JVMD @ Therap Services

How JVMD Plays a Vital Role in Therap Applications

Mojahedul Hoque Abul Hasanat
CTO, Therap Services

Therap Services, LLC

- Documentation and Communication Software for MR/DD
  - EHR for the DD industry is the closest for describing us
  - Niche segment in the health sector
- Improve quality of life for people with DD by improving efficiency of delivery through communication
- SaaS business model
- 150K+ active users
- 1000+ providers in 48 states
- State customers
  - Extensive usage for DD in DHS ND and DHHS NE
- 150+ employees
- Based in CT, dev center in Bangladesh

The Application

- The application *is* our business
- 1M+ lines of code
- 60+ modules
- 1M+ sustained HTTP requests/hour
- 30K+ peak requests/minute
- 6000+ concurrent users
- Based on JEE and the Spring Framework
  - Hibernate
  - Seam
  - GRAILS

Delivery Platform

- 2 identical sites in two states
- Primary hosts (per site):
  - 4 WebLogic application servers in cluster
  - 1 Memory based data server (in-house, java)
  - 1 Oracle database server
  - 1 NetApp storage (SAN)
  - 1 F5 Load balancer
- Supporting hosts
- Use Dyn for site high availability
- Data replication with Oracle Golden Gate

What Matters

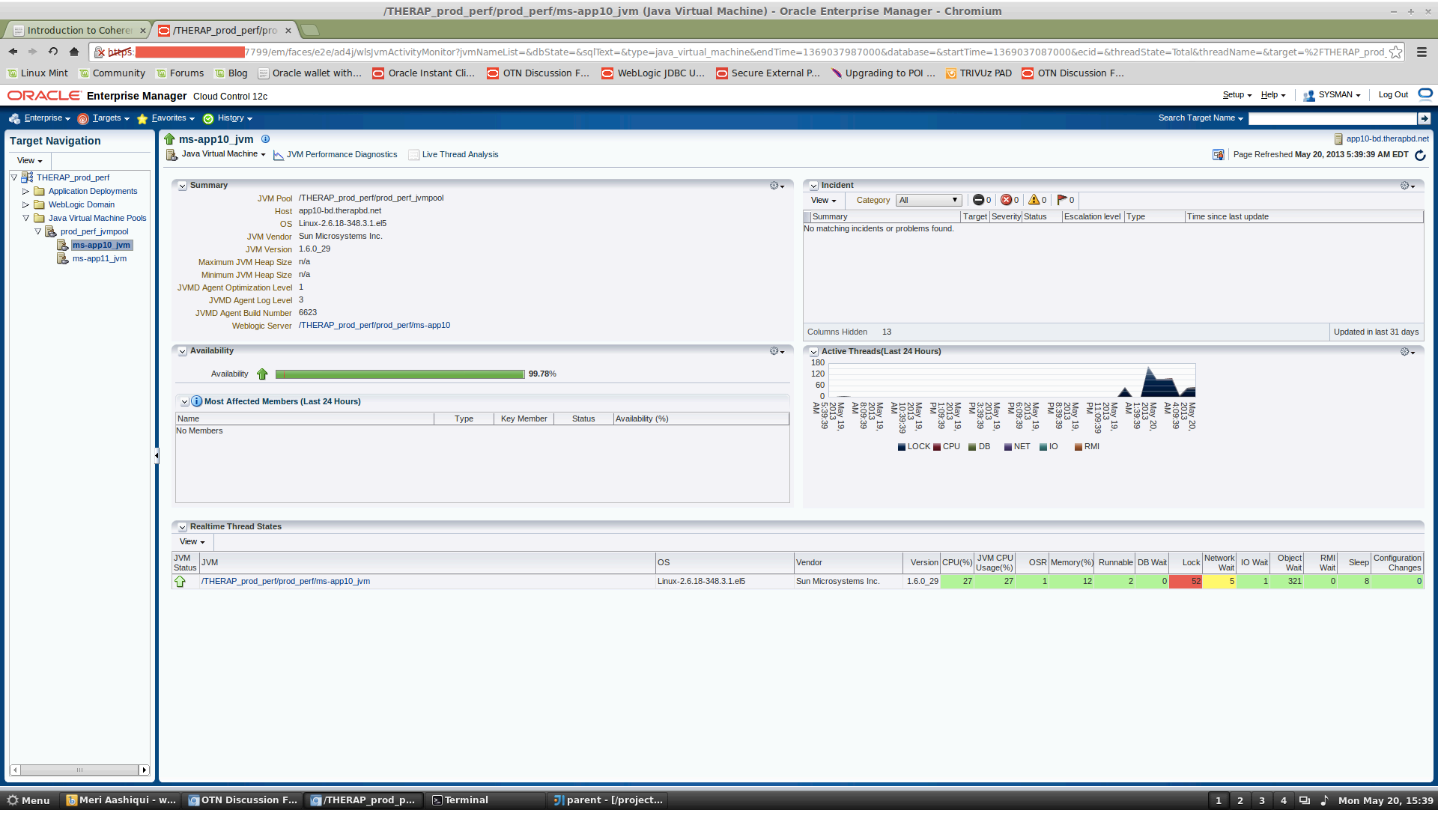
- Availability
  - Application is used 24x7
  - Application use is critical to the business of our customers
- Performance
  - A user needs to spend as little time as possible in our application
  - Most users use it daily, multiple times
- *Data integrity*
- Fast development turnaround

