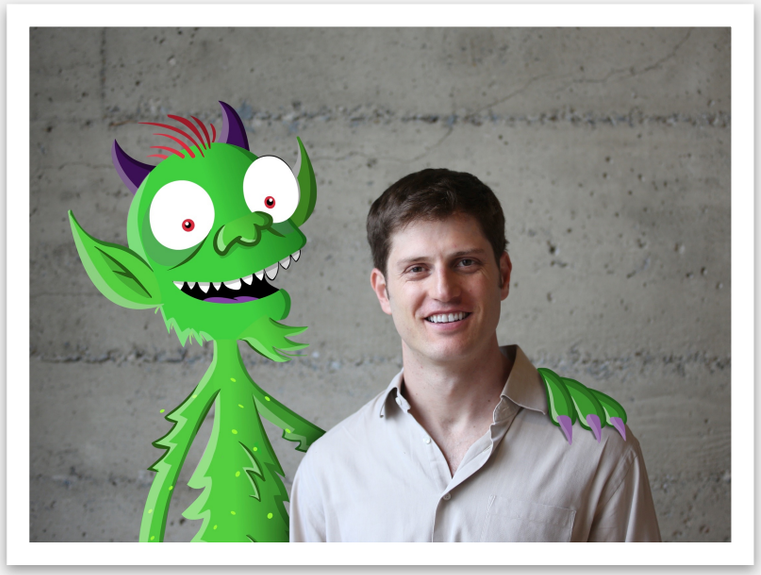
DIY Production Monitoring

About Me



Co-founder – Takipi, JVM Production Debugging.

Director, AutoCAD Web & Mobile.

Software Architect at IAI Aerospace.

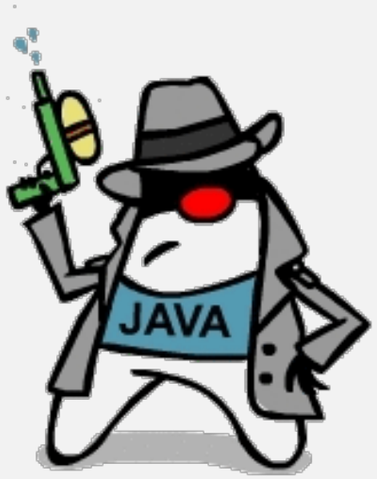
Coding for the past 16 years - C++, Delphi, .NET, Java.

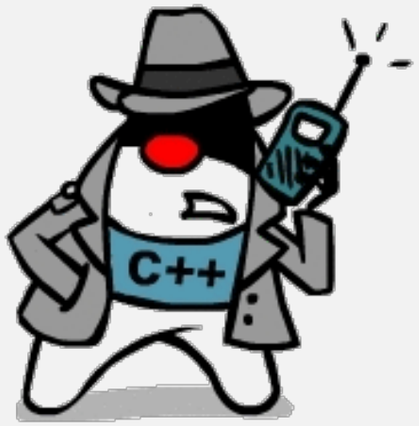
Focus on real-time, scalable systems.

Blogs at blog.takipi.com

Java Agents

- An advanced technique for instrumenting code dynamically.
- The foundation of modern profiling / debugging tools.
- Two types of agents: Java and Native.
- **Pros**: extremely powerful technique to collect state from a live app.
- **Cons**: requires knowledge of creating *verifiable* bytecode.





Agent Types

- Java agents are written in Java. Have access to the *Instrumentation* BCI API.
- Native agents – written in C++.
- Have access to JVMTI – the JVM's low-level set of APIs and capabilities.
  - JIT compilation, Garbage Collection, Monitor acquisition, Exception callbacks, ..
- More complex to write.
- Platform dependent.

Java Profiling Agents

github.com/takipi/profiling-agent

Thread Names

- Thread *name* is a mutable property.
- Can be set to hold transaction specific state.
- Some frameworks (e.g. EJB) don’t like that.
- Can be super helpful when debugging in tandem with **jstack**.

Thread Names (2)

For example:

