file?

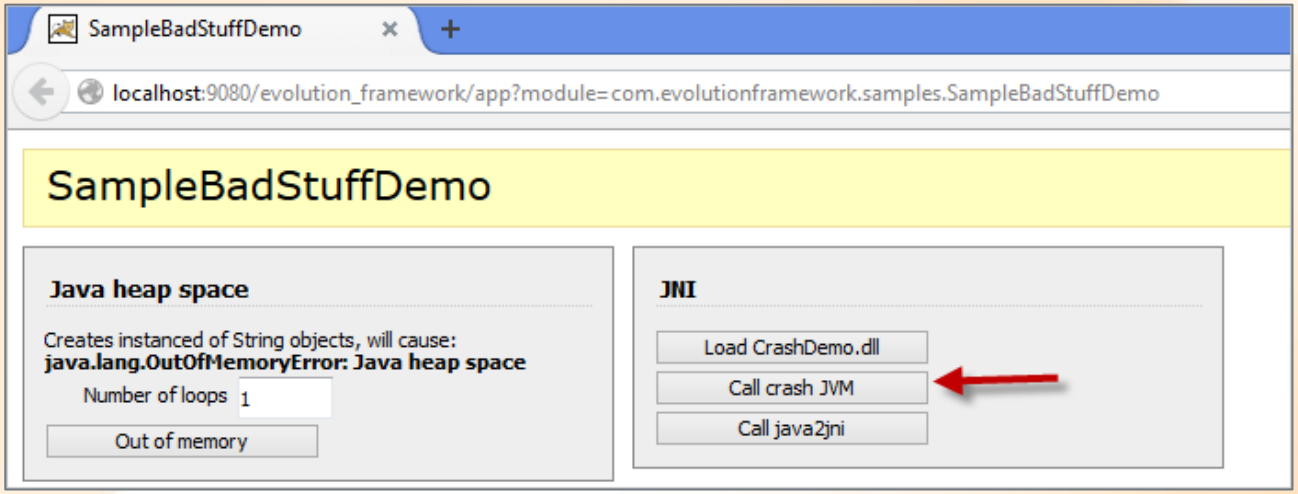
- datoteka je locirana gdje je pokrenut JVM proces
- ako je u pitanju bio Windows servis, onda može završiti u C:\Windows\System32 (za 32bit)
- Java 6+:
  - -XX:ErrorFile=./hs_err_pid<pid>.log

Uzrok?

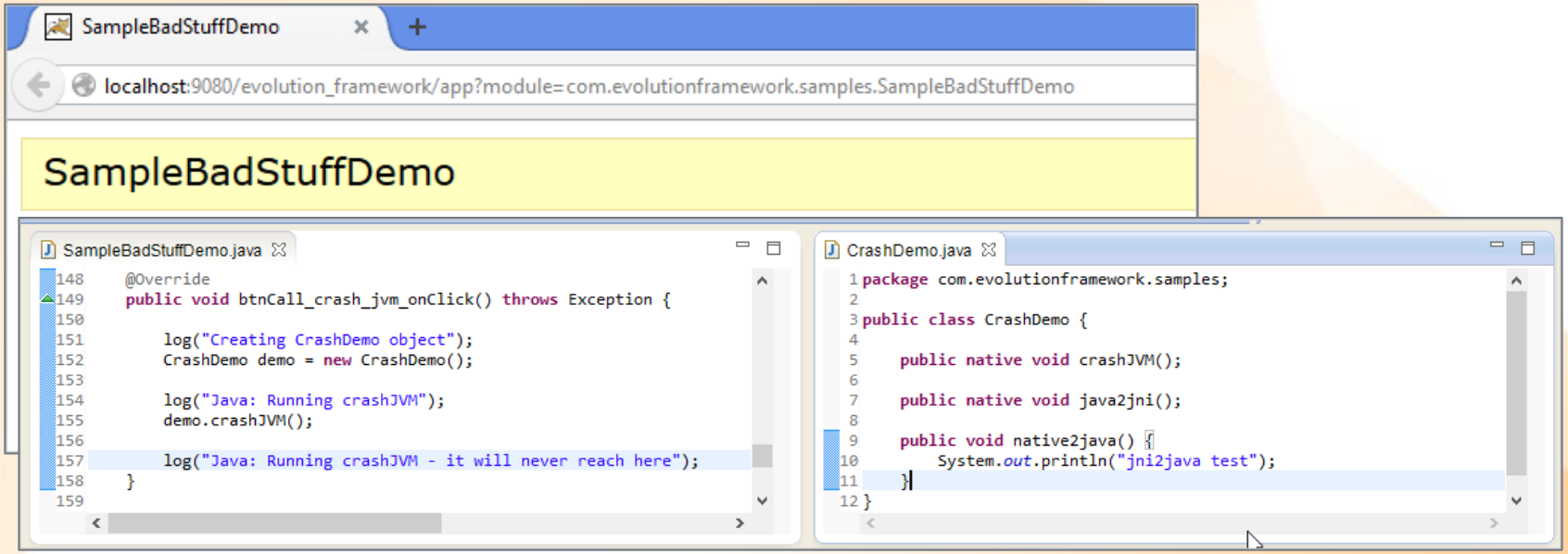
- ako koristite native kod – vrlo vjerojatno poziv vanjske native funkcije
- ako ne koristite – onda JVM bug – prijaviti grešku i uzeti novi JDK

```
Stack: [0x210c0000,0x21110000],  sp=0x2110f8c4,  free space=318k
Native frames: (J=compiled Java code, j=interpreted, Vv=VM code, C=native code)
V  [jvm.dll+0xdd35b]
V  [jvm.dll+0x20e04a]
C  [jdwp.dll+0x19d9d]
C  [jdwp.dll+0x44e0]
C  [jdwp.dll+0x451e]
```

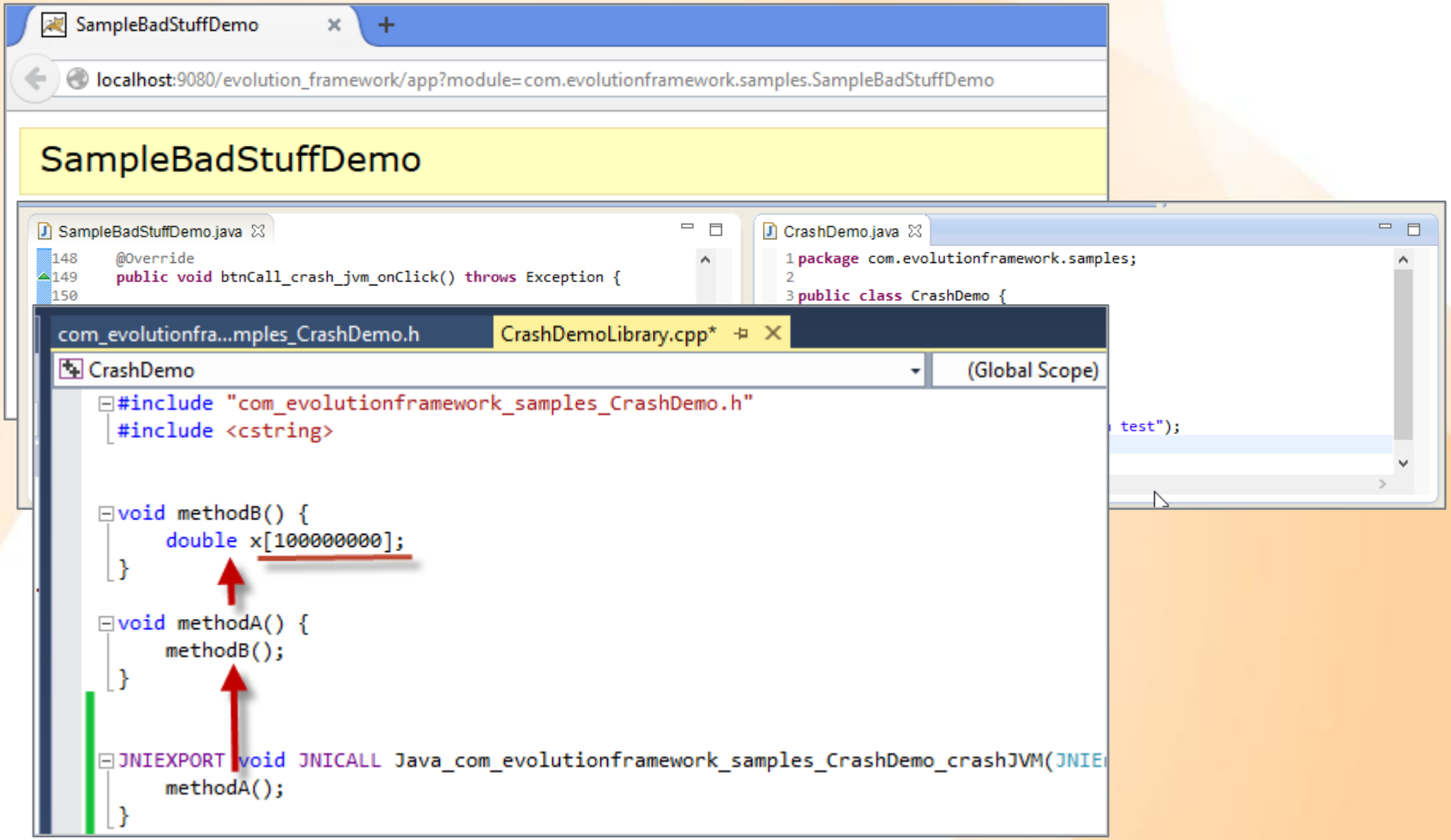
DEMO:
JVM crash – native code


SampleBadStuffDemo
localhost:9080/evolution_framework/app?module=com.evolutionframework.samples.SampleBadStuffDemo
SampleBadStuffDemo
Java heap space
Creates instanced of String objects, will cause:
java.lang.OutOfMemoryError: Java heap space
Number of loops 1
Out of memory
JNI
Load CrashDemo.dll
Call crash JVM
Call java2jni