How JVMD Helped Us

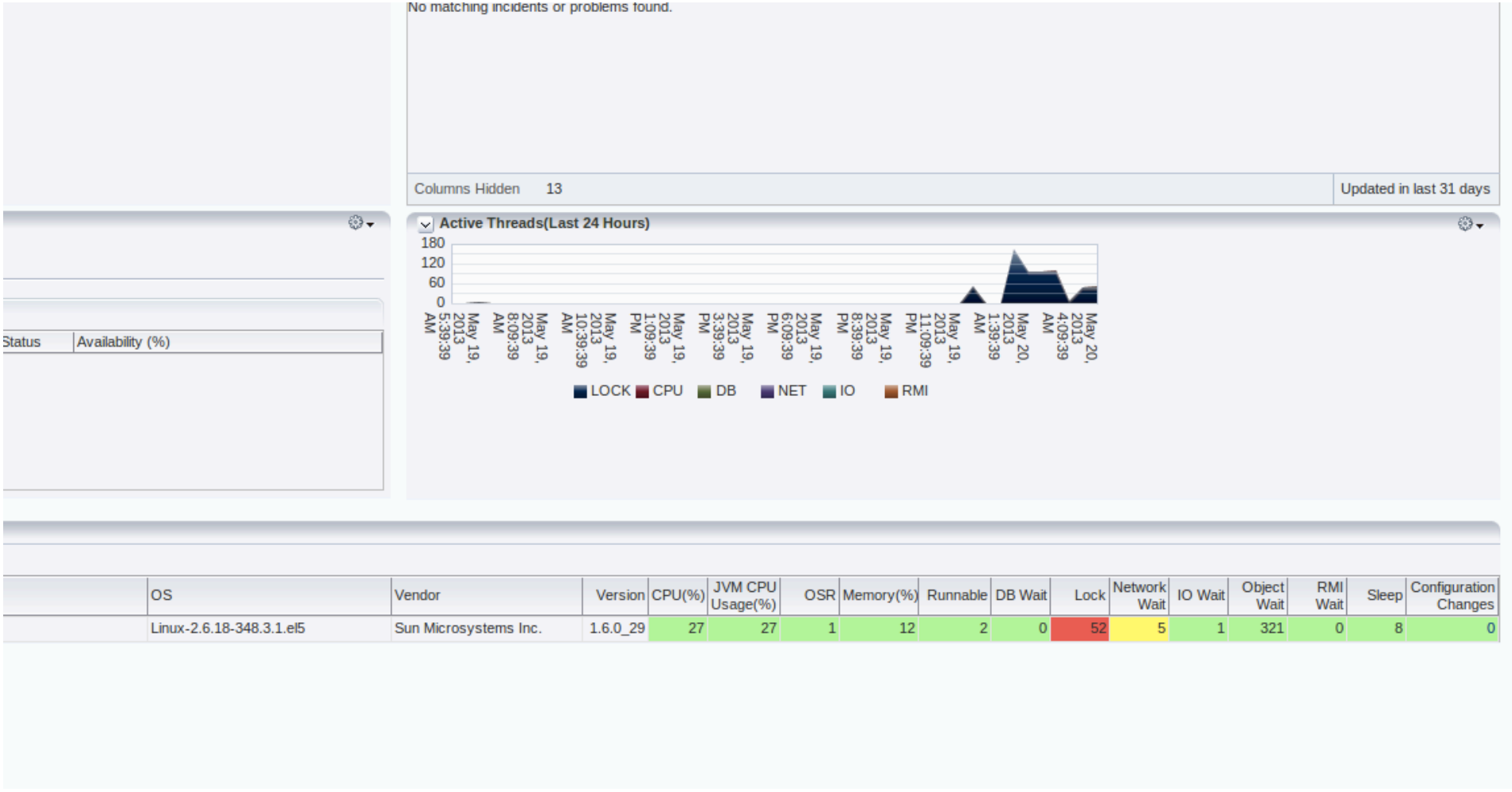
The log4j bottleneck

- During load testing, we could not increase load beyond a certain point
- CPU load was low
- JVMD showed us something that we could hardly believe
