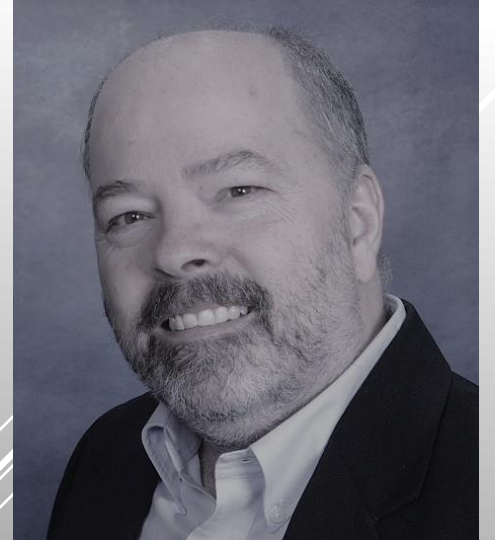


CONSIDERING MONITORING SOLUTIONS FOR CF AND LUCEE



Charlie Arehart

Independent Consultant/Server Troubleshooter

CArehart.org (@carehart)

Updated Apr 25, 2022



A large fire burning in a forest at night, with tall utility poles in the foreground. The fire is intense and bright orange, with thick black smoke rising into the dark sky. The utility poles are silhouetted against the fire, and some have blue lights. The ground in the foreground is dark and appears to be covered in ash or charred debris.

A DAY IN THE LIFE



WHO MIGHT YOU BE?



CHALLENGE: CHOOSING THE RIGHT TOOLS

WHAT'S NOT THE RIGHT TOOL?



WHO AM I?





COMMON PERF PROBLEMS OBSERVED



Code
inefficiency

SQL
inefficiency

GC
inefficiency

Need of JVM
"tuning"

CF itself is
slow

CF/Lucee
needs to be
restarted

And more

CAUSES THAT MANY SUSPECT



Config issues

External
calls
hanging

DB
connection
pool hanging

Unexpected
traffic
volume/
nature

Sudden,
excessive
GCs

Web server
connector
issues

CF engine /
JVM bugs

And more

LESS-EXPECTED CAUSES





LESS-NOTICED PROBLEMS



DBs called

APIs called

Other
processes on
server

Disk or
Network
Issues

Problem in
VM host,
other guests

And more

CAUSES OUTSIDE OF CF/LUCEE



A photograph of a server room with two technicians. One technician is standing and working on a server rack, while the other is kneeling and looking at a document. The room is filled with rows of server racks, and the image is overlaid with a dark, semi-transparent filter. The text "THE CHALLENGE OF SCALE" is centered in white, bold, sans-serif font.

THE CHALLENGE OF SCALE

Charlie Arehart (CArehart.org, @carehart)

This Photo by Unknown Author is licensed under [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/)

CF/Lucee Troubleshooting Features

CF/Lucee Monitoring Tools

Java Troubleshooting Approaches

Application Performance Monitors (APMs)

Other System Monitors

FOUR CLASSES OF TOOLS TO HELP





CF/LUCEE TROUBLESHOOTING FEATURES



CF Server
Monitor
(CF8-2016)

SeeFusion

CF PMT
(CF2018 & up)

FusionReactor

CF/LUCEE MONITORING TOOLS



Built-in or freely obtained GUI tools

JMX

Tomcat Manager

3rd party tools

Java APMs

THEN THERE ARE JAVA TOOLS...



Stack
Traces

Thread
dumps

Profiling

Heap
dumps

JAVA TROUBLESHOOTING APPROACHES





BUILT-IN JAVA TOOLS



- ▶ Again, jstack, visualvm, etc.
- ▶ FR, SeeFusion, CF PMT
- ▶ Some APMs (requires reaching into running JVM)

STACK TRACING TOOLS



- ▶ Taking thread dumps
 - ▶ Again: VisualVM, Java Mission Control, Tomcat Manager, jcmd, jstack
 - ▶ And within most APM tools (as well as FR, SF, CFM/PMT), and other ways
- ▶ Analyzing thread dumps (free, online services/tools)
 - ▶ *fastthread.io*
 - ▶ *jstack.review*
 - ▶ *spotify.github.io/threaddump-analyzer/*
 - ▶ *github.com/irockel/tda*
- ▶ More help: *baeldung.com/java-analyze-thread-dumps*

