9 Free Java Troubleshooting Tools

A guide for Application Support Engineers and Developers

Karun Subramanian

www.karunsubramanian.com

Contents

INTRODUCTION

How can you resolve any type of Application issues, FAST?

Troubleshooting Java, especially in a complex environment can be tedious. Java Enterprise Edition (JEE) literally gave birth to a new career with various titles – Application Administrator, Middleware Administrator, Application Support engineer and so forth. The core responsibility of a professional in this career is to ensure that the Application Serving environment is running at its peak performance, securely. And this is no simple task. The types of issues one can face in this profession can be numerous and while some problems take minutes to solve others take days or weeks to solve. Luckily there are lots of freely available tools that make this job easier. In this ebook, we will look at 9 tools that one can use to successfully maintain a JEE environment. Note that this list is only a small subset of tons of tools out there. You are free to take whatever works for you and leave the rest.

0. Verbose GC logs
1. IBM Pattern Modeling and Analysis Tool for Java Garbage Collector
2. IBM Thread and Monitor Dump Analyzer for Java
3. Eclipse Memory Analyzer
4. JConsole
5. VisualVM
6. Jmeter
7. Wireshark
8. Appdynamics Lite
9. NewRelic Lite

CHAPTER 0:

VERBOSE GC LOGS

A trustworthy source for analyzing JVM memory issues

Yes, I know, verbose GC log is not really a 'tool' but I have listed this as a tool because of the immense value it provides in troubleshooting Memory related issues.

How does it help you in troubleshooting?

a. Reveals the following critical information
   - Garbage Collection frequency
   - Garbage collection duration
   - Heap before/after GC
   - Shows memory allocation in various Pools *

* *Varies depending on the JVM vendor*

How can you obtain this tool?

Simply add the following parameter to the Java command line that starts the Application Server and you are good to go. The file where verbose GC logs will go depends on the Application Server. For example, with IBM WebSphere Application Server, the verbose GC logs generally go to native_stderr.log

Parameter to add:

-verbose:gc

You can add the following parameters to provide even more details:

-XX:+PrintGCDetails -XX:+PrintGCTimeStamps

Where can you find more information about this tool?

http://www.oracle.com/technetwork/java/javase/gc-tuning-6-140523.html

http://www-01.ibm.com/support/docview.wss?uid=swg21114927

https://docs.oracle.com/cd/E15289_01/doc.40/e15061/verbose.htm

CHAPTER 1:

IBM Pattern Modeling and Analysis Tool for Java Garbage Collector
Garbage Collection analysis, the easy way

This is a very good tool for visually analyze Verbose GC logs. It is time saving since you don't have to eyeball all the gory details in the verbose GC logs.

How does this tool help in troubleshooting?

a. Plots GC stats in graphs for quick analysis
b. Reveals 'Out of Memory' errors
c. Attempts to recommend solutions to problems it finds in the verbose GC logs. For example, under certain conditions, it recommends increasing the maximum heap size.
d. Allows you to quickly navigate to various verbose GC log files.
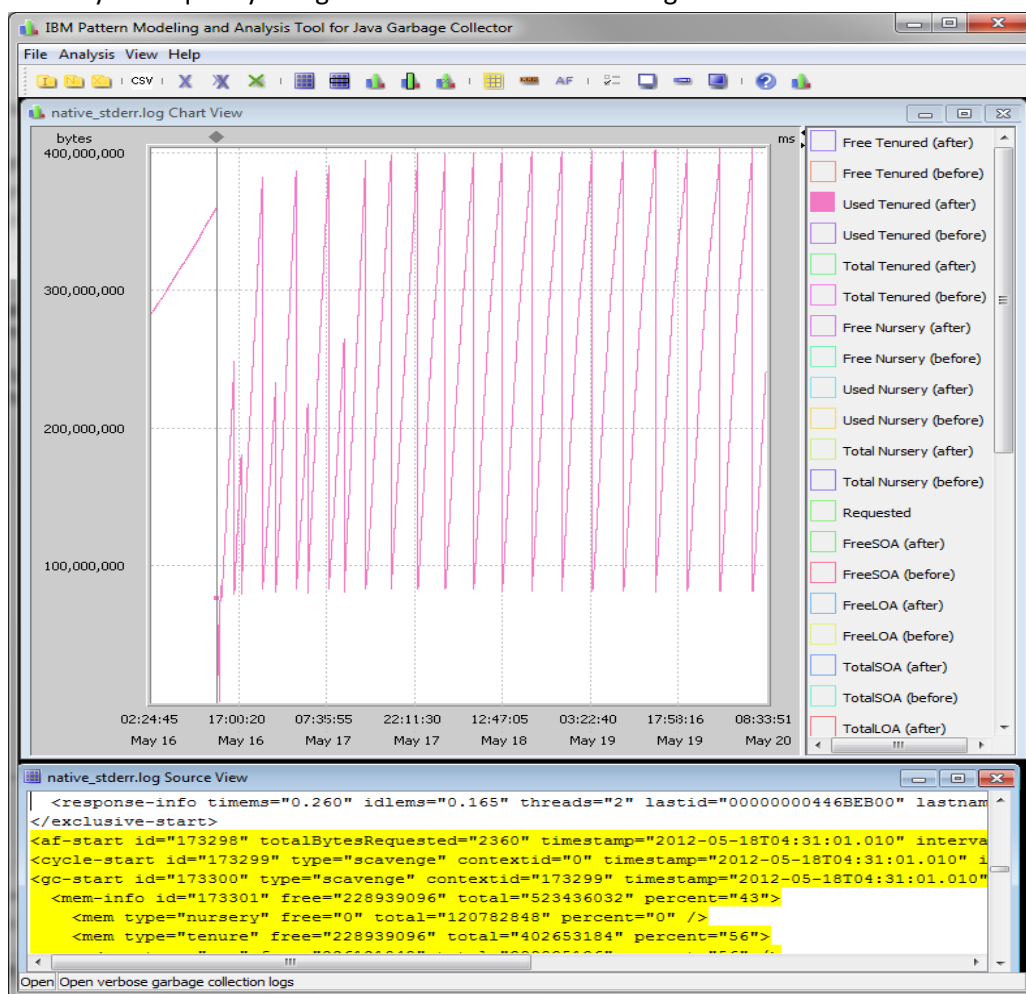

Image Source: IBM

How can you obtain this tool?