- Many threads were contending for lock for writing to the log file
- The contention only shows up at high loads
- Used JVMD heavily to find the best logging backend and the best configuration

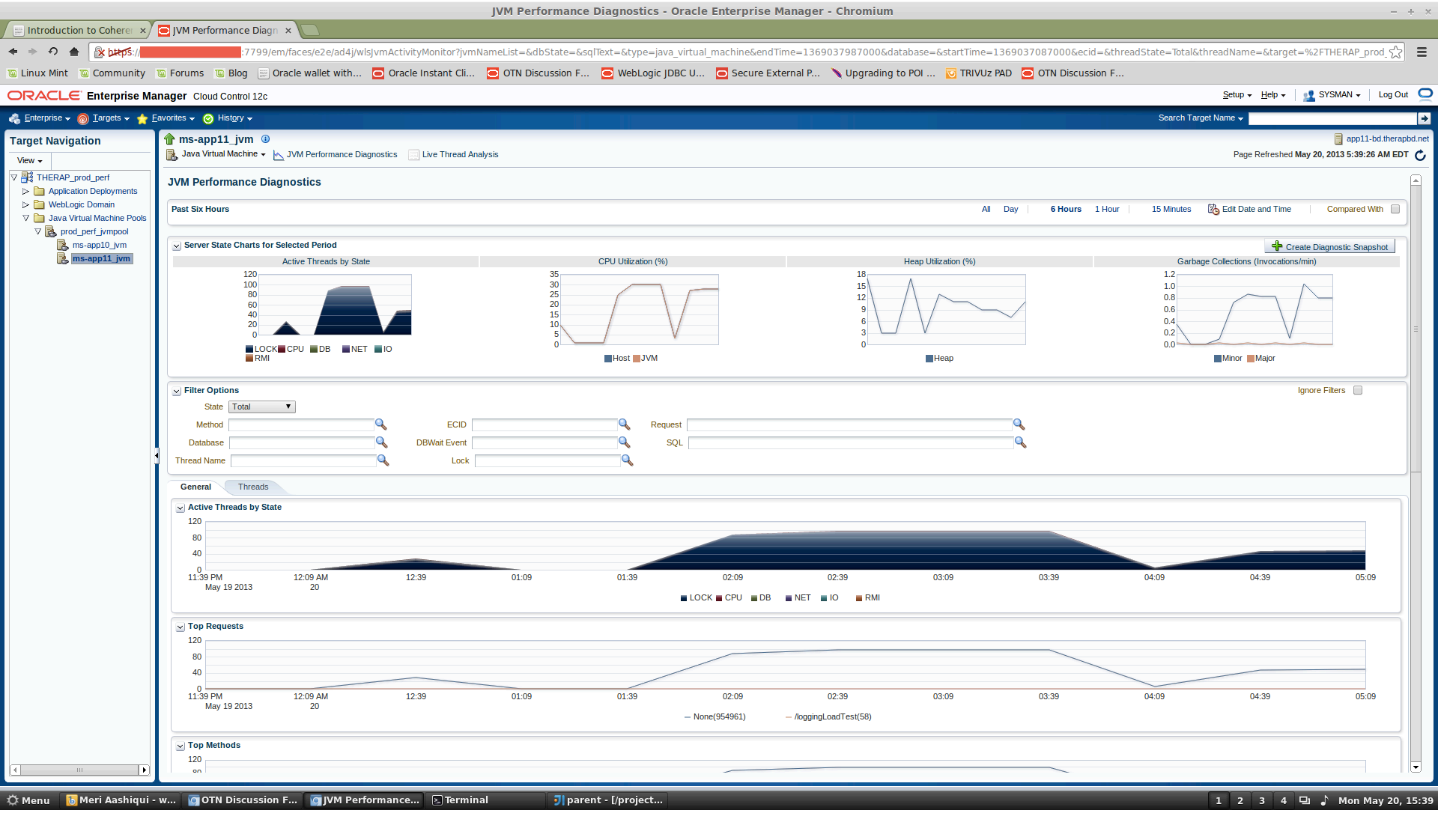
log4j...