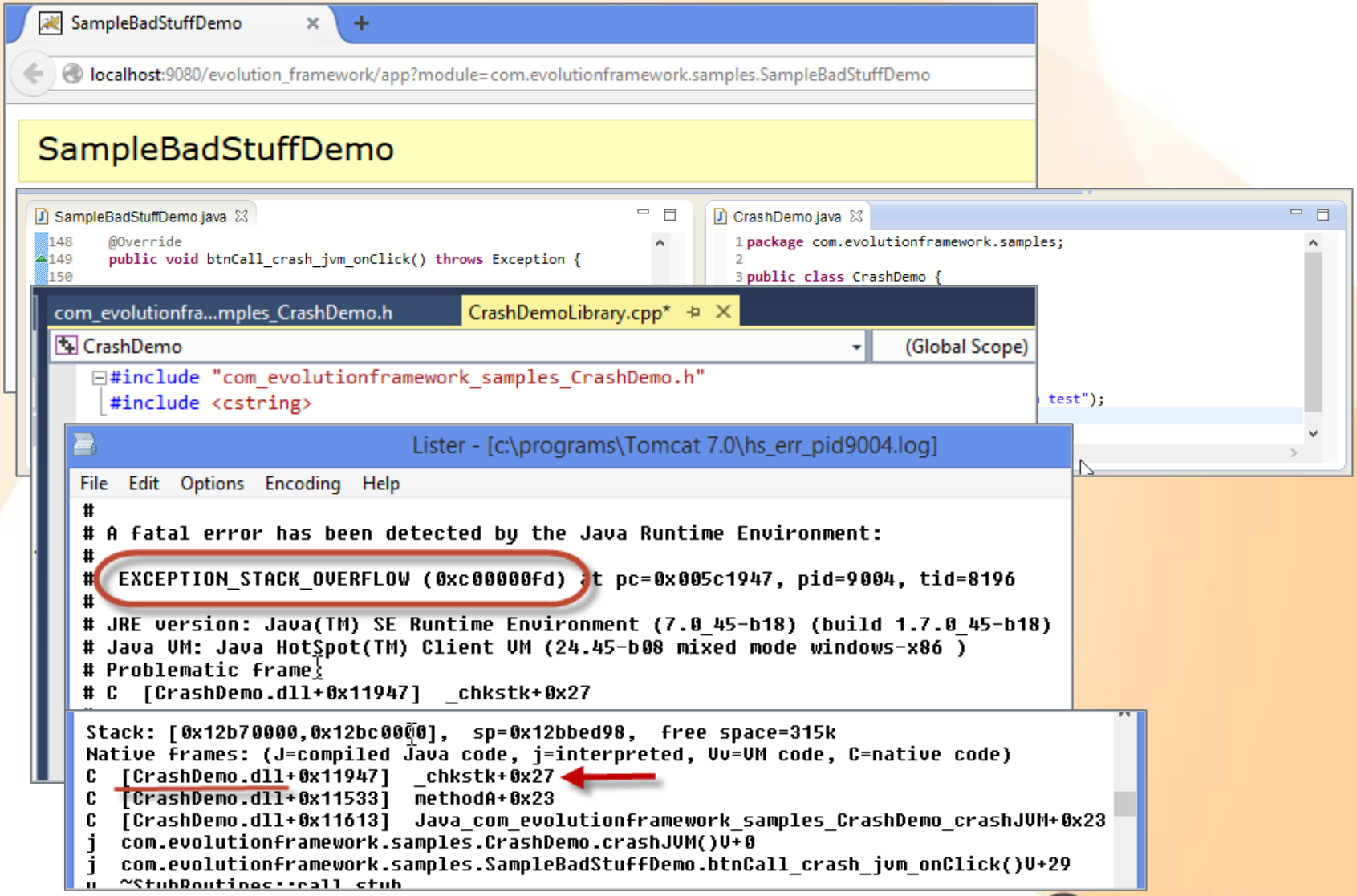
EVOLVA


SampleBadStuffDemo
localhost:9080/evolution_framework/app?module=com.evolutionframework.samples.SampleBadStuffDemo
SampleBadStuffDemo
SampleBadStuffDemo.java
148 @Override
149 public void btnCall_crash_jvm_onClick() throws Exception {
150
151 log("Creating CrashDemo object");
152 CrashDemo demo = new CrashDemo();
153
154 log("Java: Running crashJVM");
155 demo.crashJVM();
156
157 log("Java: Running crashJVM - it will never reach here");
158 }
159
CrashDemo.java
1 package com.evolutionframework.samples;
2
3 public class CrashDemo {
4
5 public native void crashJVM();
6
7 public native void java2jni();
8
9 public void native2java() {
10 System.out.println("jni2java test");
11 }
12 }

EVOLVA


SampleBadStuffDemo
localhost:9080/evolution_framework/app?module=com.evolutionframework.samples.SampleBadStuffDemo
SampleBadStuffDemo
SampleBadStuffDemo.java
148 @Override
149 public void btnCall_crash_jvm_onClick() throws Exception {
150
CrashDemo.java
1 package com.evolutionframework.samples;
2
3 public class CrashDemo {
test");
com_evolutionfra...mples_CrashDemo.h
CrashDemoLibrary.cpp*
CrashDemo
(Global Scope)
#include "com_evolutionframework_samples_CrashDemo.h"
#include <cstring>
void methodB() {
double x[100000000];
}
void methodA() {
methodB();
}
JNIEXPORT void JNICALL Java_com_evolutionframework_samples_CrashDemo_crashJVM(JNIE
methodA();
}

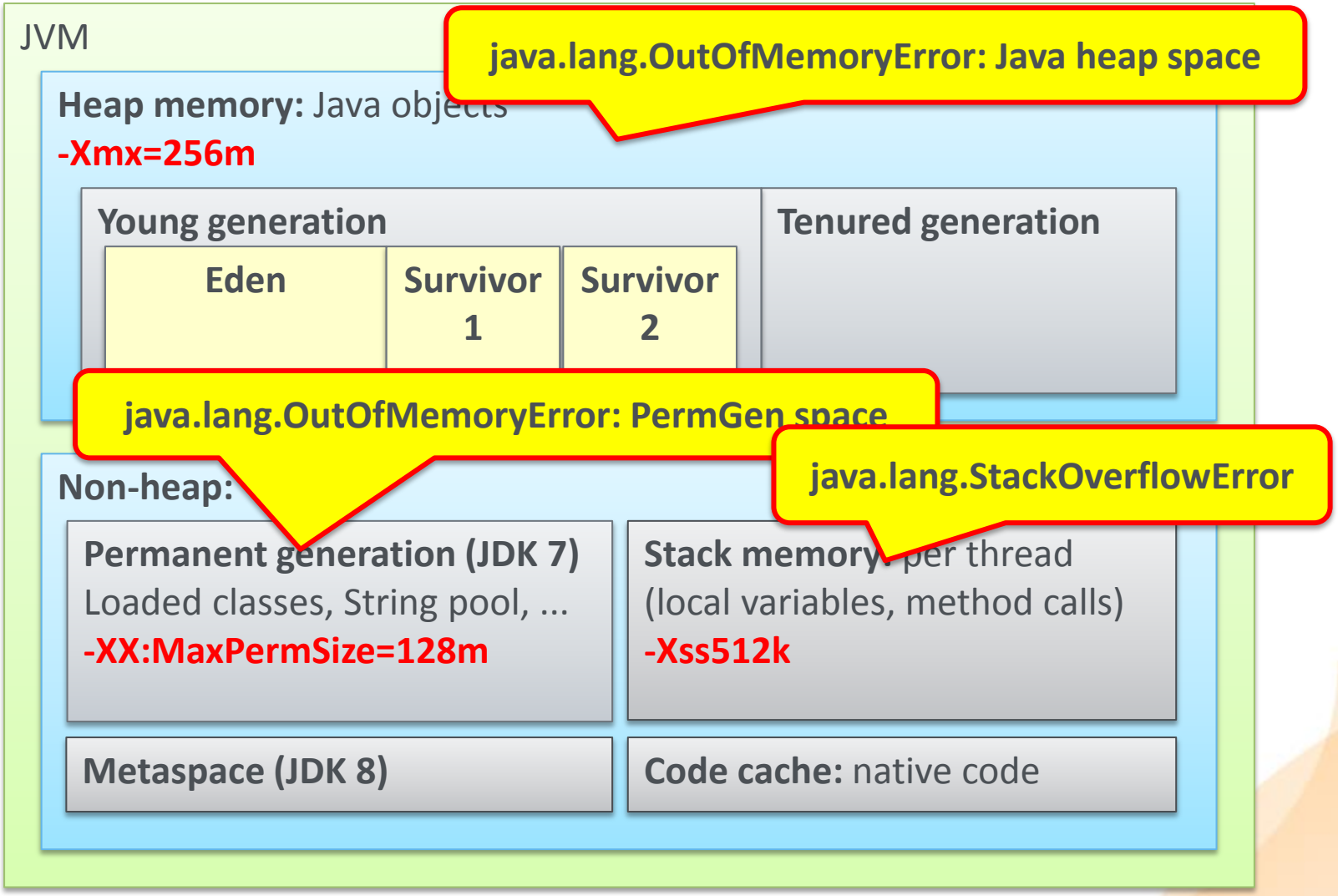
EVOLVA


SampleBadStuffDemo
localhost:9080/evolution_framework/app?module=com.evolutionframework.samples.SampleBadStuffDemo
SampleBadStuffDemo
SampleBadStuffDemo.java
148 @Override
149 public void btnCall_crash_jvm_onClick() throws Exception {
150
CrashDemo.java
1 package com.evolutionframework.samples;
2
3 public class CrashDemo {
com_evolutionfra...mples_CrashDemo.h
CrashDemoLibrary.cpp*
CrashDemo
(Global Scope)
#include "com_evolutionframework_samples_CrashDemo.h"
#include <cstring>
test");
Lister - [c:\programs\Tomcat 7.0\hs_err_pid9004.log]
File Edit Options Encoding Help
#
A fatal error has been detected by the Java Runtime Environment:
#
EXCEPTION_STACK_OVERFLOW (0xc00000fd) at pc=0x005c1947, pid=9004, tid=8196
#
JRE version: Java(TM) SE Runtime Environment (7.0_45-b18) (build 1.7.0_45-b18)
Java VM: Java HotSpot(TM) Client VM (24.45-b08 mixed mode windows-x86)
Problematic frame:
C [CrashDemo.dll+0x11947] _chkstk+0x27
Stack: [0x12b70000,0x12bc0000], sp=0x12bbed98, free space=315k
Native frames: (J=compiled Java code, j=interpreted, Vv=VM code, C=native code)
C [CrashDemo.dll+0x11947] _chkstk+0x27
C [CrashDemo.dll+0x11533] methodA+0x23
C [CrashDemo.dll+0x11613] Java_com_evolutionframework_samples_CrashDemo_crashJVM+0x23
j com.evolutionframework.samples.CrashDemo.crashJVM()V+0
j com.evolutionframework.samples.SampleBadStuffDemo.btnCall_crash_jvm_onClick()V+29