Free download:

https://www.ibm.com/developerworks/community/groups/service/html/communityview?communityUuid=22d56091-3a7b-4497-b36e-634b51838e11

Where can you find more information about this tool?

https://www.ibm.com/developerworks/community/groups/service/html/communityview?communityUuid=22d56091-3a7b-4497-b36e-634b51838e11

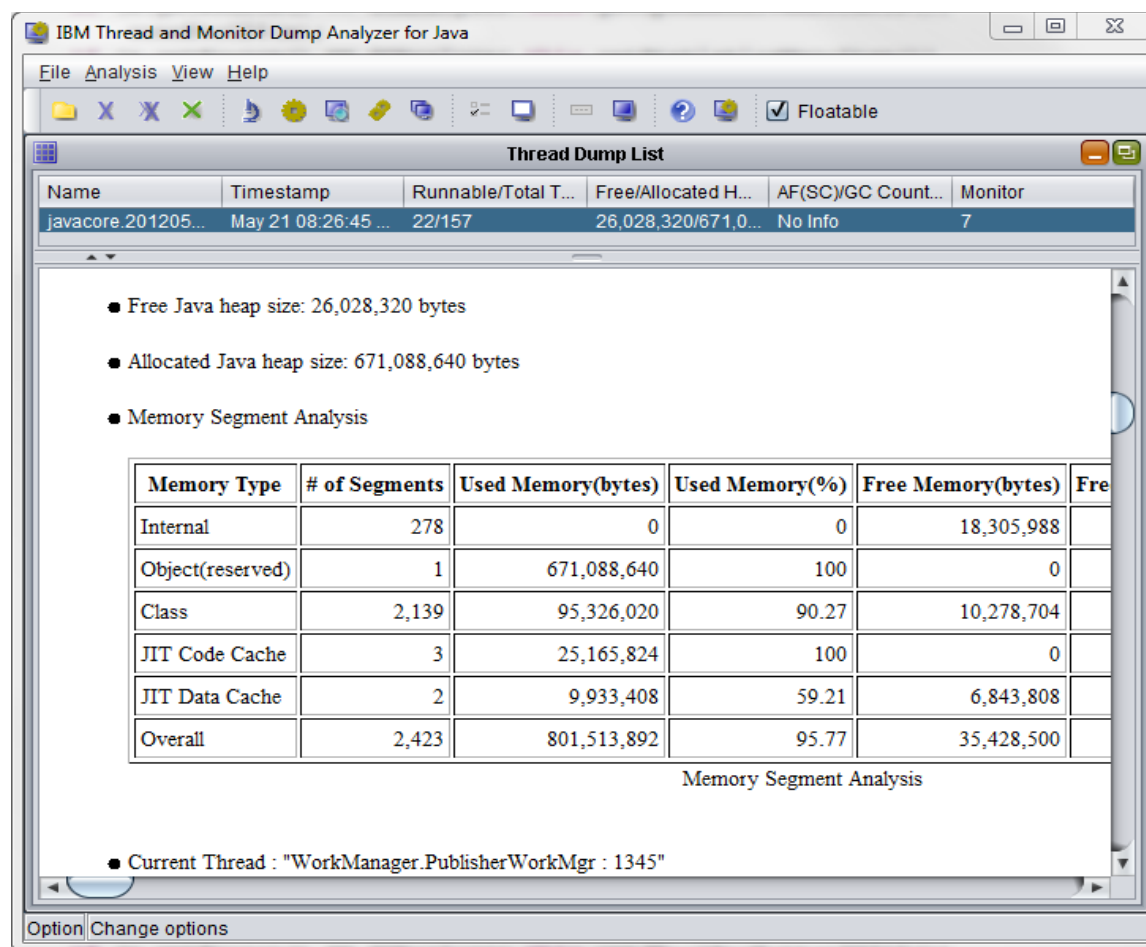http://websphere.sys-con.com/node/1229281

CHAPTER 2:

IBM Thread and Monitor Dump Analyzer for Java
Kill the mystery of Thread Dead Lock

This is an awesome tool to analyze Thread dumps. Even though it is IBM tool, it can analyze Thread dumps from most of the major JVM vendors.Thread dump analysis is one of the critical troubleshooting steps in order to analyze 'slow' or 'hung' Applications.

How does it help you in troubleshooting?

a. Easy to read graphical display of all threads.
b. **Highlights dead locks**
c. Thread Detail view and Monitor Detail view
d. Shows 'Waiting Threads' and 'Blocked by' for each Thread.
e. Ability to 'Compare' two Thread dumps. Extremely useful in revealing 'hung' threads.
f. Detailed 'memory segments' view ('JIT code cache', 'class' etc)

IBM Thread and Monitor Dump Analyzer for Java

File Analysis View Help

☑ Floatable

Thread Dump List

Name	Timestamp	Runnable/Total T...	Free/Allocated H...	AF(SC)/GC Count...	Monitor
javacore.201205...	May 21 08:26:45 ...	22/157	26,028,320/671,0...	No Info	7

- Free Java heap size: 26,028,320 bytes

- Allocated Java heap size: 671,088,640 bytes

- Memory Segment Analysis

Memory Type	# of Segments	Used Memory(bytes)	Used Memory(%)	Free Memory(bytes)	Fre
Internal	278	0	0	18,305,988	
Object(reserved)	1	671,088,640	100	0	
Class	2,139	95,326,020	90.27	10,278,704	
JIT Code Cache	3	25,165,824	100	0	
JIT Data Cache	2	9,933,408	59.21	6,843,808	
Overall	2,423	801,513,892	95.77	35,428,500	

Memory Segment Analysis

- Current Thread : "WorkManager.PublisherWorkMgr : 1345"

Option Change options

How can you obtain this tool?