/THERAP_prod_perf/prod_perf/ms-app10_jvm (Java Virtual Machine) - Oracle Enterprise Manager - Chromium
Introduction to Cohere
/THERAP_prod_perf/pro
https://[redacted]7799/em/faces/e2e/ad4j/wlsJvmActivityMonitor?jvmNameList=&dbState=&sqlText=&type=java_virtual_machine&endTime=1369037987000&database=&startTime=1369037087000&ecid=&threadState=Total&threadName=&target=%2FTHERAP_prod
Linux Mint
Community
Forums
Blog
Oracle wallet with...
Oracle Instant Cli...
OTN Discussion F...
WebLogic JDBC U...
Secure External P...
Upgrading to POI ...
TRIVUz PAD
OTN Discussion F...
ORACLE Enterprise Manager Cloud Control 12c
Setup
Help
SYSMAN
Log Out
Enterprise
Targets
Favorites
History
Search Target Name
Target Navigation
View
THERAP_prod_perf
Application Deployments
WebLogic Domain
Java Virtual Machine Pools
prod_perf_jvmpool
ms-app10_jvm
ms-app11_jvm
ms-app10_jvm
Java Virtual Machine
JVM Performance Diagnostics
Live Thread Analysis
app10-bd.therapbd.net
Page Refreshed May 20, 2013 5:39:39 AM EDT
Summary
JVM Pool /THERAP_prod_perf/prod_perf_jvmpool
Host app10-bd.therapbd.net
OS Linux-2.6.18-348.3.1.el5
JVM Vendor Sun Microsystems Inc.
JVM Version 1.6.0_29
Maximum JVM Heap Size n/a
Minimum JVM Heap Size n/a
JVMD Agent Optimization Level 1
JVMD Agent Log Level 3
JVMD Agent Build Number 6623
Weblogic Server /THERAP_prod_perf/prod_perf/ms-app10
Incident
View
Category All
Summary
Target
Severity
Status
Escalation level
Type
Time since last update
No matching incidents or problems found.
Columns Hidden 13
Updated in last 31 days
Availability
Availability
99.78%
Most Affected Members (Last 24 Hours)
Name
Type
Key Member
Status
Availability (%)
No Members
Active Threads(Last 24 Hours)
180
120
60
0
LOCK
CPU
DB
NET
IO
RMI
Realtime Thread States
View
JVM Status
JVM
OS
Vendor
Version
CPU(%)
JVM CPU Usage(%)
OSR
Memory(%)
Runnable
DB Wait
Lock
Network Wait
IO Wait
Object Wait
RMI Wait
Sleep
Configuration Changes
/THERAP_prod_perf/prod_perf/ms-app10_jvm
Linux-2.6.18-348.3.1.el5
Sun Microsystems Inc.
1.6.0_29
27
27
1
12
2
0
52
5
1
321
0
8
0
Menu
Meri Aashiqui - w...
OTN Discussion F...
/THERAP_prod_p...
Terminal
parent - [/project...
1
2
3
4
Mon May 20, 15:39