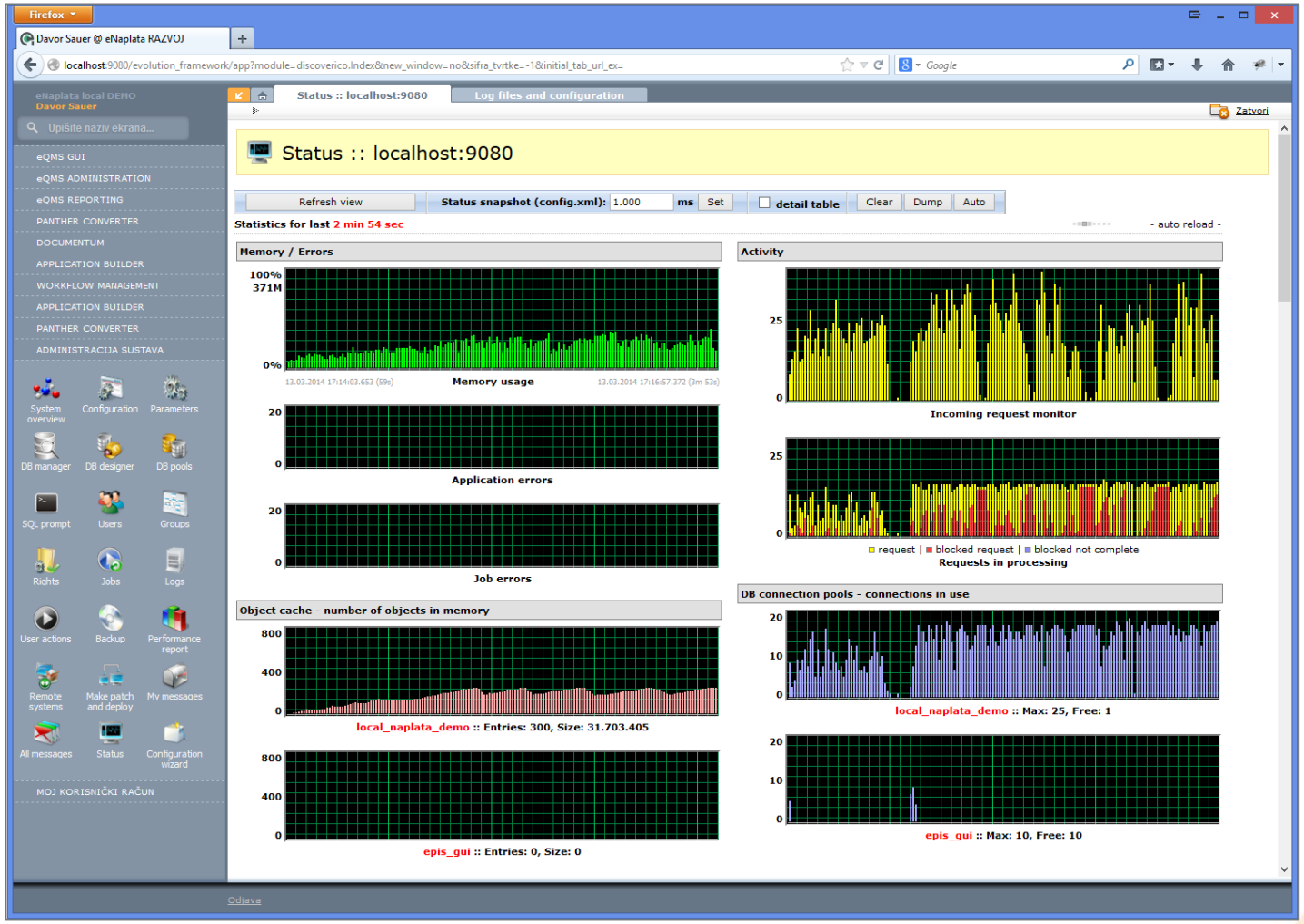
EVOLVA

Analiza memorije

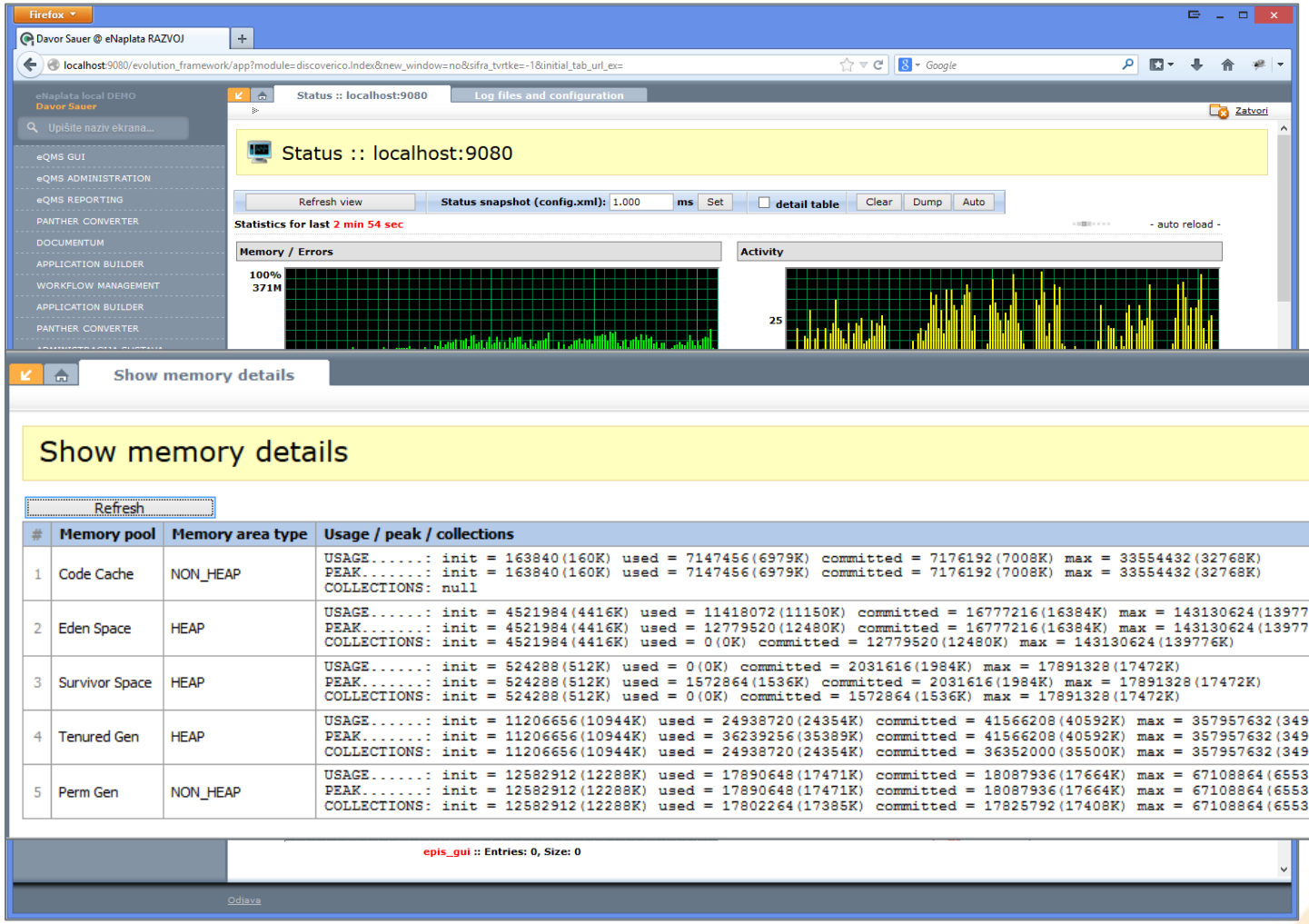
Struktura memorije JVM-a



Potrošnja memorije u usporedbi s ostalim



Potrošnja memorije u usporedbi s ostalim

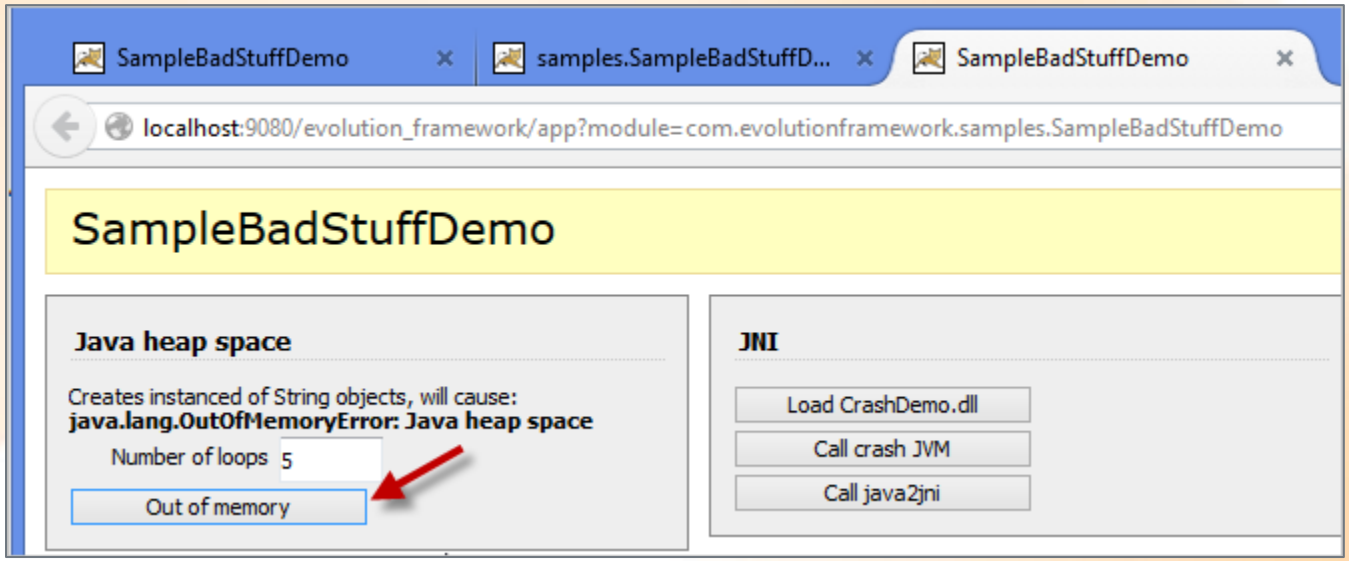


#	Memory pool	Memory area type	Usage / peak / collections
1	Code Cache	NON_HEAP	USAGE.....: init = 163840(160K) used = 7147456(6979K) committed = 7176192(7008K) max = 33554432(32768K) PEAK.....: init = 163840(160K) used = 7147456(6979K) committed = 7176192(7008K) max = 33554432(32768K) COLLECTIONS: null
2	Eden Space	HEAP	USAGE.....: init = 4521984(4416K) used = 11418072(11150K) committed = 16777216(16384K) max = 143130624(139776K) PEAK.....: init = 4521984(4416K) used = 12779520(12480K) committed = 16777216(16384K) max = 143130624(139776K) COLLECTIONS: init = 4521984(4416K) used = 0(0K) committed = 12779520(12480K) max = 143130624(139776K)
3	Survivor Space	HEAP	USAGE.....: init = 524288(512K) used = 0(0K) committed = 2031616(1984K) max = 17891328(17472K) PEAK.....: init = 524288(512K) used = 1572864(1536K) committed = 2031616(1984K) max = 17891328(17472K) COLLECTIONS: init = 524288(512K) used = 0(0K) committed = 1572864(1536K) max = 17891328(17472K)
4	Tenured Gen	HEAP	USAGE.....: init = 11206656(10944K) used = 24938720(24354K) committed = 41566208(40592K) max = 357957632(349568K) PEAK.....: init = 11206656(10944K) used = 36239256(35389K) committed = 41566208(40592K) max = 357957632(349568K) COLLECTIONS: init = 11206656(10944K) used = 24938720(24354K) committed = 36352000(35500K) max = 357957632(349568K)