Free Download:

https://www.ibm.com/developerworks/community/groups/service/html/communityview?communityUuid=2245aa39-fa5c-4475-b891-14c205f7333c

Where can you find more information about this tool?

Webcast replay: Analysis of hangs, deadlocks, and resource contention or monitor bottlenecks using IBM Thread and Monitor Dump Analyzer for Java Technology:

http://www-01.ibm.com/support/docview.wss?uid=swg27011855

CHAPTER 3:

Eclipse Memory Analyzer
Uncover Memory Leaks with Ease

It is the best Java Heap analyzer. Since it is Eclipse based, the user interface is very smooth, rich and intuitive. It comes free with wide varieties of features.

Analyzing Java heap is a sure way to reveal Memory leaks and to fix 'Out of Memory Errors'. But analyzing heap can be very complex and confusing as Heap can literally contain millions of objects. Eclipse Memory analyzer makes analyzing heap effortless.

How does it help you in troubleshooting?

a. Easy to read, graphical display of Java heap
b. Automatically finds 'Leak Suspects'
c. Sorted listing of objects based on the size, enabling you to quickly identify the biggest objects
d. Complete Thread stack trace when available, enabling you to identify the method causing the leak
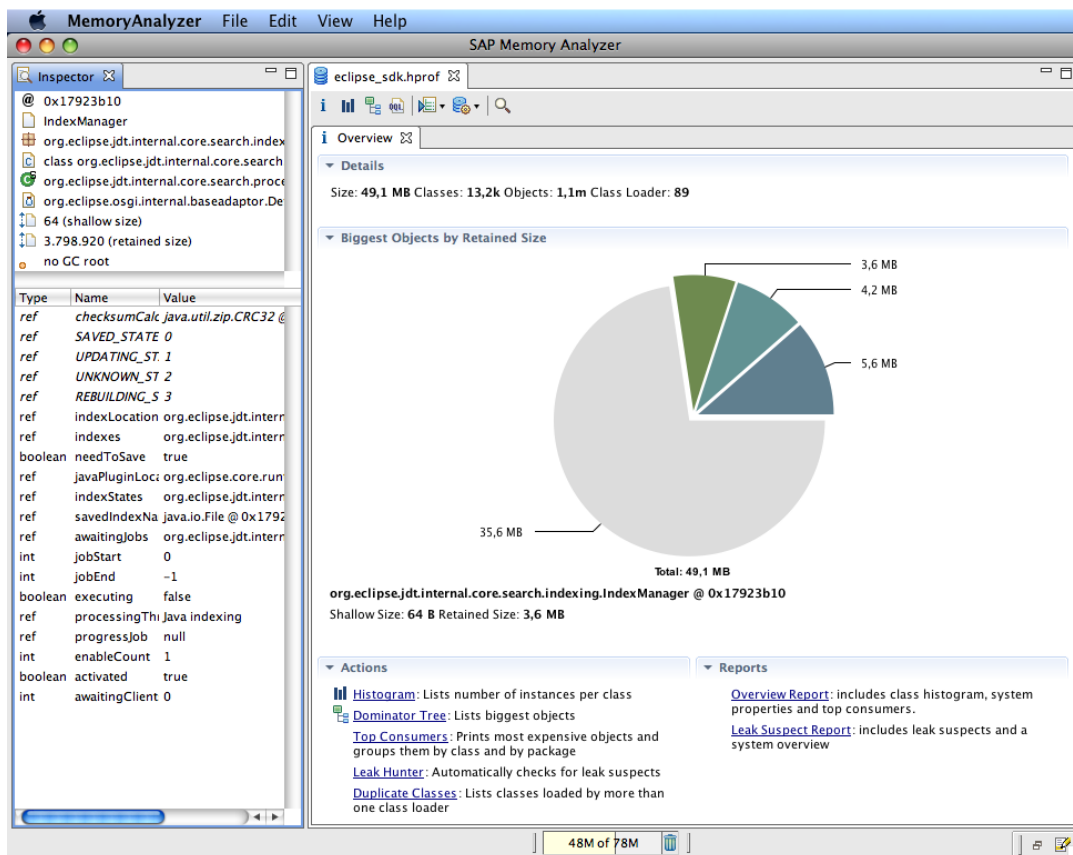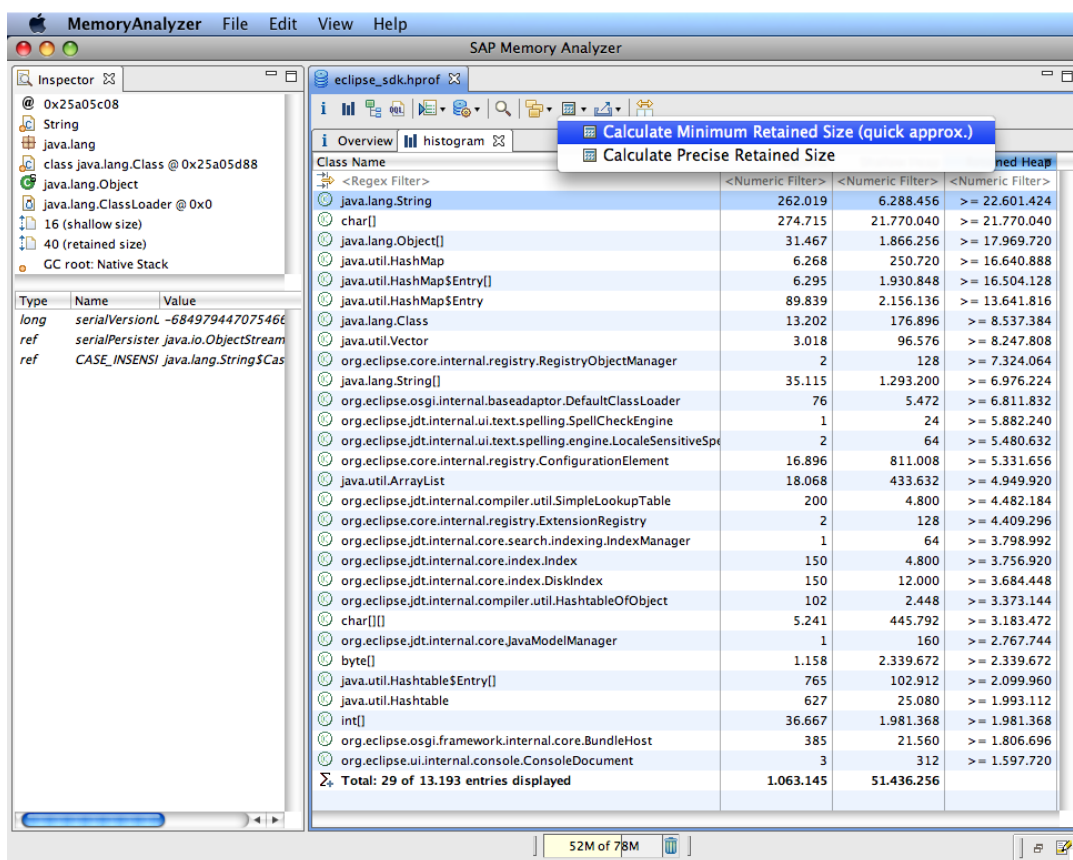
Images source: Eclipse.org