log4j…


No matching incidents or problems found.
Columns Hidden 13
Updated in last 31 days
Active Threads(Last 24 Hours)
180
120
60
0
5:39:39 AM May 19, 2013
8:09:39 AM May 19, 2013
10:39:39 AM May 19, 2013
1:09:39 PM May 19, 2013
3:39:39 PM May 19, 2013
6:09:39 PM May 19, 2013
8:39:39 PM May 19, 2013
11:09:39 PM May 19, 2013
1:39:39 AM May 20, 2013
4:09:39 AM May 20, 2013
LOCK CPU DB NET IO RMI
Status
Availability (%)
OS
Vendor
Version
CPU(%)
JVM CPU Usage(%)
OSR
Memory(%)
Runnable
DB Wait
Lock
Network Wait
IO Wait
Object Wait
RMI Wait
Sleep
Configuration Changes
Linux-2.6.18-348.3.1.el5
Sun Microsystems Inc.
1.6.0_29
27
27
1
12
2
0
52
5
1
321
0
8
0

log4j...


JVM Performance Diagnostics - Oracle Enterprise Manager - Chromium
Introduction to Cohere
JVM Performance Diagn
Linux Mint
Community
Forums
Blog
Oracle wallet with...
Oracle Instant Cli...
OTN Discussion F...
WebLogic JDBC U...
Secure External P...
Upgrading to POI ...
TRIVUz PAD
OTN Discussion F...
ORACLE Enterprise Manager Cloud Control 12c
Setup
Help
SYSMAN
Log Out
Enterprise
Targets
Favorites
History
Search Target Name
Target Navigation
View
THERAP_prod_perf
Application Deployments
WebLogic Domain
Java Virtual Machine Pools
prod_perf_jvmpool
ms-app10_jvm
ms-app11_jvm
ms-app11_jvm
Java Virtual Machine
JVM Performance Diagnostics
Live Thread Analysis
app11-bd.therapbd.net
Page Refreshed May 20, 2013 5:39:26 AM EDT
JVM Performance Diagnostics
Past Six Hours
All
Day
6 Hours
1 Hour
15 Minutes
Edit Date and Time
Compared With
Server State Charts for Selected Period
Create Diagnostic Snapshot
Active Threads by State
CPU Utilization (%)
Heap Utilization (%)
Garbage Collections (Invocations/min)
LOCK CPU DB NET IO RMI
Host JVM
Heap
Minor Major
Filter Options
Ignore Filters
State
Total
Method
ECID
Request
Database
DBWait Event
SQL
Thread Name
Lock
General
Threads
Active Threads by State
11:39 PM May 19 2013
12:09 AM 20
12:39
01:09
01:39
02:09
02:39
03:09
03:39
04:09
04:39
05:09
LOCK CPU DB NET IO RMI
Top Requests
None(954961)
/loggingLoadTest(58)
Top Methods
Menu
Meri Aashiqui - w...
OTN Discussion F...
JVM Performance...
Terminal
parent - [/project...
Mon May 20, 15:39