5	Perm Gen	NON_HEAP	USAGE.....: init = 12582912(12288K) used = 17890648(17471K) committed = 18087936(17664K) max = 67108864(65536K) PEAK.....: init = 12582912(12288K) used = 17890648(17471K) committed = 18087936(17664K) max = 67108864(65536K) COLLECTIONS: init = 12582912(12288K) used = 17802264(17385K) committed = 17825792(17408K) max = 67108864(65536K)

Heap dump - analiza

- Ručno:
  - %JAVA_HOME%\bin\jmap -dump:format=b, file=d:\manual_heap_dump.bin <pid>
- Automatski:
  - -XX:+HeapDumpOnOutOfMemoryError
  - -XX:HeapDumpPath=D:\auto_heap_dump.bin

- Oracle JDK
  - heapdump file: *.bin, *.hprof
  - Eclipse Memory Analyzer Toolkit:
    - http://www.eclipse.org/mat/
  - Visual VM:
    - http://visualvm.java.net/
- IBM WebSphere
  - javacore file - *.phd (portable heap dump)
  - IBM HeapAnalyzer
    - https://www.ibm.com/developerworks/community/groups/service/html/communityview?communityUuid=4544bafe-c7a2-455f-9d43-eb866ea60091

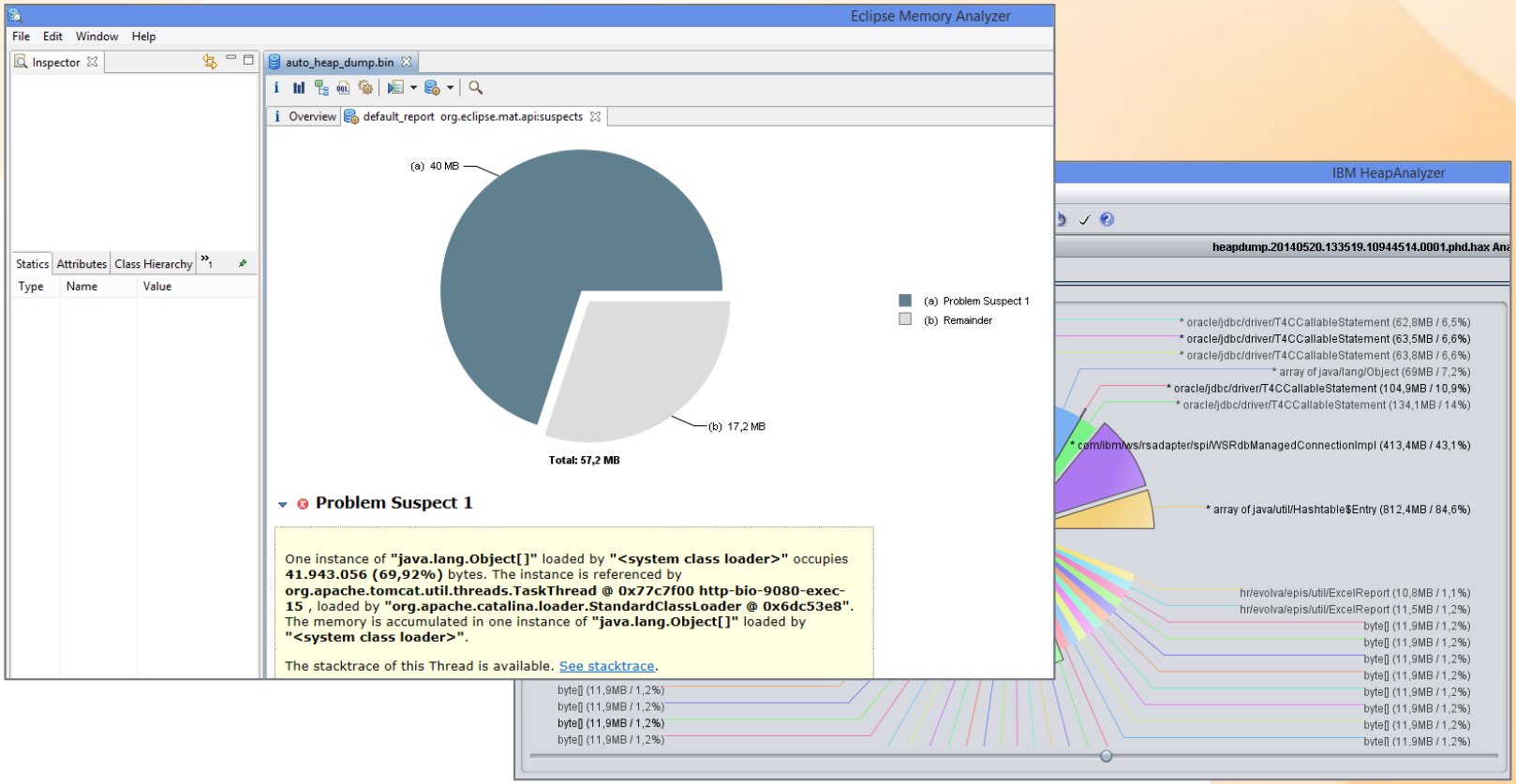



```
2015-04-28 22:35:08.528  REQ00026 0034(  #169 ms)    3 ms      Invoking .btnOutOfMemory_onClick() (start)
2015-04-28 22:35:08.529  REQ00026 0034(  #170 ms)    1 ms        0 ... 18.777.824 of 129.761.280
2015-04-28 22:35:08.725  REQ00026 0034(  #366 ms)  196 ms        5.000.000 ... 41.788.504 of 129.761.280
2015-04-28 22:35:08.900  REQ00026 0034(  #540 ms)  174 ms        10.000.000 ... 39.232.600 of 129.761.280
java.lang.OutOfMemoryError: Java heap space
Dumping heap to d:\_demo\auto_heap_dump.bin ...
Heap dump file created [67683802 bytes in 0.345 secs]
java.lang.OutOfMemoryError: Java heap space
        at java.util.Arrays.copyOf(Arrays.java:2245)
        at java.util.Arrays.copyOf(Arrays.java:2219)
        at java.util.Vector.grow(Vector.java:262)
        at java.util.Vector.ensureCapacityHelper(Vector.java:242)
```

EVOLVA

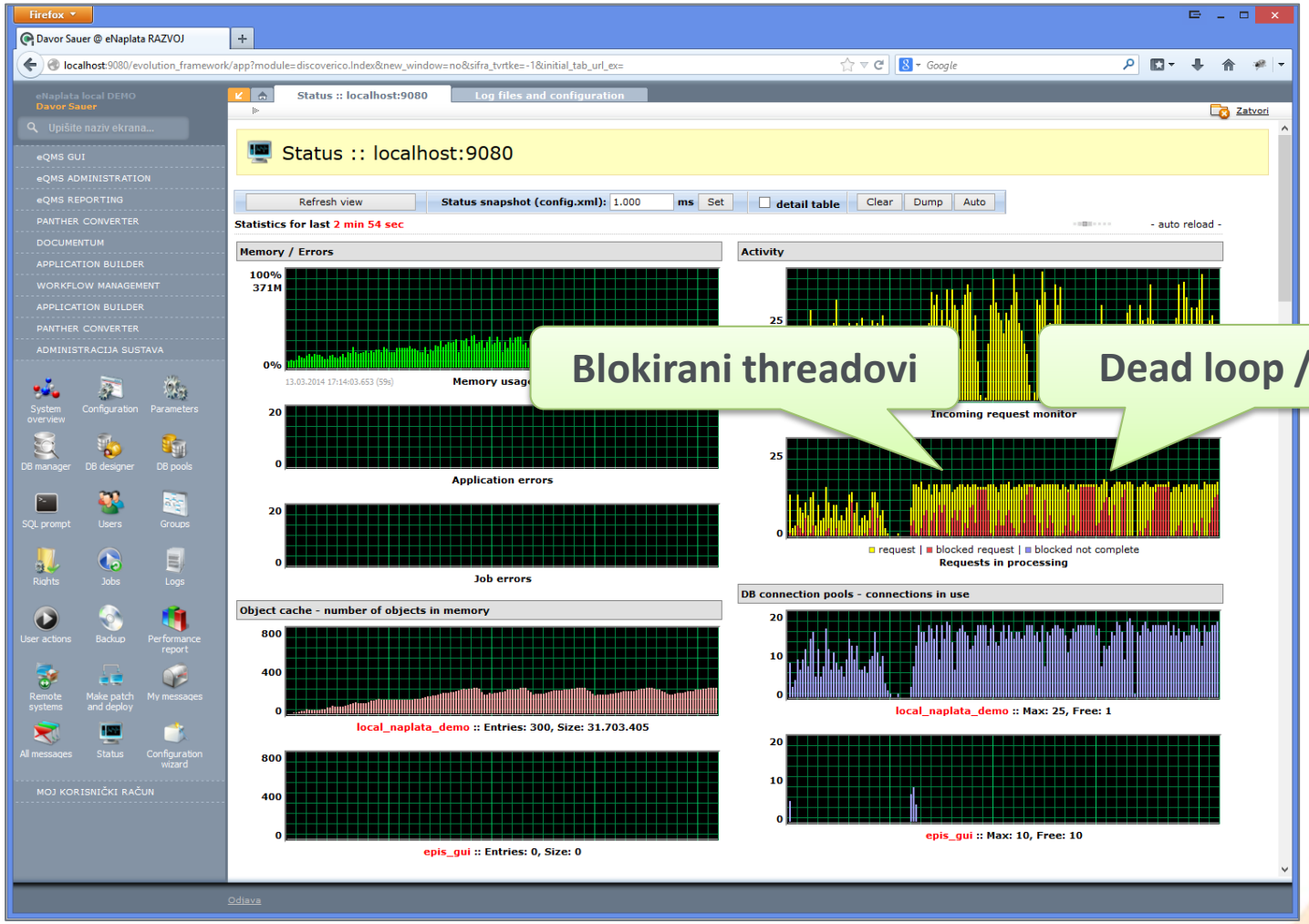
DEMO:

Eclipse MAT, IBM Heap Analyser, Visual VM (runtime)

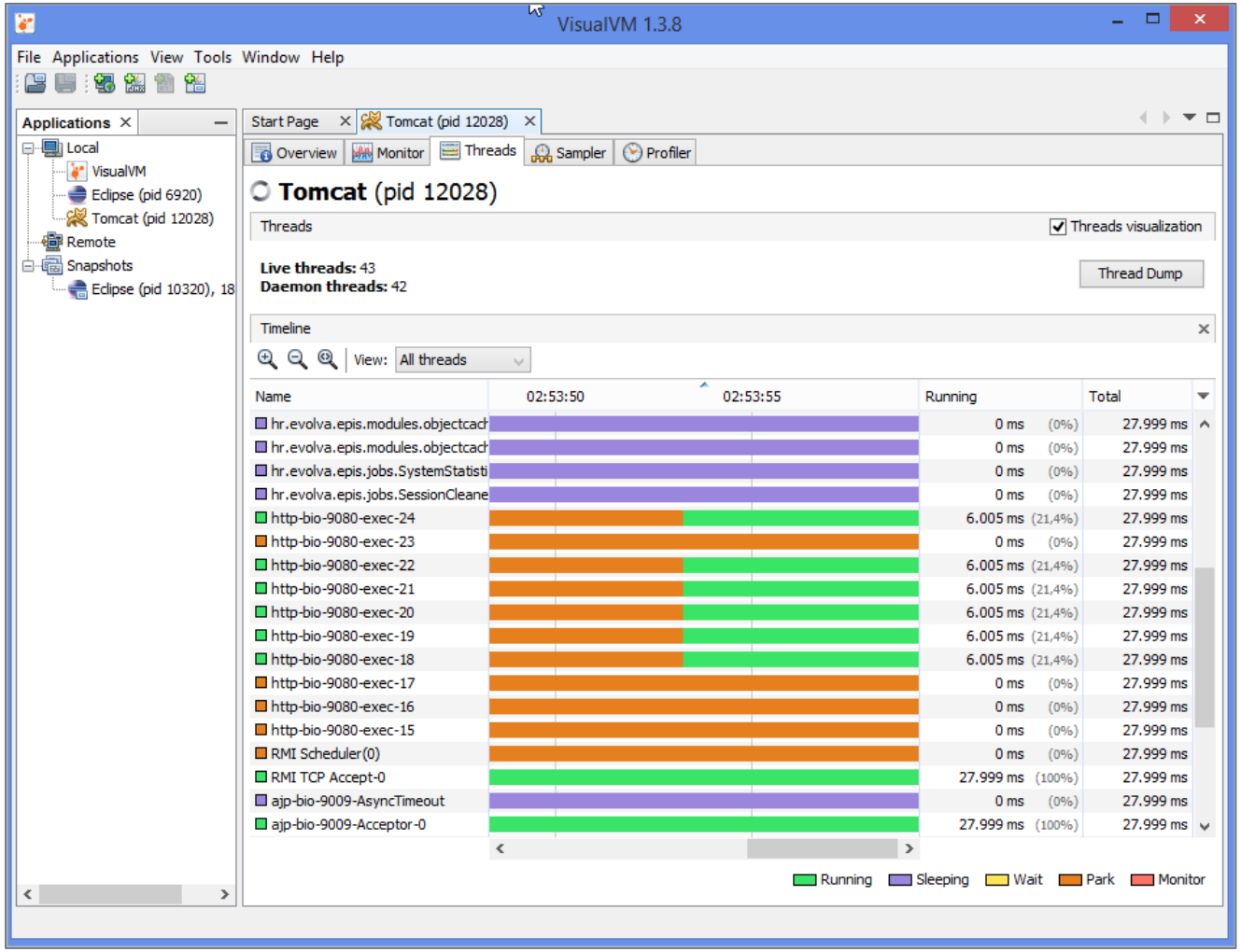


Analiza threadova

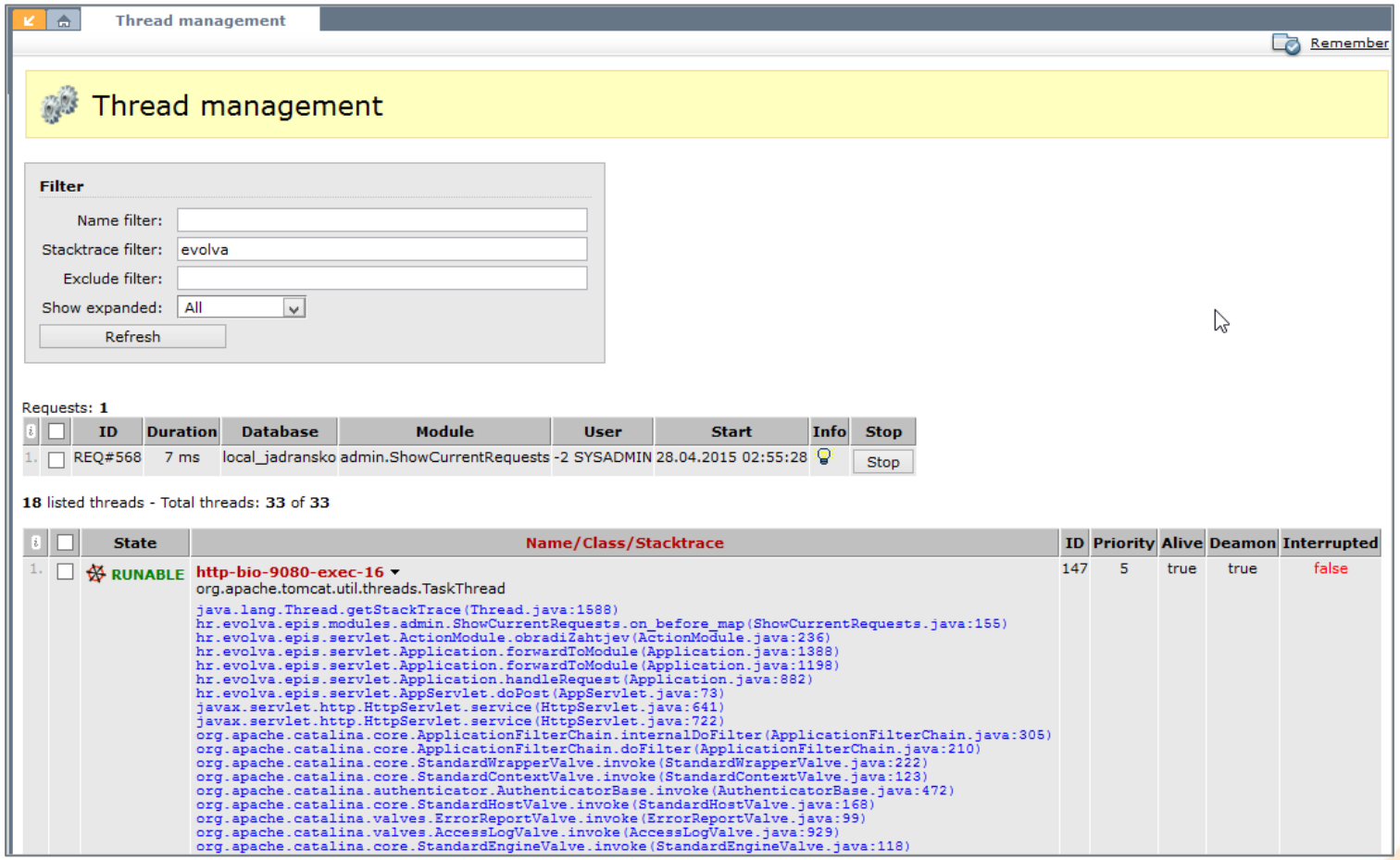
Praćenje threadova – blokade / dead loop



