Java™

Java is a trademark of Sun Microsystems, Inc.

JavaOne℠

Monitoring and Troubleshooting Java™ Platform Applications with JDK™ Software

Mandy Chung
Tomas Hurka
Sun Microsystems, Inc.

Agenda

- JDK Monitoring and Troubleshooting Tools
- Common Performance Problems
  - Deadlocks
  - Memory leak
  - Finalizers
  - High lock contentions
  - OutOfMemoryError

Monitoring and Troubleshooting Tools

Feature	CLI Tools	GUI Tools
List JVMs and runtime info	jps, jinfo	jconsole, jvisualvm
Thread dump Detect deadlock	jstack	jconsole, jvisualvm
Memory usage	jmap, jstat	jconsole, jvisualvm
Heap dump and analysis	jmap and jhat	jconsole, jvisualvm
JVM perf counters	jstat	jvisualvm
Application and JVM MBeans		jconsole, jvisualvm

JConsole

- <JDK>/bin/jconsole (since JDK 5)
- JMX-compliant monitoring tool
  - Connect to a local or remote application
  - Monitor VM resources such as memory, threads, classes and application MBeans

Java VisualVM

- <JDK>/bin/jvisualvm (since 6u7)
  - List local and remote Java applications
  - Show JVM configuration and runtime env.
  - Monitor performance, memory consumption, threads
  - Lightweight profiler and heap walker
  - Ability to snapshot thread dump and heap dump
- Extensibility
  - Add plugins e.g. VisualGC, MBeans plugin
  - Build your own plugin

Java VisualVM Demos

- Deadlocks
- Memory leak
- Finalizers
- High lock contentions
- Profiling support

Diagnose Deadlocks

- HotSpot VM built-in deadlock detection
  - Object monitors and java.util.concurrent ownable synchronizers
- Tools
  - jvisualvm
  - jstack [-l] <pid>
    - -l option to list owned java.util.concurrent locks

Diagnose Memory Leak

- Monitor memory usage
  - Symptom: unexpected increasing use of memory
  - Tool: jvisualvm or jstat
- Identify a leak suspect from heap histogram
  - Symptom: a class with a high growth rate or unexpected number of instances
  - Tool: jvisualvm or jmap -histo[:live]
- Obtain and analyze heap snapshot
  - Symptom: objects being referenced unintentionally
  - Tool: jvisualvm or jmap -dump[:live] with jhat

Diagnose Excessive Use of Finalizers

- Similar to memory leak
  - Symptom: non-decreasing memory usage
- Watch number of objects pending for finalization
  - Tool: jvisualvm or jconsole

Diagnose Hot Lock Contention

- Threads are blocked and waiting to acquire locks
- Tool:
  - jvisualvm to check thread state
  - jconsole to check thread contention statistics

Diagnose OutOfMemoryError

- Error message indicates which type of memory: Java heap space, PermGen space, native
- Heap dump generated at OutOfMemoryError
- Tool:
  - jvisualvm
  - jinfo -flag +HeapDumpOnOutOfMemoryError

Diagnose Hung Process

- Inspect a hung process on Solaris and Linux
  - jvisualvm
  - jmap -F [-histo|-dump] <pid>|<corefile>
  - jstack -F [-m] <pid>|<corefile>
  - jinfo <pid>|<corefile>
- Will support on Windows in a future release

Solaris DTrace

- HotSpot built-in probes to observe GC, threads, class loading, etc
- Solaris Dynamic Tracing Guide
  - http://docs.sun.com/app/docs/doc/817-6223
- HotSpot JVM DTrace Probes
  - http://java.sun.com/javase/6/docs/technotes/guides/vm/dtrace.html

More Information

- TS-4247 Getting More Out Of Java VisualVM
  - Friday 12:10pm - 1:10pm
- Sun VM Performance Booth
  - Thursday 10:00am - 2:00pm
- VisualVM Project
  - http://visualvm.dev.java.net
  - Check out the VisualVM screencast
- VisualVM Blogging Contest
  - http://java.sun.com/community/javavisualvm/
-

References

- Article
  - http://java.sun.com/developer/technicalArticles/J2SE/monitoring/
- Documentations
  - http://java.sun.com/javase/6/docs/technotes/guides/management/
  - http://java.sun.com/javase/6/webnotes/trouble/
  - http://java.sun.com/javase/6/docs/technotes/guides/visualvm/

Q&A

Thank You

Mandy Chung
http://weblogs.java.net/blog/mandychung

Tomas Hurka
http://visualvm.dev.java.net/