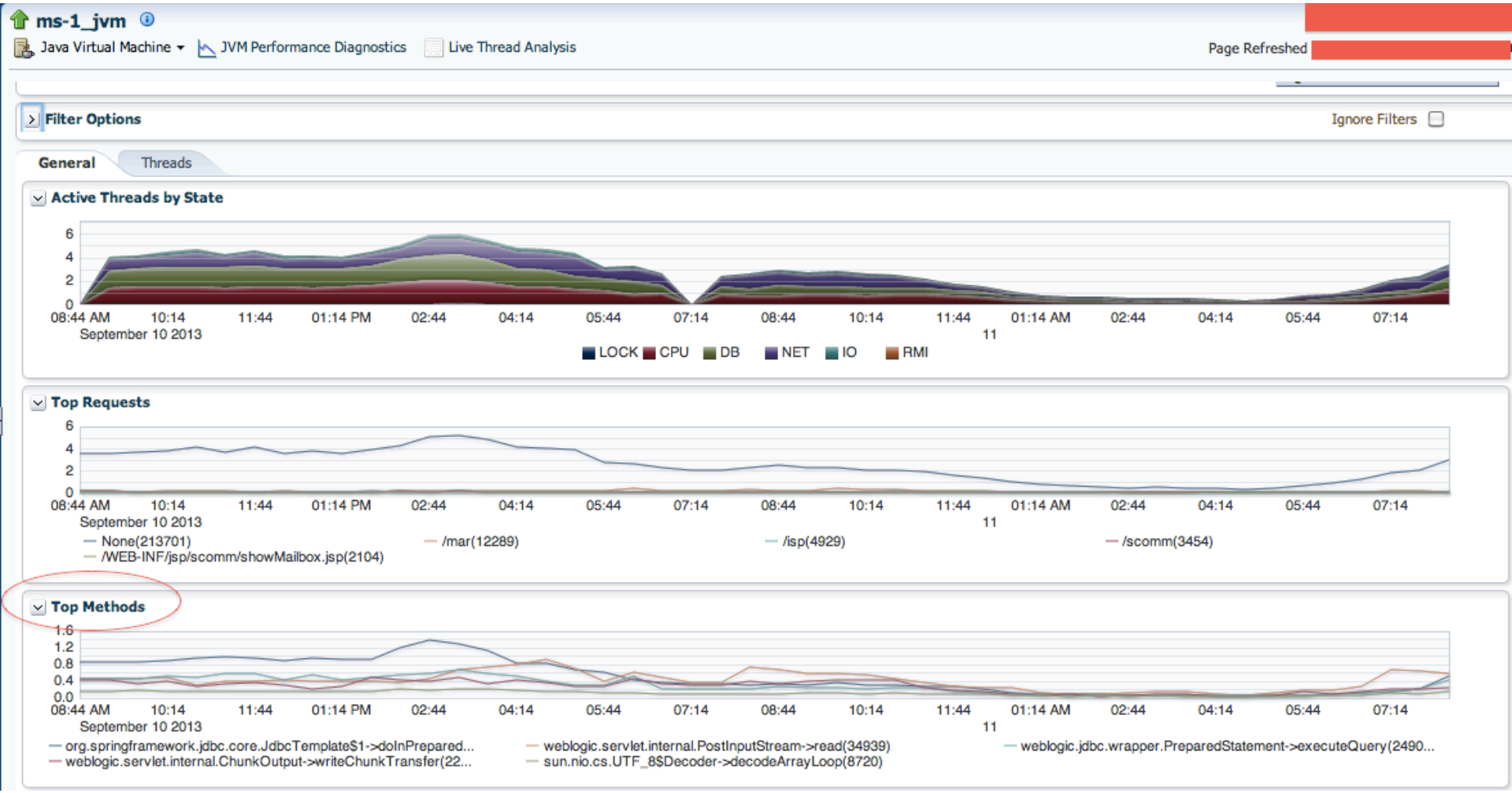
Logging Bottleneck

- 2000 logs/s – log4j with sync appender
- 4000 logs/s – logback with sync appender
- **8000 logs/s – logback with async appender**
- 12000 logs/s – log4j v2 with sync appender