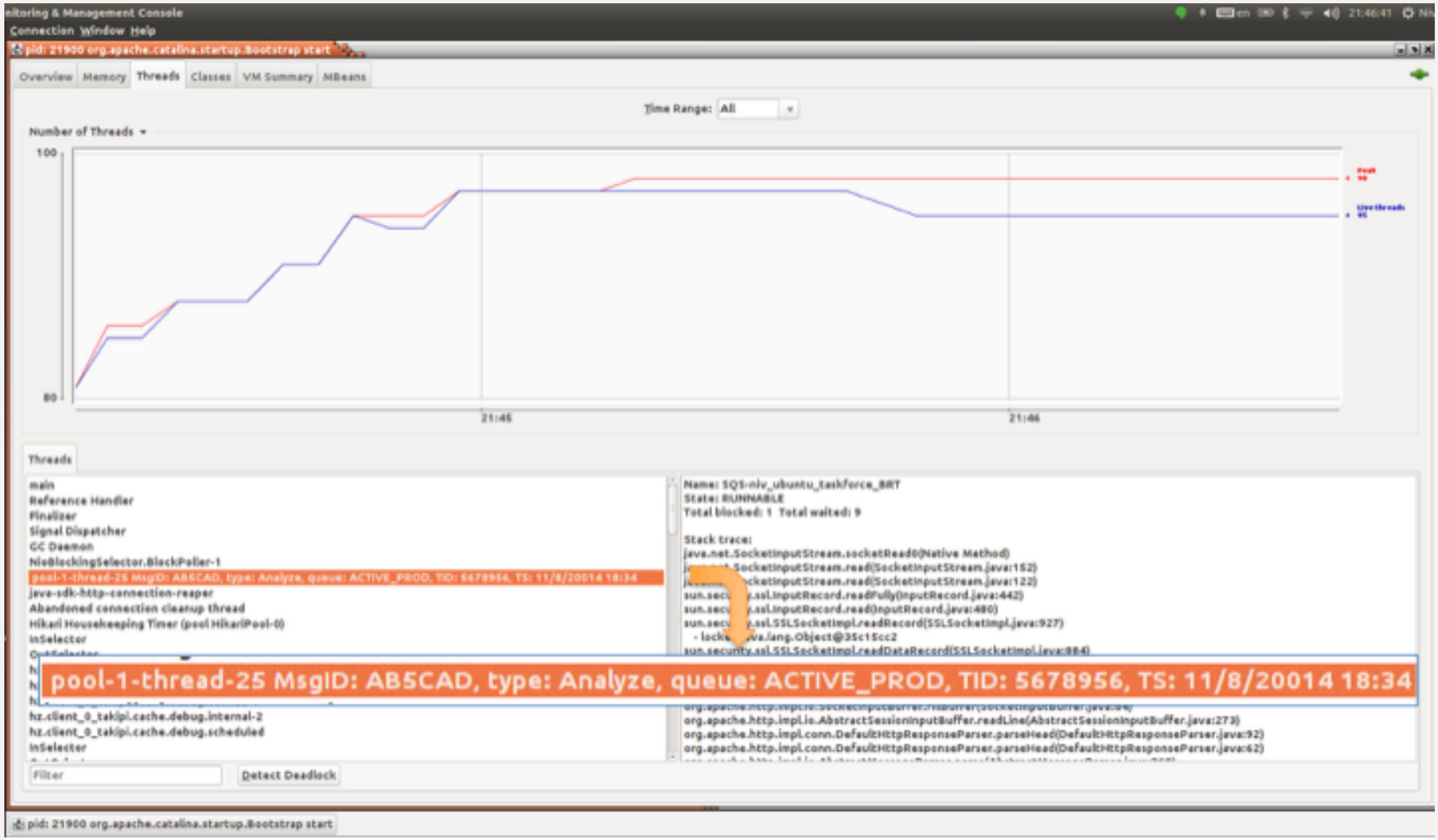
```
Thread.currentThread().setName(
    Context + TID + Params + current Time, ...);
```

Before:

```
"pool-1-thread-1" #17 prio=5 os_prio=31 tid=0x00007f9d620c9800
nid=0x6d03 in Object.wait() [0x000000013ebcc000
```

After:

```
"Queue Processing Thread, MessageID: AB5CAD, type: AnalyzeGraph,
queue: ACTIVE_PROD, Transaction_ID: 5678956, Start Time:
10/8/2014 18:34" #17 prio=5 os_prio=31 tid=0x00007f9d620c9800
nid=0x6d03 in Object.wait() [0x000000013ebcc000]
```


Connection Window Help
pid: 21900 org.apache.catalina.startup.Bootstrap start
Overview
Memory
Threads
Classes
VM Summary
MBeans
Time Range: All
Number of Threads
100
80
21:45
21:46
Threads
main
Reference Handler
Finalizer
Signal Dispatcher
GC Daemon
NioBlockingSelector.BlockPoller-1
pool-1-thread-25 MsgID: AB5CAD, type: Analyze, queue: ACTIVE_PROD, TID: 5678956, TS: 11/8/20014 18:34
java-sdk-http-connection-reaper
Abandoned connection cleanup thread
Hikari Housekeeping Timer (pool HikariPool-0)
InSelector
hz.client_0_takipi.cache.debug.internal-2
hz.client_0_takipi.cache.debug.scheduled
InSelector
Filter
Detect Deadlock
Name: SQS-niv_ubuntu_taskforce_BRT
State: RUNNABLE
Total blocked: 1 Total waited: 9
Stack trace:
java.net.SocketInputStream.socketRead0(Native Method)
org.apache.http.impl.io.AbstractSessionInputBuffer.readLine(AbstractSessionInputBuffer.java:273)
org.apache.http.impl.conn.DefaultHttpResponseParser.parseHead(DefaultHttpResponseParser.java:92)
org.apache.http.impl.conn.DefaultHttpResponseParser.parseHead(DefaultHttpResponseParser.java:62)
pool-1-thread-25 MsgID: AB5CAD, type: Analyze, queue: ACTIVE_PROD, TID: 5678956, TS: 11/8/20014 18:34
pid: 21900 org.apache.catalina.startup.Bootstrap start