How can you obtain this tool?

Free download:

https://eclipse.org/mat/downloads.php

Where can you find more information about this tool?

Webminar: Memory Analyzer 1.0 - Not Only New, but Noteworthy

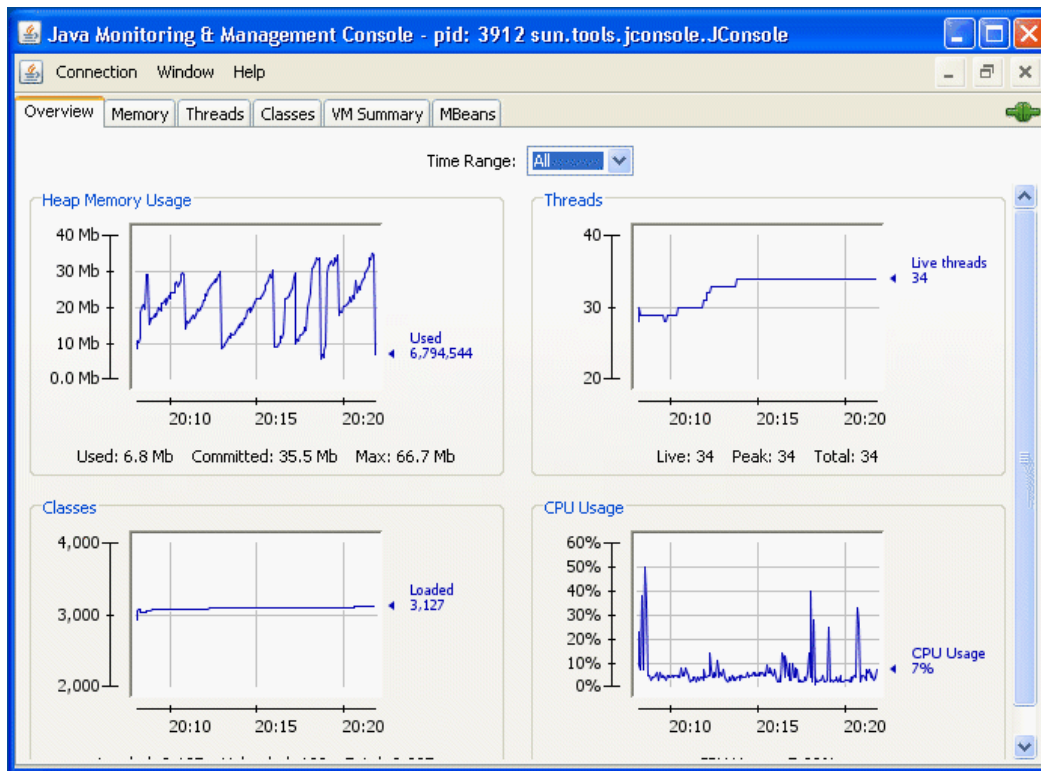http://live.eclipse.org/node/939
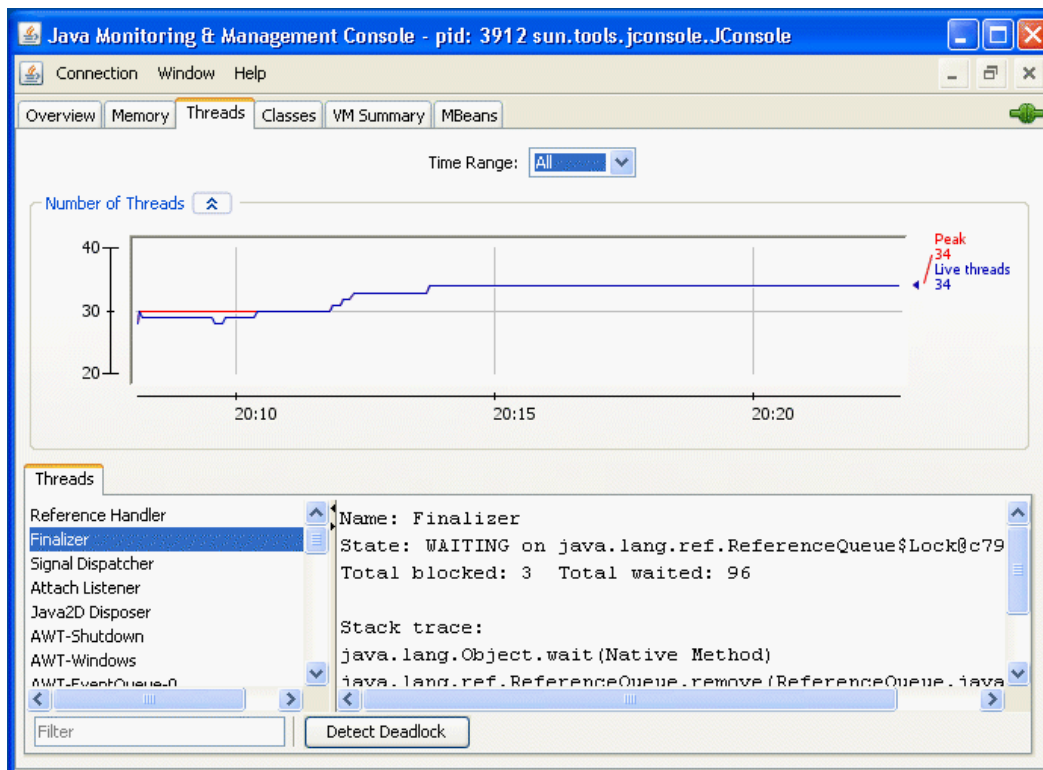
CHAPTER 4:

JConsole
Fast and easy JVM monitoring

A JMX based JVM monitoring utility that comes free with every JDK (note: not JRE). It enables you to monitor several key metrics on your Java application.

How does it help you in troubleshooting?

  a. Easy to read graphical, tabbed view of Key metrics
  b. Monitors the following metrics
   - Memory (including breakdown of various Memory pools such as eden space, tenured space etc)
   - Threads
   - Classes
   - MBeans (you can invoke operations on Mbeans)

Important Note: Jconsole is resource intensive and hence be cautious if you have run this in production environment.

Images source: Oracle

How can you obtain this tool?

Jconsole comes with every JDK. Look for the command in <JDK Install dir>/bin. (Jconsole.exe on Windows)

Note: Jconsole does not come with JRE.

Where can you find more information about this tool?

Oracle Document:

http://docs.oracle.com/javase/7/docs/technotes/guides/management/jconsole.html
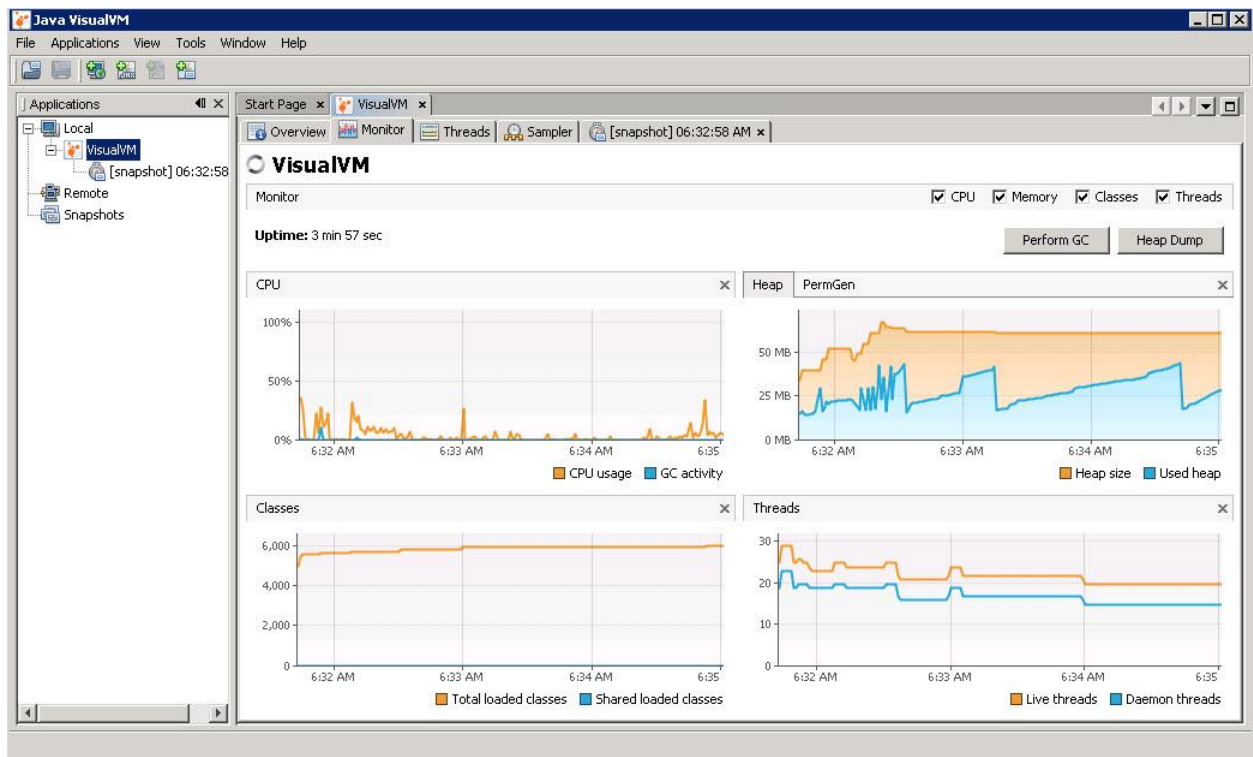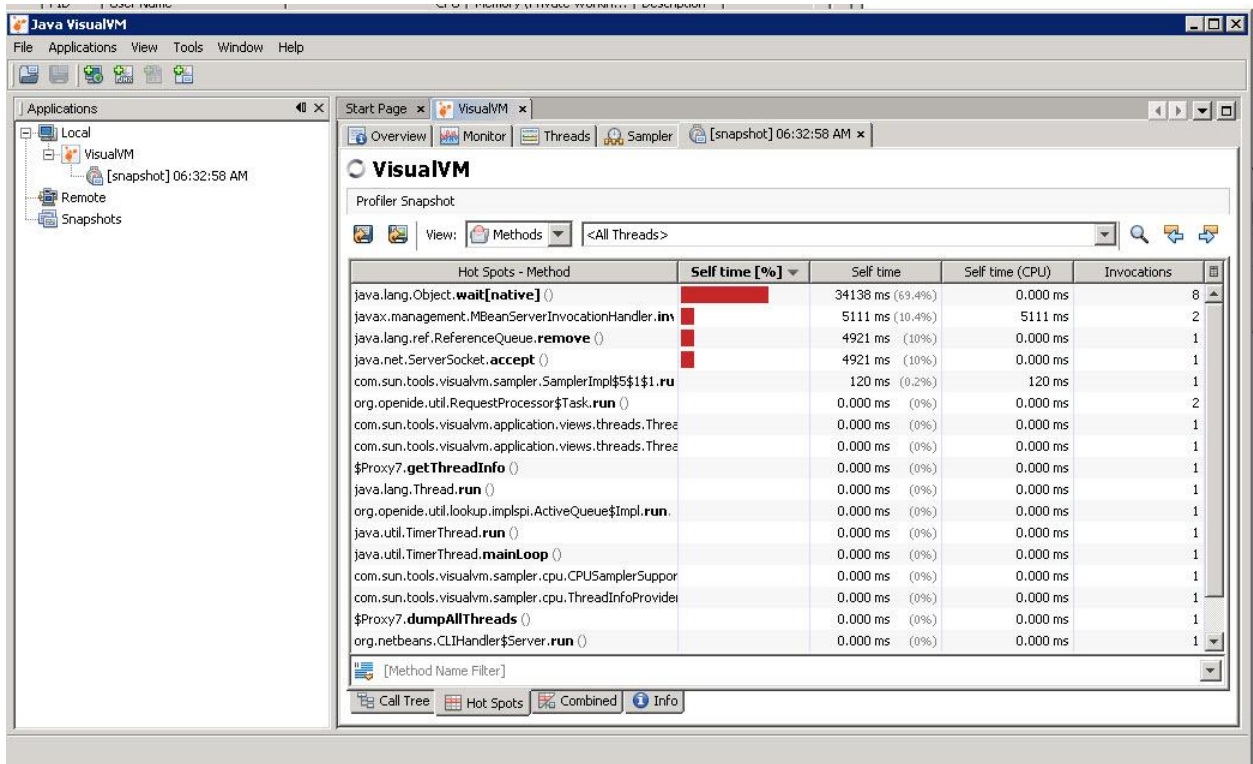
CHAPTER 5:

VisualVM

Need to profile your application? No problem.

VisualVM is a very comprehensive Java monitoring tool that comes with the JDK. It has all the capabilities of Jconsole and some more. The two striking differences are the resource utilization (lighter than Jconsole) and the built in light weight profiler. An additional feature is the ability to integrate the tool with Eclipse.

How does it help you in troubleshooting?

a. Easy to read, graphical display of all the critical metrics
b. Java Heap monitoring with the breakdown of various memory pools (such as Eden space, Tenured space etc)
c. Ability to take and view Thread Dump
d. Ability to take and view Heap Dump
e. Light weight Java profiler – to analyze application performance and memory allocation.
f. Can be integrated with Eclipse or NetBeans.

How can you obtain this tool?

Jconsole comes with every JDK. The tool can be found at <JDK Install dir>/bin (jvisualvm.exe on Windows)

Note: Visual VM does not come with JRE

Where can you find more information about this tool?

Oracle Document:

http://docs.oracle.com/javase/8/docs/technotes/guides/visualvm/index.html
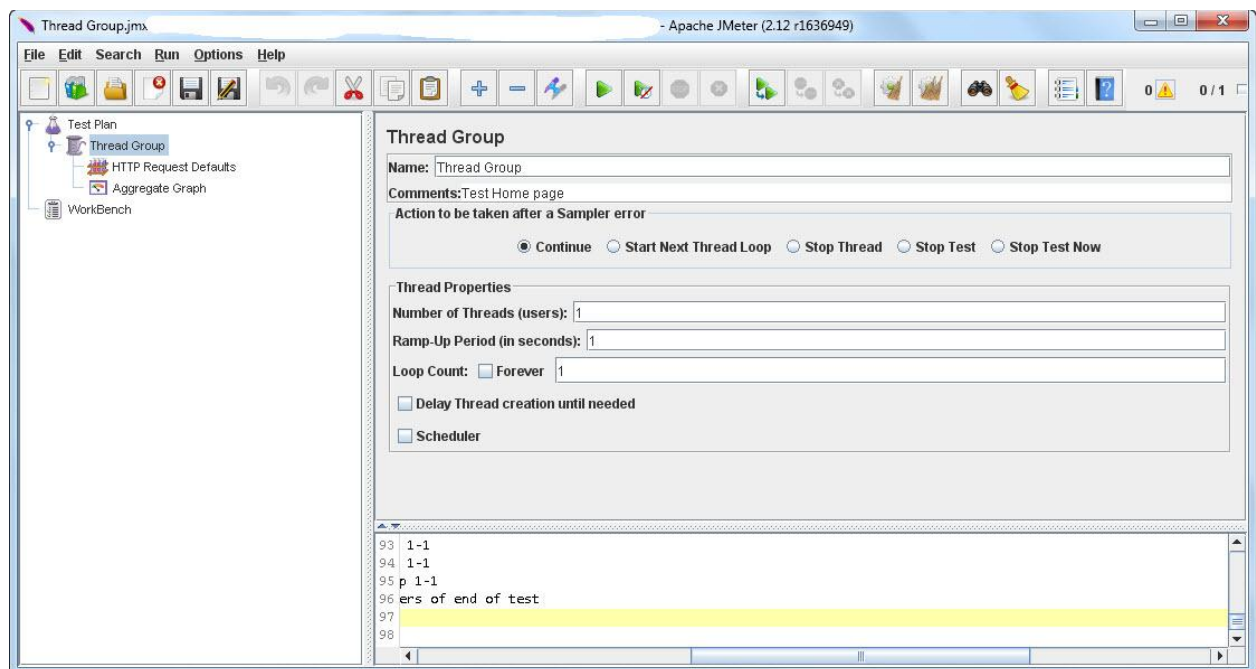
CHAPTER 6:

Jmeter

To fix a problem, you may have to reproduce it first. Say hello to Jmeter.