THREAD DUMP TOOLS



- ▶ These monitor time spent within or across threads, by java method
- ▶ Available in VisualVM and Java Mission Control, as well as:
 - ▶ Some APMs (as well as FR, SF, CFSM/PMT)
 - ▶ Yourkit: yourkit.com
 - ▶ Jprofiler: ej-technologies.com
 - ▶ Xrebel: xrebel.com
 - ▶ Netbeans: netbeans.apache.org
- ▶ More help: baeldung.com/java-profilers

PROFILING TOOLS



- ▶ Taking heap dumps
 - ▶ VisualVM, Java Mission Control, Tomcat Manager
 - ▶ jcmd, jmap, jmx, HeapDumpOnOutOfMemoryError
 - ▶ And most APM tools (including FR, PMT)
 - ▶ Lucee also has a Heap Dump extension
- ▶ Analyzing them
 - ▶ Eclipse memory analyzer tool (eclipse.org/mat)
 - ▶ heaphero.io
 - ▶ jxray.com
- ▶ More info: dzone.com/articles/how-to-capture-java-heap-dumps-7-options

MEMORY/HEAP ANALYSIS TOOLS



- ▶ Forcing GCs: `jvisualVM`, `jcmt`, most APM tools (and FR, SF, CFM/PMT)
- ▶ Obtaining GC logs: `-XX:+PrintGCDetails` and related args
- ▶ Tools to analyze GC logs
 - ▶ `gceasy.io`
 - ▶ `gcplot.com`
 - ▶ `github.com/chewiebug/GCViewer`
 - ▶ *And others*

GC TOOLS



- ▶ jmx: java mgt extensions
 - ▶ Many java apps, including Tomcat, expose metrics via jmx
- ▶ Tools to view them include:
 - ▶ jvisualVM, Tomcat Manager
 - ▶ jmc, jconsole
 - ▶ FusionReactor
 - ▶ Some APM tools
 - ▶ *jolokia.org*

JMX TOOLS



- ▶ Appdynamics
- ▶ Dynatrace
- ▶ NewRelic
- ▶ Datadog
- ▶ Glowroot
- ▶ Solarwinds AppOptics
- ▶ Sumo Logic
- ▶ Stackify (prefix and retrace)
- ▶ JavaMelody
- ▶ Perfino
- ▶ Sematext
- ▶ MoSKito
- ▶ Scouter
- ▶ and still others!

JAVA APMS: MANY, MANY



- ▶ yCrash (Java Root Cause Analyzer)
- ▶ Overops (app observability/root cause analysis)
- ▶ Akamas (autonomous JVM performance optimization)

SOME NON-APM JAVA TOOLS OF NOTE



- ▶ FR and some APMs export their data to cloud monitoring/telemetry
 - ▶ Vital when managing many instances, containers
- ▶ FR/SF/PMT as well as some APM's also offer:
 - ▶ Alerting about trouble
 - ▶ Error tracking
- ▶ And FR/PMT and some APM's also offer:
 - ▶ Tracking end-user response time
 - ▶ Tracking sessions
 - ▶ Reporting

OTHER FEATURES TO WATCH FOR



- ▶ Use to track baselines, then changes over time, such as due to:
 - ▶ Code changes
 - ▶ CF/Lucee admin
 - ▶ JVM config changes
 - ▶ Changing volume/nature of traffic/load

OTHER BENEFITS OF MONITORING





FINALLY, IMPORTANT TO MONITOR OTHER
PARTS OF ENVIRONMENT
(CF411.COM/MON)



- ▶ We've seen many problems, approaches, tools
- ▶ Still, even basic tools (task mgr/top, cfstat/perfmon, etc.) better than nothing
- ▶ But often need to see inside CF/CFML processing: requests, queries, more
- ▶ And Java tools can help (or not), but you maybe learned of new ones
- ▶ Get and use SOME diagnostics. Don't just "restart" CF/Lucee! 😊
- ▶ For more, on this topic or other troubleshooting:
 - ▶ charlie@carehart.org
 - ▶ @carehart (on twitter, linkedin, github, facebook, etc.)

CONCLUSION