Modern Stacks - Java 8

```
Stream lengths = names.stream().map(name -> check(name));

at LmbdaMain.check(LmbdaMain.java:19)
at LmbdaMain.lambda$0(LmbdaMain.java:37)
at LmbdaMain$$Lambda$1/821270929.apply(Unknown Source)
at java.util.stream.ReferencePipeline$3$1.accept(ReferencePipeline.java:193)
at java.util.Spliterators$ArraySpliterator.forEachRemaining(Spliterators.java:948)
at java.util.stream.AbstractPipeline.copyInto(AbstractPipeline.java:512)
at java.util.stream.AbstractPipeline.wrapAndCopyInto(AbstractPipeline.java:502)
at java.util.stream.ReduceOps$ReduceOp.evaluateSequential(ReduceOps.java:708)
at java.util.stream.AbstractPipeline.evaluate(AbstractPipeline.java:234)
at java.util.stream.LongPipeline.reduce(LongPipeline.java:438)
at java.util.stream.LongPipeline.sum(LongPipeline.java:396)
at java.util.stream.ReferencePipeline.count(ReferencePipeline.java:526)
at LmbdaMain.main(LmbdaMain.java:39)
```

Modern Stacks - Scala

```scala
val lengths = names.map(name => check(name.length))
```

```
at Main$.check(Main.scala:6)
at Main$$anonfun$1.apply(Main.scala:12)
at Main$$anonfun$1.apply(Main.scala:12)
at scala.collection.TraversableLike$$anonfun$map$1.apply(TraversableLike.scala:244)
at scala.collection.immutable.List.foreach(List.scala:318)
at scala.collection.TraversableLike$class.map(TraversableLike.scala:244)
at scala.collection.AbstractTraversable.map(Traversable.scala:105)
at Main$delayedInit$body.apply(Main.scala:12)
at scala.Function0$class.apply$mcV$sp(Function0.scala:40)
at scala.runtime.AbstractFunction0.apply$mcV$sp(AbstractFunction0.scala:12)
at scala.App$$anonfun$main$1.apply(App.scala:71)
at scala.App$$anonfun$main$1.apply(App.scala:71)
at scala.collection.immutable.List.foreach(List.scala:318)
at scala.collection.generic.TraversableForwarder$class.foreach(TraversableForwarder.scala:3
at scala.App$class.main(App.scala:71)
at Main$.main(Main.scala:1)
at Main.main(Main.scala)
```

```java
ScriptEngineManager manager = new ScriptEngineManager();
ScriptEngine engine = manager.getEngineByName("nashorn");

String js = "var map = Array.prototype.map \n";
js += "var names = ['Saab', 'Volvo', '']\n";
js += "var a = map.call(names, function(name) { return Java.type(\"preemptiveJstack.ActivateJstack\").check(name) })
js += "print(a)";
engine.eval(js);
```

```
    at preemptiveJstack.ActivateJstack.check(ActivateJstack.java:114)
    at jdk.nashorn.internal.scripts.Script$\^eval\_._L3(<eval>:3)
    at jdk.nashorn.internal.objects.NativeArray$10.forEach(NativeArray.java:1304)
    at jdk.nashorn.internal.runtime.arrays.IteratorAction.apply(IteratorAction.java:124)
    at jdk.nashorn.internal.objects.NativeArray.map(NativeArray.java:1315)
    at jdk.nashorn.internal.runtime.ScriptFunctionData.invoke(ScriptFunctionData.java:522)
    at jdk.nashorn.internal.runtime.ScriptFunction.invoke(ScriptFunction.java:206)
    at jdk.nashorn.internal.runtime.ScriptRuntime.apply(ScriptRuntime.java:378)
    at jdk.nashorn.internal.objects.NativeFunction.call(NativeFunction.java:161)
    at jdk.nashorn.internal.scripts.Script$\^eval\_.runScript(<eval>:3)
    at jdk.nashorn.internal.runtime.ScriptFunctionData.invoke(ScriptFunctionData.java:498)
    at jdk.nashorn.internal.runtime.ScriptFunction.invoke(ScriptFunction.java:206)
    at jdk.nashorn.internal.runtime.ScriptRuntime.apply(ScriptRuntime.java:378)
    at jdk.nashorn.api.scripting.NashornScriptEngine.evalImpl(NashornScriptEngine.java:546)
    at jdk.nashorn.api.scripting.NashornScriptEngine.evalImpl(NashornScriptEngine.java:528)
    at jdk.nashorn.api.scripting.NashornScriptEngine.evalImpl(NashornScriptEngine.java:524)
    at jdk.nashorn.api.scripting.NashornScriptEngine.eval(NashornScriptEngine.java:194)
    at javax.script.AbstractScriptEngine.eval(AbstractScriptEngine.java:264)
    at preemptiveJstack.ActivateJstack.main(ActivateJstack.java:128)
```