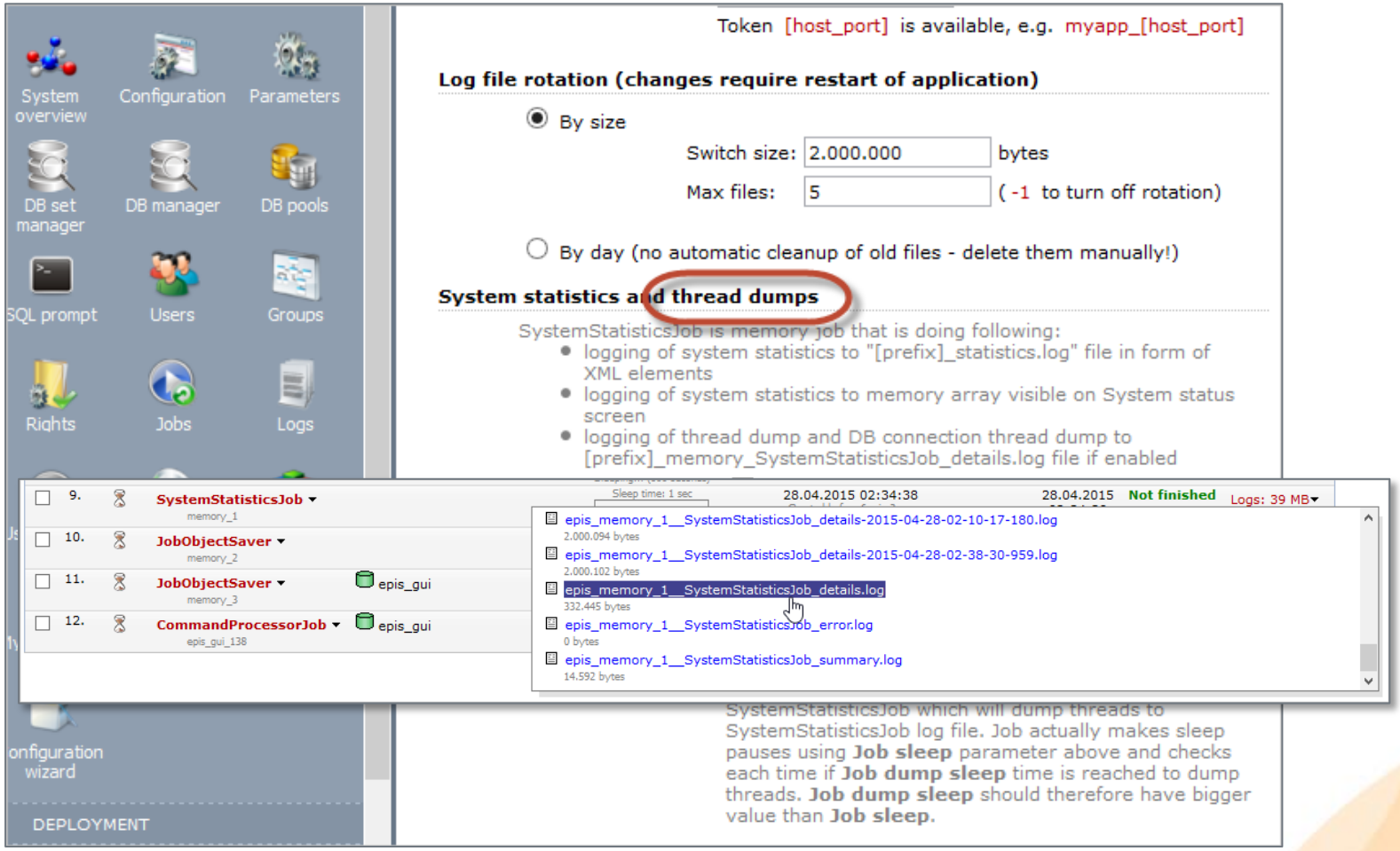
Praćenje threadova - vizualno



Praćenje threadova - pretraživanje

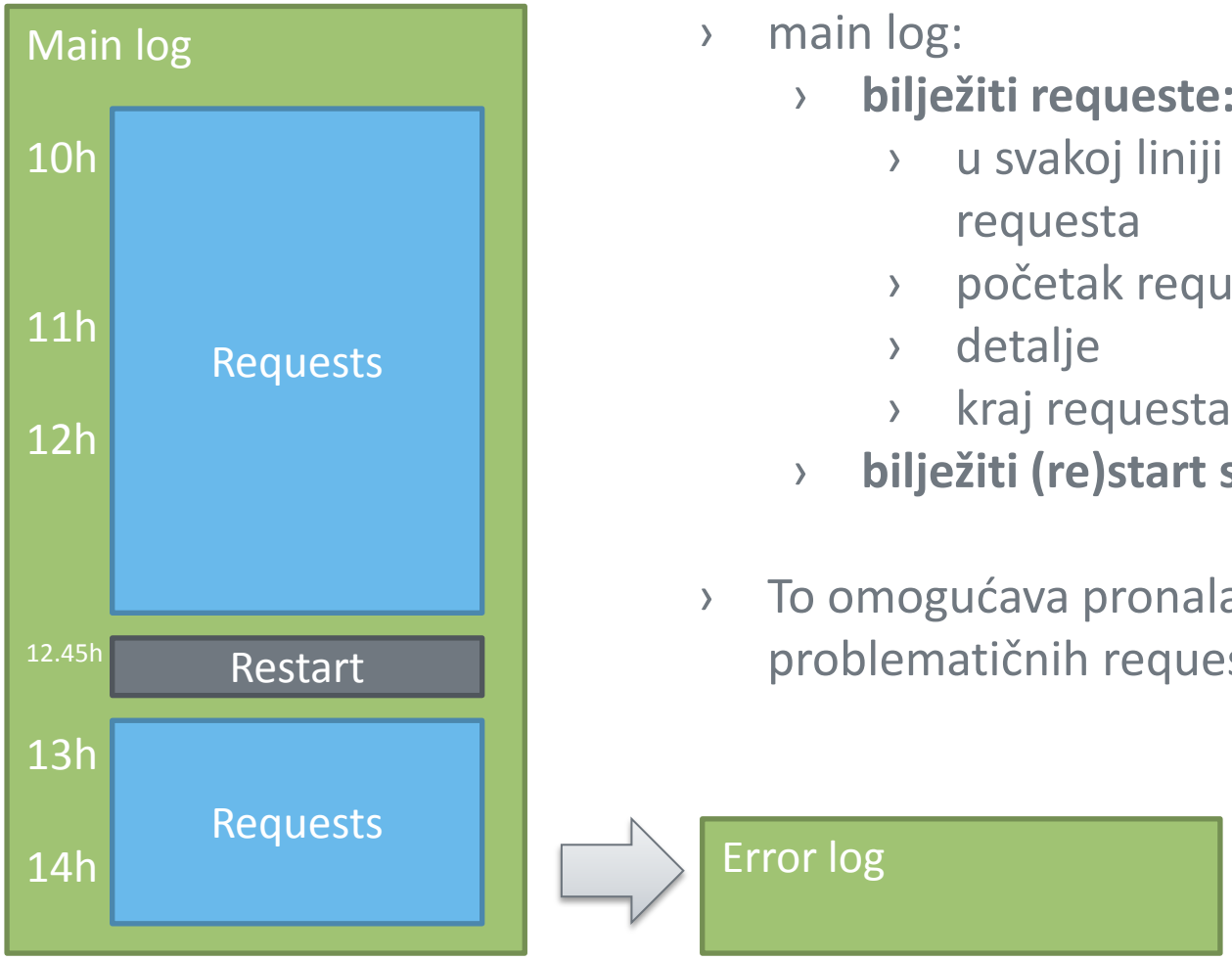


Praćenje threadova - uzorkovanje



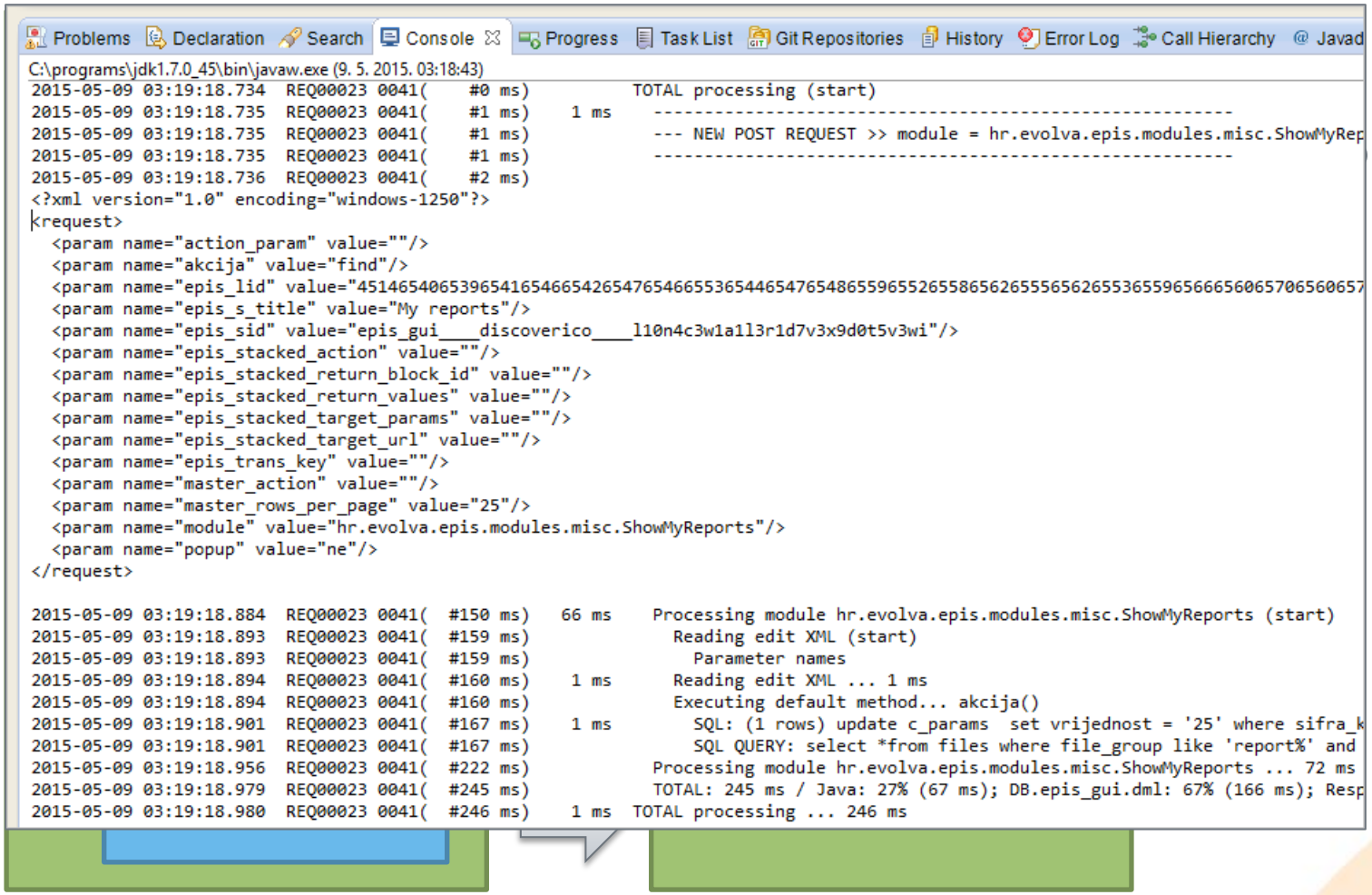
Logging
Orphan requests
Log extractor

Struktura loga

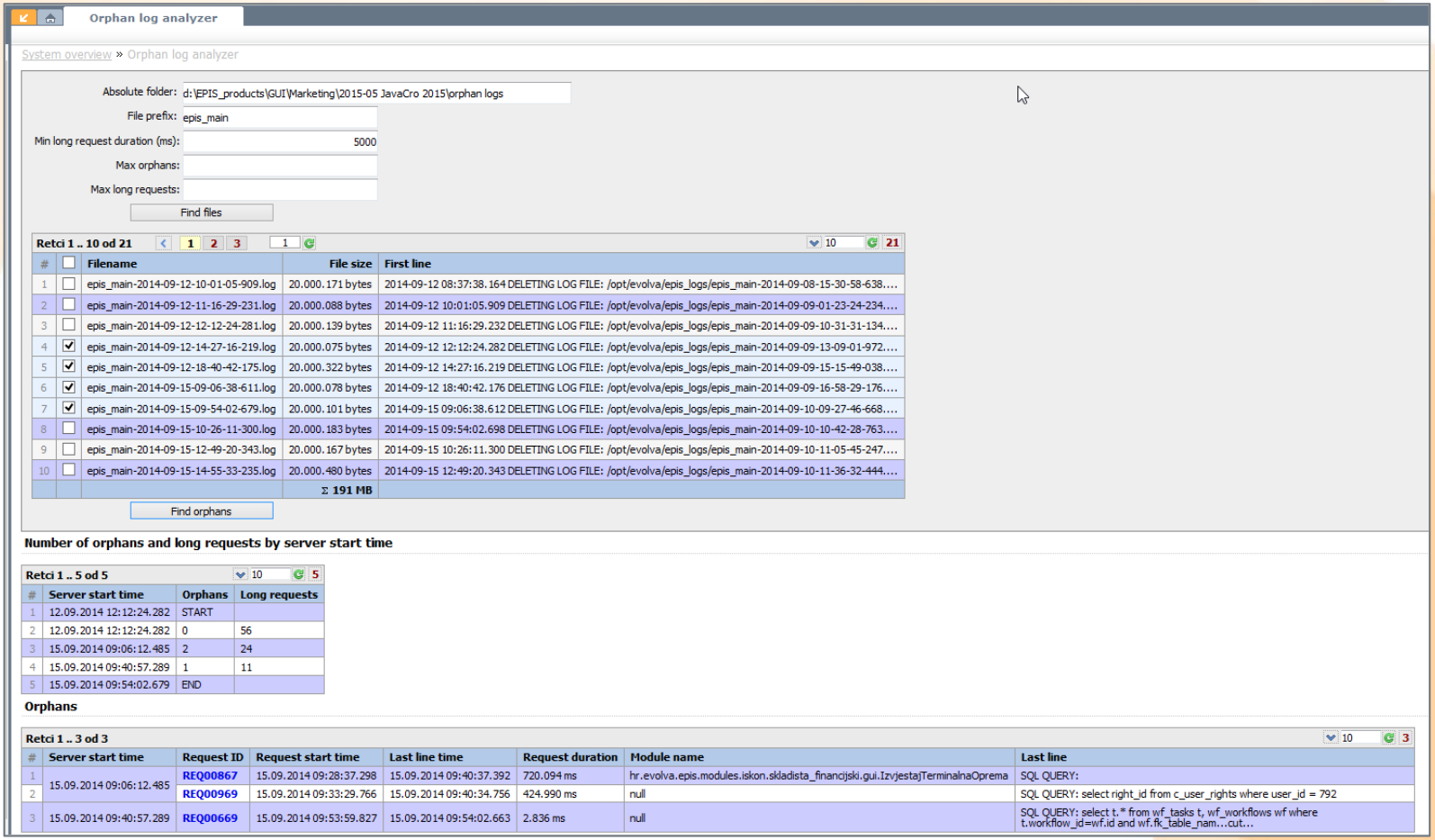


- main log:
  - **bilježiti requeste:**
    - u svakoj liniji identifikator requesta
    - početak requesta
    - detalje
    - kraj requesta
  - **bilježiti (re)start sustava**
- To omogućava pronalazak dugih i problematičnih requesta

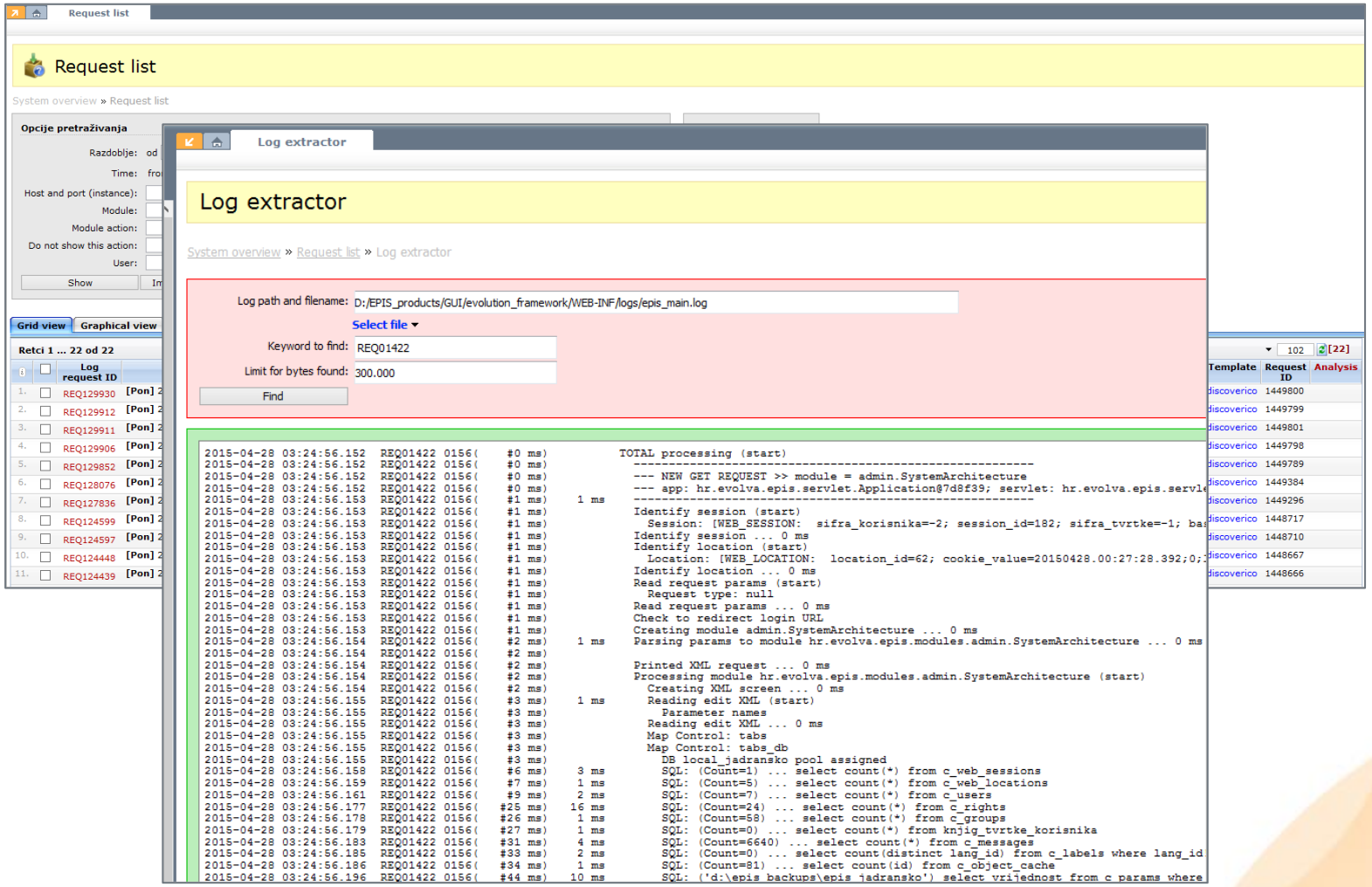
Struktura loga



DEMO:
Orphan log analyser



Request logging + log extractor



Praćenje konekcija na bazi

Dekoracija konekcija

Advanced Oracle features