Unexpected Top Method

- Noticed a JMS listener in the top method list
- In production!
- Did not show up during synthetic load testing
- We forgot to add a “message selector” on the listener

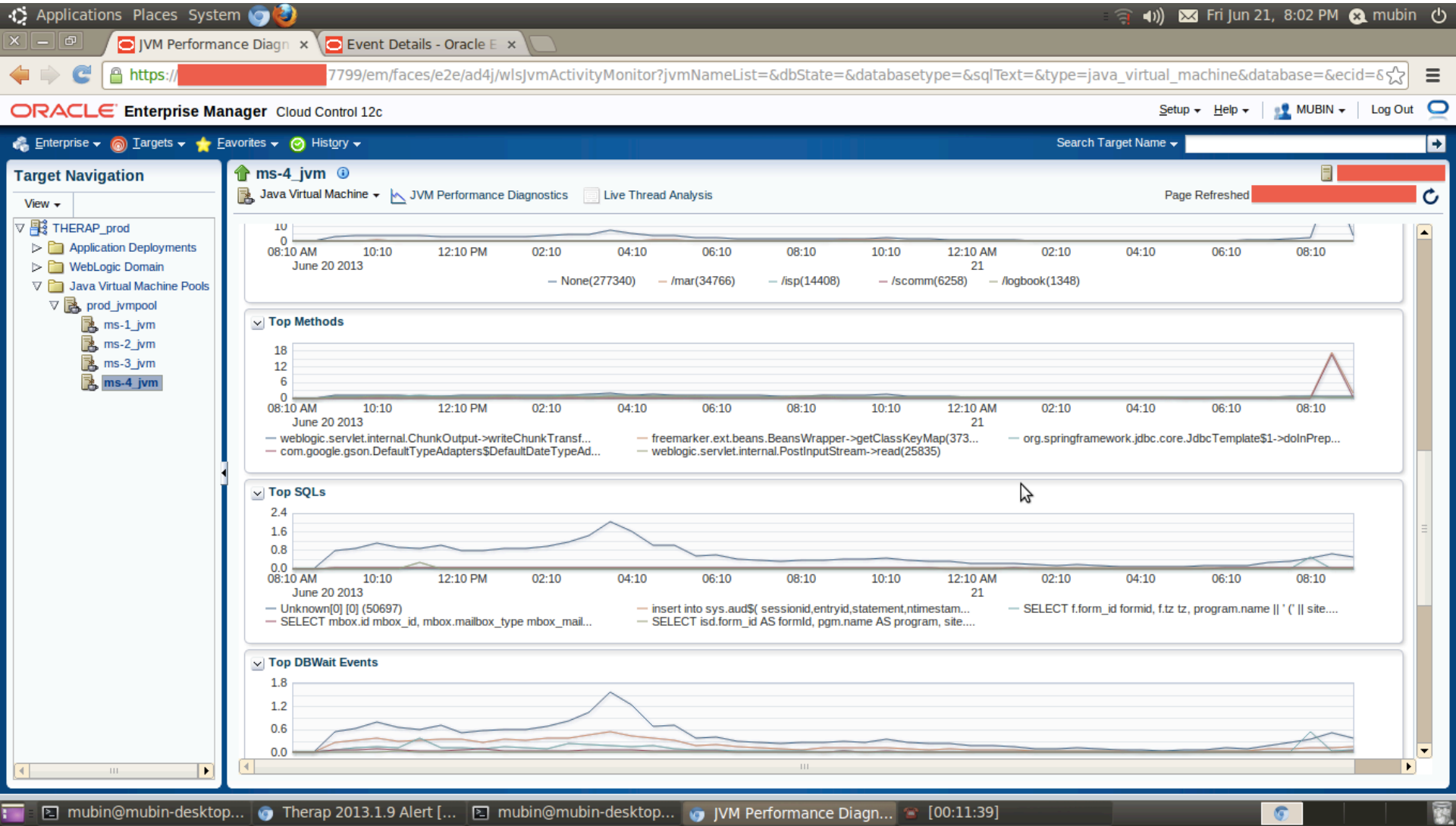
Top Method…


ms-1_jvm
Java Virtual Machine
JVM Performance Diagnostics
Live Thread Analysis
Page Refreshed
Filter Options
Ignore Filters
General
Threads
Active Threads by State
6
4
2
0
08:44 AM
10:14
11:44
01:14 PM
02:44
04:14
05:44
07:14
08:44
10:14
11:44
01:14 AM
02:44
04:14
05:44
07:14
September 10 2013
11
LOCK
CPU
DB
NET
IO
RMI
Top Requests
6
4
2
0
08:44 AM
10:14
11:44
01:14 PM
02:44
04:14
05:44
07:14
08:44
10:14
11:44
01:14 AM
02:44
04:14
05:44
07:14
September 10 2013
11
None(213701)
/mar(12289)
/isp(4929)
/scomm(3454)