Jmeter is a fantastic tool for load testing your application to reveal functional and performance issues. Even though it is not a troubleshooting tool by itself, it aids greatly in troubleshooting slow applications. It is very easy to setup.

How does it help you in troubleshooting?

When combined with other free tools such as Visual VM, Jmeter helps in following ways:

1. Load test the application to reveal memory/resource issues or functional issues.
2. Retrieve and analyze the HTTP response codes from the application being tested.
3. Record and repeat tests
4. Non-GUI mode available

How can you obtain this tool?

Free download:

http://jmeter.apache.org/download_jmeter.cgi

Where can you find more information about this tool?

User's manual:

http://jmeter.apache.org/usermanual/index.html

CHAPTER 7:

WIRESHARK

Awaken the Network Guru within you

Wireshark is the best tool for analyzing network packets between the client and the Server. It is fast and intuitive and reveals each request/response including frame and packet level details. For example, if you are troubleshooting SSL handshake issue between the client and server, you can use wireshark to analyze the network capture.

How does it help you in troubleshooting?

1. Capture and analyze tcpdump/windump or other packet capture programs
2. Very detailed protocol information for every packet
3. Filter packets on many conditions (such as all packets originating from an IP address)
4. Search packets
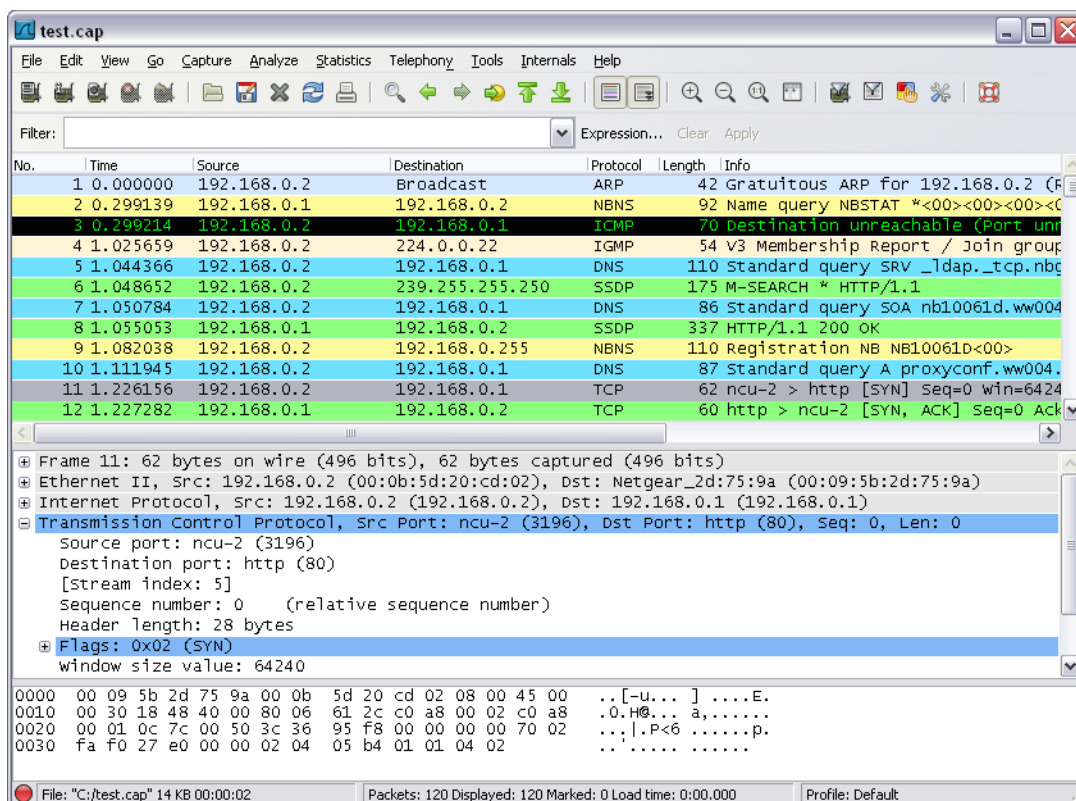5. Versatile GUI with ability to colorize packets based on conditions

Image source: wireshark.org

How can you obtain this tool?

Free download:

https://www.wireshark.org/download.html

Where can you find more information about this tool?

User's manual:

https://www.wireshark.org/docs/wsug_html_chunked/
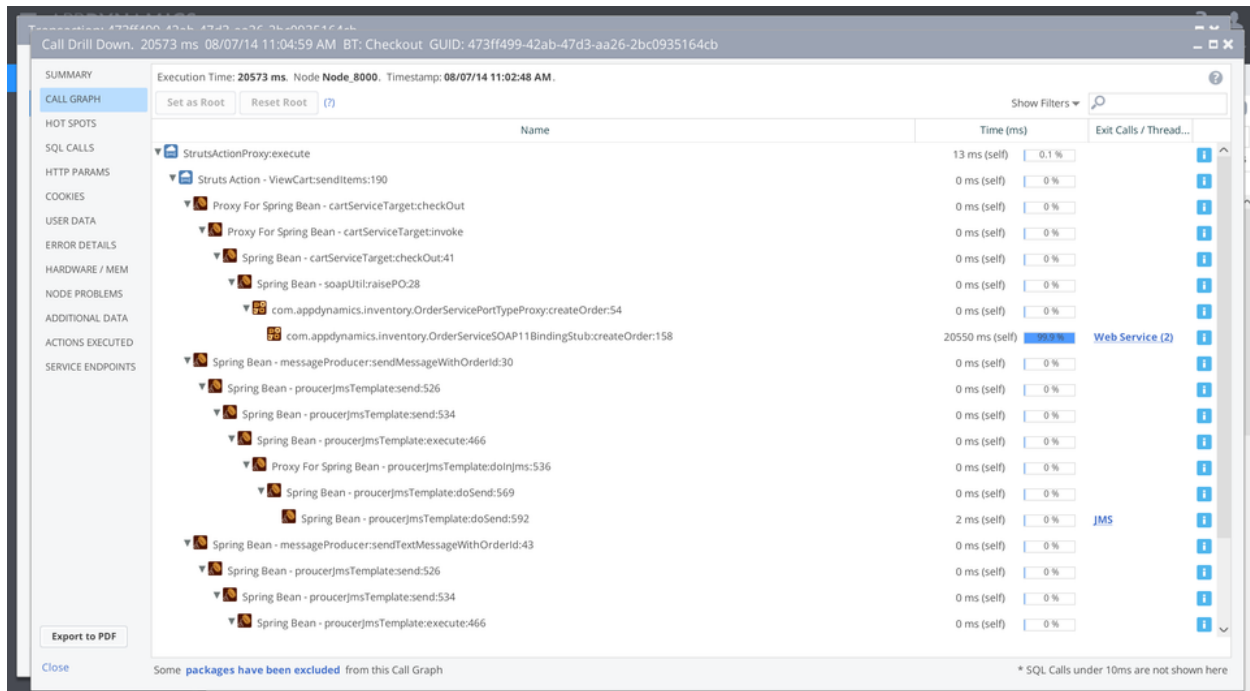
CHAPTER 8:

AppDynamics Lite
Tool so rich you will want to pay for it

Appdynamics Lite is a commercial APM (Application Performance Management) tool. It is one of the new-generation APM tools that provides unbelievable amount of insight into every aspect of your java application.

How does it help you in troubleshooting?

1. Fast, versatile, intuitive GUI revealing key metrics
2. Automatically plots the application flow including discovering standard back ends such as RDBMS
3. Ability to record metrics for 24 hours
4. Deep diagnostics including code level visibility
5. Groups transactions as configurable 'Business Transactions'

Images source: appdynamics.com

How can you obtain this tool?

Free download:

https://portal.appdynamics.com/account/signup/signupform/

Where can you find more information about this tool?

Documentation from Appdynamics site:

https://docs.appdynamics.com/display/PRO40/AppDynamics+Essentials

CHAPTER 9:

NewRelic Lite

You be the Hero who saves the day. NewRelic says 'you are welcome'

New Relic Lite is a commercial APM (Application Performance Management) tool. Like Appdynamics, It is one of the new-generation APM tools that provides unbelievable amount of insight into every aspect of your java application.

How does it help you in troubleshooting?

1. Fast, intuitive graphs
2. Quick revelation of time consuming transactions and methods
3. Application Histogram
4. Lots of Plugins available from New Relic Platform.
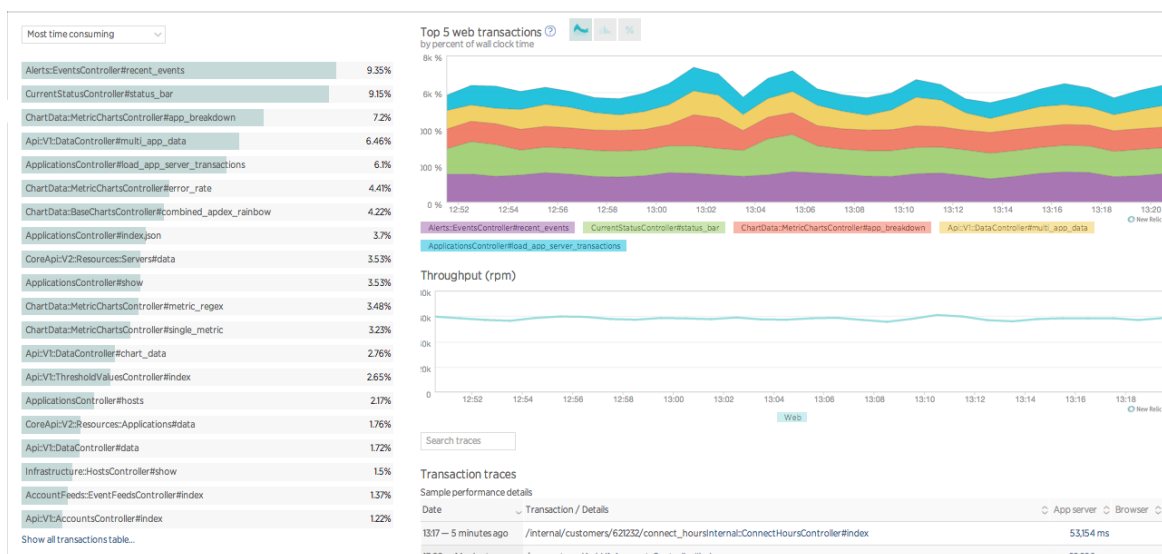
How can you obtain this tool?

Free download: https://newrelic.com/application-monitoring

Where can you find more information about this tool?

Getting Started Guide:

https://try.newrelic.com/rs/newrelic/images/NR_Getting_Started_Guide.pdf?_ga=1.226635741.172
7088364.1416987622

CHAPTER 10:

Conclusion

At the end of the day, you are paid for the VALUE you provide

There is no question regarding the importance of having the right tools to troubleshoot your Java application. The 9 tools presented here are not just the only tools available. But if you master them, you will be well prepared to tackle any type of issue. With any new tool, there is a learning curve and you must be willing to spend some time learning and playing with it. Surely, it will be an investment with huge return – and you will undoubtedly become more valuable to your organization.

Good luck.

About the Author

Karun Subramanian

Application Support Expert and Productivity Enthusiast

I am always in the lookout for tools and processes to increase the productivity of IT support engineers. In my years of experience as a consultant, I have helped several fortune 100 companies run their mission critical applications at its peak performance, securely.

www.karunsubramanian.com