☑ Decorate connections for v$session

```
SELECT username,
     schemaname,
     osuser,
     machine,
     program,
     client_info,
     module,
     action
FROM v$session b
WHERE type='USER' AND username!='ANONYMOUS'
ORDER BY username, client_info
```

Retci 1 ... 2 od 2 — 25 [2]

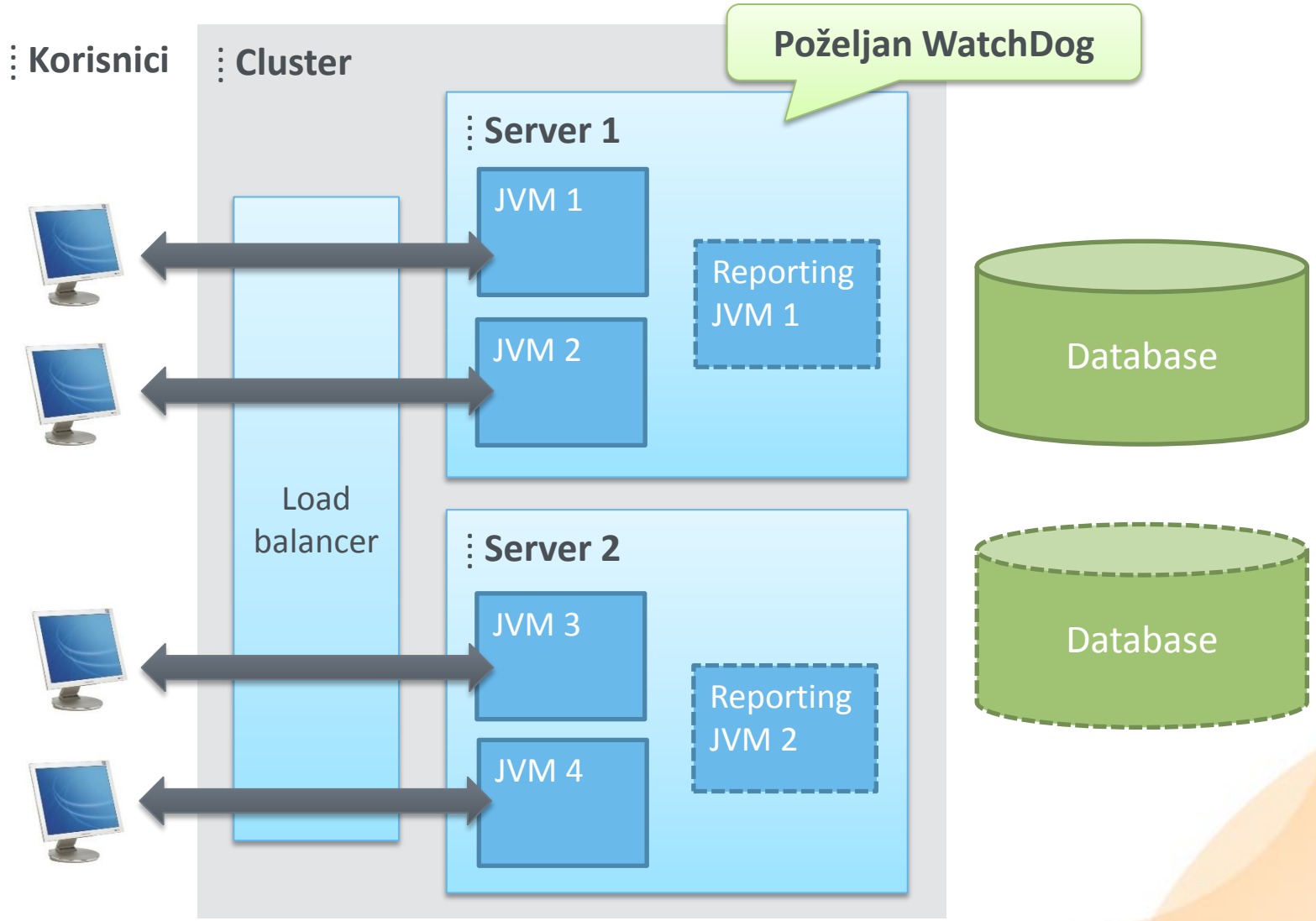
#	USERNAME	SCHEMANAME	OSUSER	MACHINE	PROGRAM	CLIENT_INFO	MODULE	ACTION
1.	EVOLUTION_JADRANSKO	EVOLUTION_JADRANSKO	mtomaskovic	mtomaskovic-dell-17	JDBC Thin Client	localhost:9080 (DB local_jadransko) *FREE* 28.04.2015 02:49:31	CommandProcessorJob	Processing...
2.	EVOLUTION_JADRANSKO	EVOLUTION_JADRANSKO	mtomaskovic	mtomaskovic-dell-17	JDBC Thin Client	localhost:9080 (EPIS DB local_jadransko) 28.04.2015 02:49:34	DBPoolManager	getDatabase

