



# X33EJA Výkonové Aspekty Java/JEE Monitoring a optimalizace



Martin Ptáček, Praha 2011

**I**

**Hlavní oblasti ovlivňující výkon Java/JEE**

**II**

**Proč monitorovat?**

**III**

**Nástroje pro monitoring**

**IV**

**Monitoring Glassfish**

**V**

**Otázky a odpovědi**

- ❑ Správa paměti
- ❑ Správa prostředků a zdrojů
- ❑ Optimalizace datové vrstvy
- ❑ Návrh a implementace

### ❑ **Nastavení JVM**

- Optimalizace parametrů, GC Tuning
- -server -client
- -Xmx -Xms -Xss -XX:MaxPermSize
- <http://java.sun.com/performance/reference/whitepapers/tuning.html>

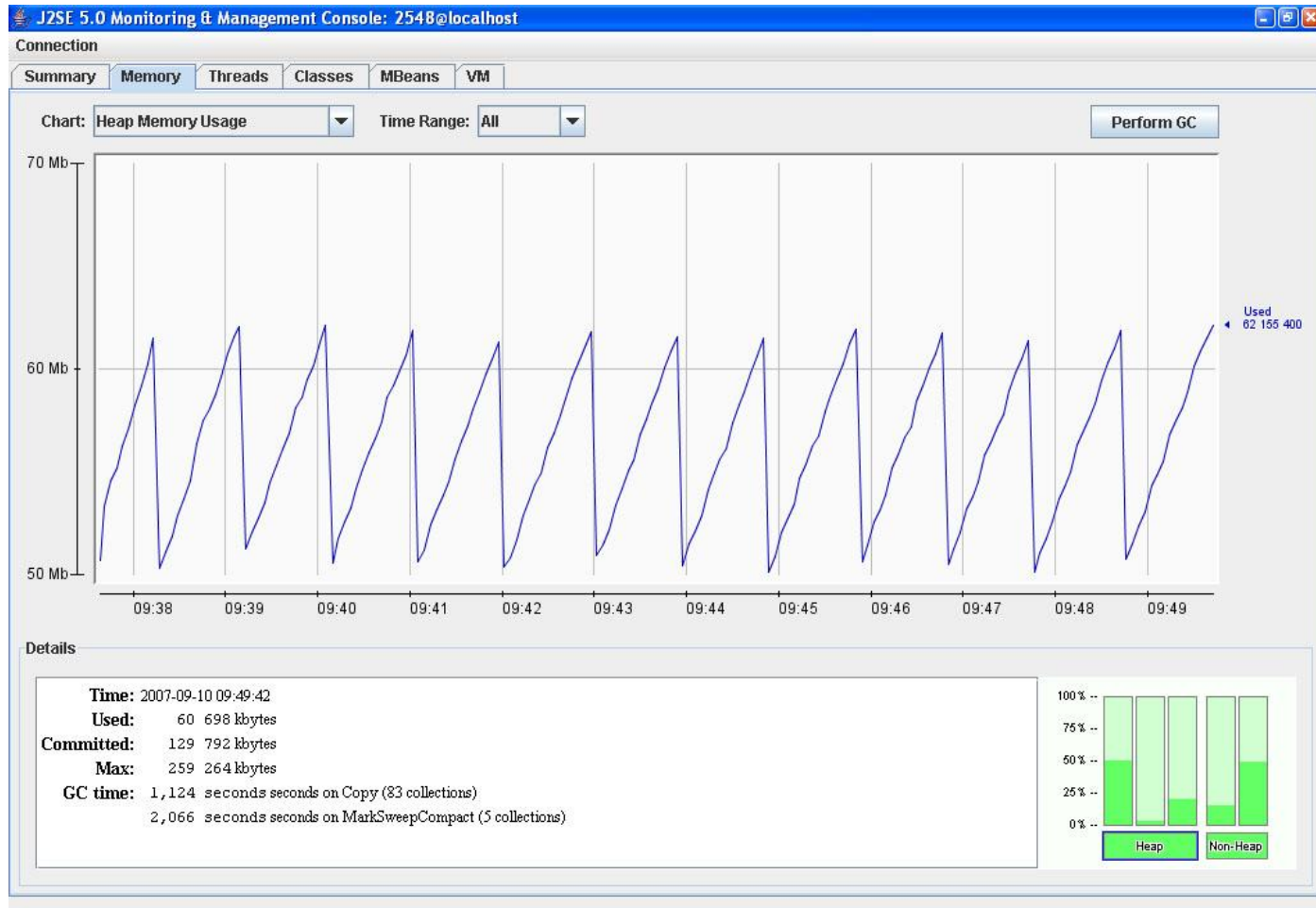
### ❑ **Memory Leaks**

### ❑ **Optimalizace programu**

- HTTP Session
- Object Pooling
- Caching

# Hlavní oblasti ovlivňující výkon Java/JEE

## Správa paměti



### ❑ Execution Threads

- Maximální počet paralelně zpracovávaných požadavků

### ❑ Pooling & Caching

- Connection Pool
- Thread Pool
- Instance Pool

### ❑ Clustering

- LoadBalancing, FailOver

- ❑ **Výběr kvalitního JDBC driveru**
  
- ❑ **Správa zdrojů**
  - Connection Pool (min, max, validation, ...)
  - PreparedStatement, Statement Caching
  - Instance Pool
  
- ❑ **Databáze**
  - Analýza a optimalizace SQL
  - Indexy
  - Parametry databáze, update statistik

