



Evaluating Java Mission Control

Lightning Talk

Scott Hurley

16/08/2018

Introduction

About Me

From Paisley, Scotland
Study Computer Science
at the University of
Strathclyde

...



Scott Hurley

Objectives

Evaluate Oracle's Java Mission Control and Java Flight Recorder in the context of Databases Applications Services using these technologies:

Oracle WebLogic

Apache Tomcat

CERN IT-DB Services consists of 250 Java Servers

Java Performance App

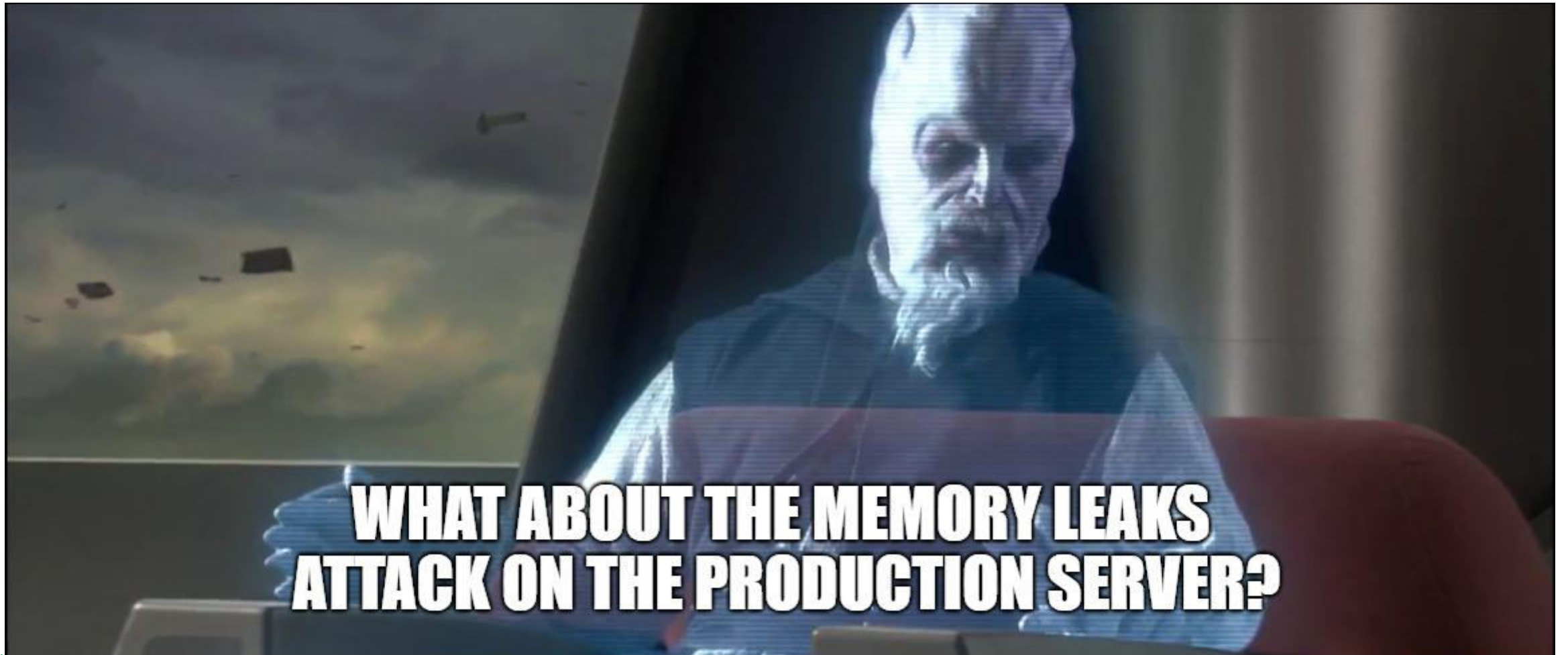
Uses the Spring framework, a MySQL database and Maven due to popularity within the CERN development community

Demonstrates:

- Deadlock
- Gridlock
- Excessive Garbage Collection
- Effects of varying thread pool size
- Memory leaks

Deployed on tomcat

Why Do We Need Monitoring Tools?



Why Do We Need Monitoring Tools?



Java Mission Control

JDK profiling and diagnostics tools platform

Included by default since JDK 7

Main plug-ins are JMX and JFR

Can install other plug ins - experimental

Used exclusively through its UI

Requires a commercial licence for remote monitoring

JMC Features

JMX -

Real-time monitoring

Can connect to remote JVMs and execute diagnostic commands

Tracked stats include CPU and memory usage

JFR -

Runtime data saved for later review

Low performance overhead

Tracked stats included GC pauses, time spent in methods

Triggers -

Actions set to occur when certain conditions are met

e.g. send an email to the system administrator when too much memory is used by the JVM

Used only through UI, not automatable

JMC Pros & Cons

Pros - JMC UI is intuitive and informative

Statistics provided by JMX and JFC are very useful for identifying issues

JFR has low performance overhead

Recordings are highly customisable

Triggers are extremely powerful

Cons - JMC can only be used through its UI – no plans to address this

This means its actions cannot be automated by Puppet

Especially unfortunate considering usefulness of triggers

If the JVM being monitored becomes unresponsive, so does JMX

If the monitored JVM exits unexpectedly, in-progress recordings will be lost

What I've learned

How to use Spring and Maven

How to deploy apps using tomcat

How to write useful documentation

I hate the weather here

Deliverables

Create a Java application to demonstrate common performance issues:
https://gitlab.cern.ch/db/java_perf_issues.git

Write documentation on how to install and use JMC and JFR:
<https://twiki.cern.ch/twiki/bin/view/DB/Private/JavaMissionControl>

Report evaluating JMC and JFR

Thanks to: CERN
CERN openlab
CERN openlab Summer Student program staff
CERN summer students
IT-DB-IMS team
Luis Rodriguez Fernandez



Thank You

scott.james.hurley.97@gmail.com