Prevencija: što raditi
da Java aplikacija radi bez zastoja

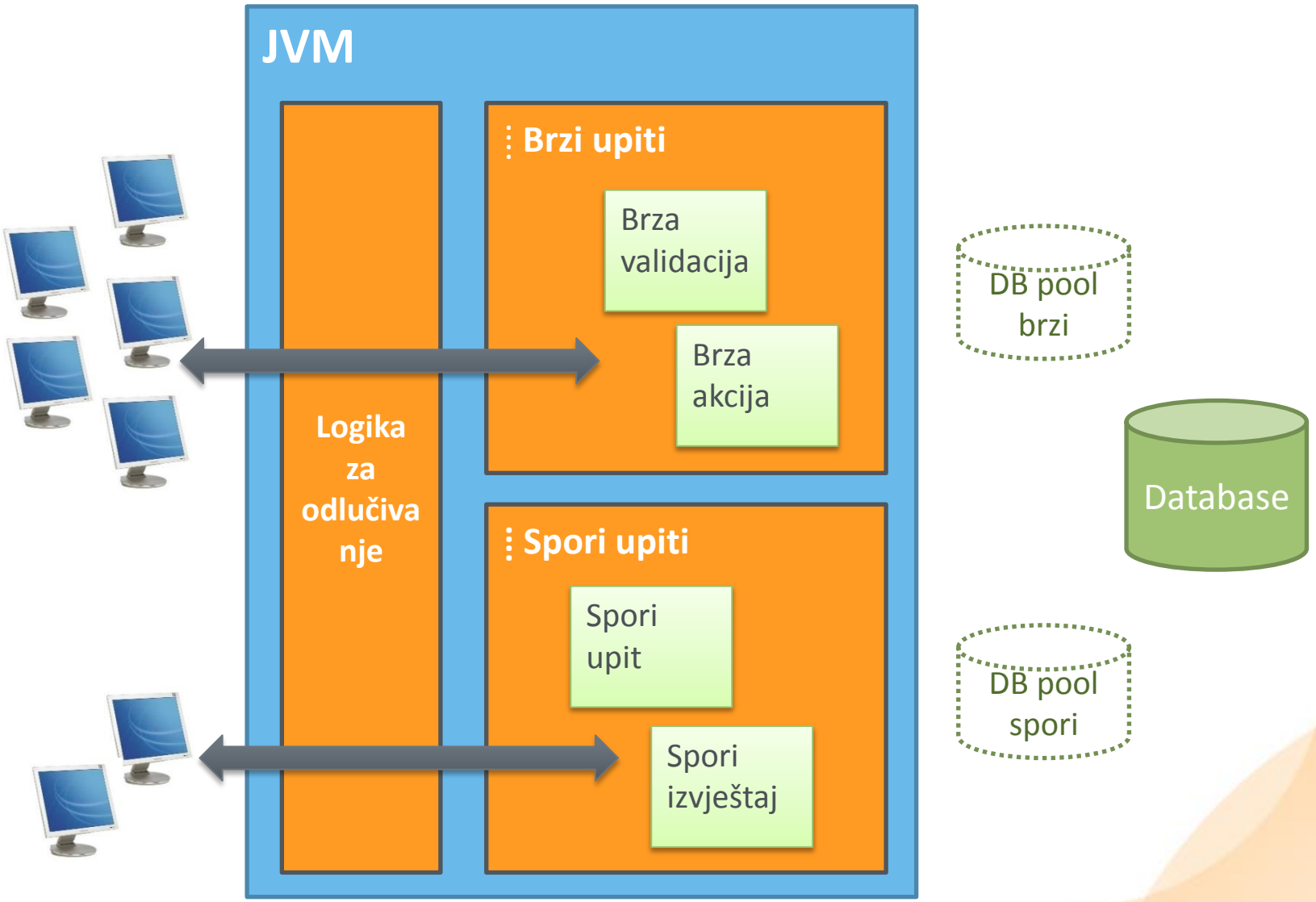
ISO 9001:2008
sustav upravljanja
kvalitetom

EVOLVA

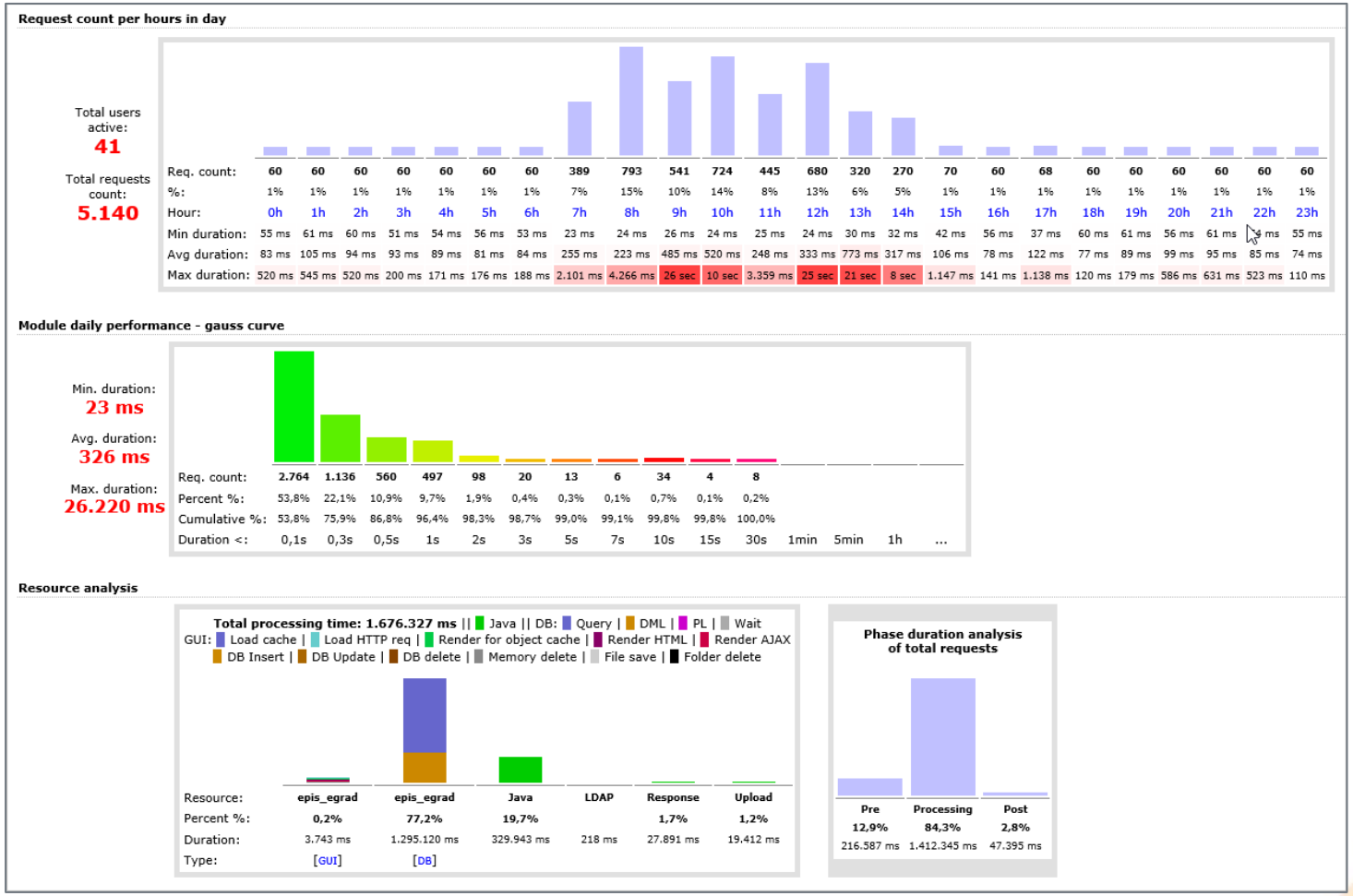
Dobro planirati arhitekturu



Segregacija upita/servisa



Pratiti performanse sustava



Zaključak

Preporuka: sveobuhvatni report o ispadu

- **Pokušati izbjeći**
- **Biti spreman**
- **Generalno**
  - kronologija rada / trenutak ispada
  - ... informacije o okolini koje se ne vide (lako) u logu
- **PAD / ZASTOJ**
  - hs error report (u slučaju JVM crash)
  - heap dump
  - thread dump
    - thread dump uzorci koji su prethodili zastoju
  - request log
    - orphan analysis

Hvala na pažnji

www.evolva.hr

info@evolva.hr

EVOLVA