- ❑ **Využití asynchronních procesů**
  - JMS
  
- ❑ **Minimalizace použití XML**
  
- ❑ **Využití cache**
  - Hibernate Second Level Cache
  - Ehcache
  
- ❑ **Minimalizace použití vzdáleného volání (Remoting)**
  
- ❑ **Pečlivý výběr knihoven třetích stran**



- ❑ **Zajištění dostupnosti aplikace**
  - Preemptivní X Reakční
  
- ❑ **Odhalení performance problémů**
  - Testovací prostředí
  - Produkční prostředí
  
- ❑ **Analýza problémových stavů**
  
- ❑ **Reporting**

### Profiling X Monitoring

### Profiler

- Vývojářský nástroj
- Velký overhead

### Monitorovací nástroj

- Vhodný do testovacího i produkčního prostředí

- Nízký overhead**
- Neovlivňování monitorované aplikace**
- Kvalita poskytovaných metrik a efektivní práce**
- Monitoring 24x7**
- Snadná instalace a konfigurace**
- System Alert - Action**
- Reporty**

### ❑ Problémy s CPU

- Thread Dump

### ❑ Problémy s pamětí

- Heap Dump
  - -XX:+HeapDumpOnOutOfMemoryError
  - Analýza pomocí podpůrných nástrojů
- Různé oblasti paměti (Heap, Non-Heap)

### ❑ Problémy s využíváním ostatních zdrojů

- JMX metriky

### ❑ JMX (Java Management Extensions)

- <http://www.oracle.com/technetwork/java/javase/tech/javamanagement-140525.html>
- Umožňuje přístup k aplikačním objektům
  - Nastavovat atributy
  - Volat metody, inicializovat aplikace nebo komponenty
- Možnost získávat statická data
- Registrovat události
- Atd...

### ❑ MC4J (<http://sourceforge.net/projects/mc4j/>)

- JMX Browser

- ❑ **jps** (JVM Process Status Tool)
  - list JVM process
  
- ❑ **jstat** (JVM Statistics Monitoring Tool)
  - performance statistics about class loader, compiler, gc
  - *jstat -gcutil 1234 250 7*
  
- ❑ **jmap, jhat**
  - memory profiling, HeapDump
  - *jmap -dump:format=b,file=heap.bin 1234*
  - *jhat -J-mx768m -stack false heap.bin*
  
- ❑ **jstack**
  - ThreadDump, Threads Stack Trace
  - *jstack 1234*

- ❑ **jconsole (Java Monitoring and Management Console)**
  - Memory, Threads, Classes, MBeans
  - <http://download.oracle.com/javase/6/docs/technotes/guides/management/jconsole.html>
  
- ❑ **visualvm (All-in-One Java Troubleshooting Tool)**
  - <http://visualvm.java.net/>
  - Modernější nástroj pro monitoring Java aplikací
  - Obsahuje Profiler
  - <http://download.oracle.com/javase/6/docs/technotes/guides/visualvm/index.html>
  - <http://pietrowski.info/2009/06/visualvm-performance-tuning-tool/>
  - jvisualvm in last java distribution

- ❑ Log4j, SLF4j (<http://www.slf4j.org/>)
- ❑ SAP Memory Analyzer
- ❑ Memory Analyzer (<http://www.eclipse.org/mat/>)
- ❑ P6Spy, Iron Track SQL (<http://sourceforge.net/projects/p6spy/>)
  - Monitoring JDBC rozhraní
- ❑ JIP (<http://jiprof.sourceforge.net/>)
  - Profiler



### ❑ CA Wily Introscope

- Java/JEE, .NET
- Real-Time Application Performance Monitoring (APM)
- Proaktivní detekce a diagnostika performance problémů
- Identifikace komponent a jejich závislostí

### ❑ HP Diagnostics

### ❑ APMXpert (<http://www.opnet.com>)

### ❑ DynaTrace (<http://www.dynatrace.com>)

## ❑ Static JMX Service URL

- *service:jmx:rmi:///jndi/rmi://localhost:8686/jmxrmi*

## ❑ Monitoring Service

- Nastavitelné v „Admin Console“
- JVM, HTTP Service, Transaction Service, JMS Service, ORB, Web Container, EJB Container, JDBC Connection Pool, Thread Pool

## ❑ Application Server / Monitor

- Možnost jednoduchého prohlížení metrik přímo v „Admin Console“

## □ jconsole, visualvm

- Memory, CPU, Classes, Threads, Profiler
- JMX Metrics: root *com.sun.appserv*
  - *jdbc-connection-pool*
  - *http-listener-1*
  - *keep-alive*
  - *connection-queue*
  - ...

## □ proprietary

- JMX Metric web viewer
- JMX Metric automatic store

## □ Tuning Glassfish

- <http://download.oracle.com/docs/cd/E19879-01/820-4343/index.html>
- [http://download.oracle.com/docs/cd/E18930\\_01/html/821-2416/ghmct.html](http://download.oracle.com/docs/cd/E18930_01/html/821-2416/ghmct.html)
- <http://jfarvand.wordpress.com/2009/11/27/putting-glassfish-v3-in-production-essential-surviving-guide/>
- [http://blogs.sun.com/binublog/entry/glassfish\\_tuning\\_http\\_connection\\_queue](http://blogs.sun.com/binublog/entry/glassfish_tuning_http_connection_queue)
- <http://www.docstoc.com/docs/3667971/Tuning-Your-GlassFish-Performance-Tips-Deep-Singh-Enterprise-Java-Performance>

**Otázky?**

**Martin Ptáček**

[martin.ptacek@wincor-nixdorf.cz](mailto:martin.ptacek@wincor-nixdorf.cz)

+420 739 681 075

**Wincor Nixdorf s.r.o.**

Lužná 591, 160 00 Praha 6

<http://www.wincor-nixdorf.cz>