/WEB-INF/jsp/scomm/showMailbox.jsp(2104)
Top Methods
1.6
1.2
0.8
0.4
0.0
08:44 AM
10:14
11:44
01:14 PM
02:44
04:14
05:44
07:14
08:44
10:14
11:44
01:14 AM
02:44
04:14
05:44
07:14
September 10 2013
11
org.springframework.jdbc.core.JdbcTemplate$1->doInPrepared...
weblogic.servlet.internal.PostInputStream->read(34939)
weblogic.jdbc.wrapper.PreparedStatement->executeQuery(2490...
weblogic.servlet.internal.ChunkOutput->writeChunkTransfer(22...
sun.nio.cs.UTF_8$Decoder->decodeArrayLoop(8720)

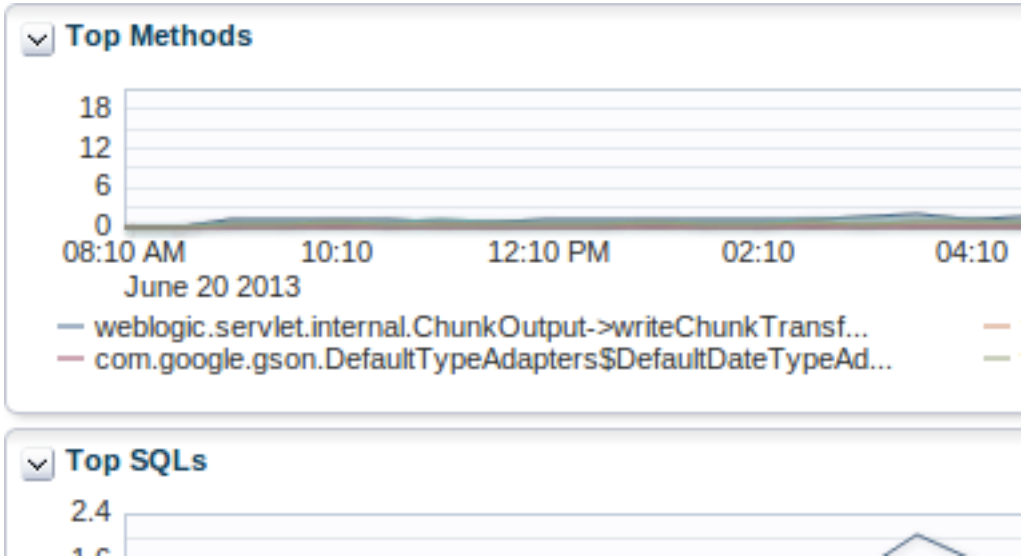
The Slow Library

- A library call for producing JSON showed on the top method list
- JSON is needed for AJAX
- It was totally unexpected
- The library was old and inefficient
- Replaced it with a newer and more efficient library

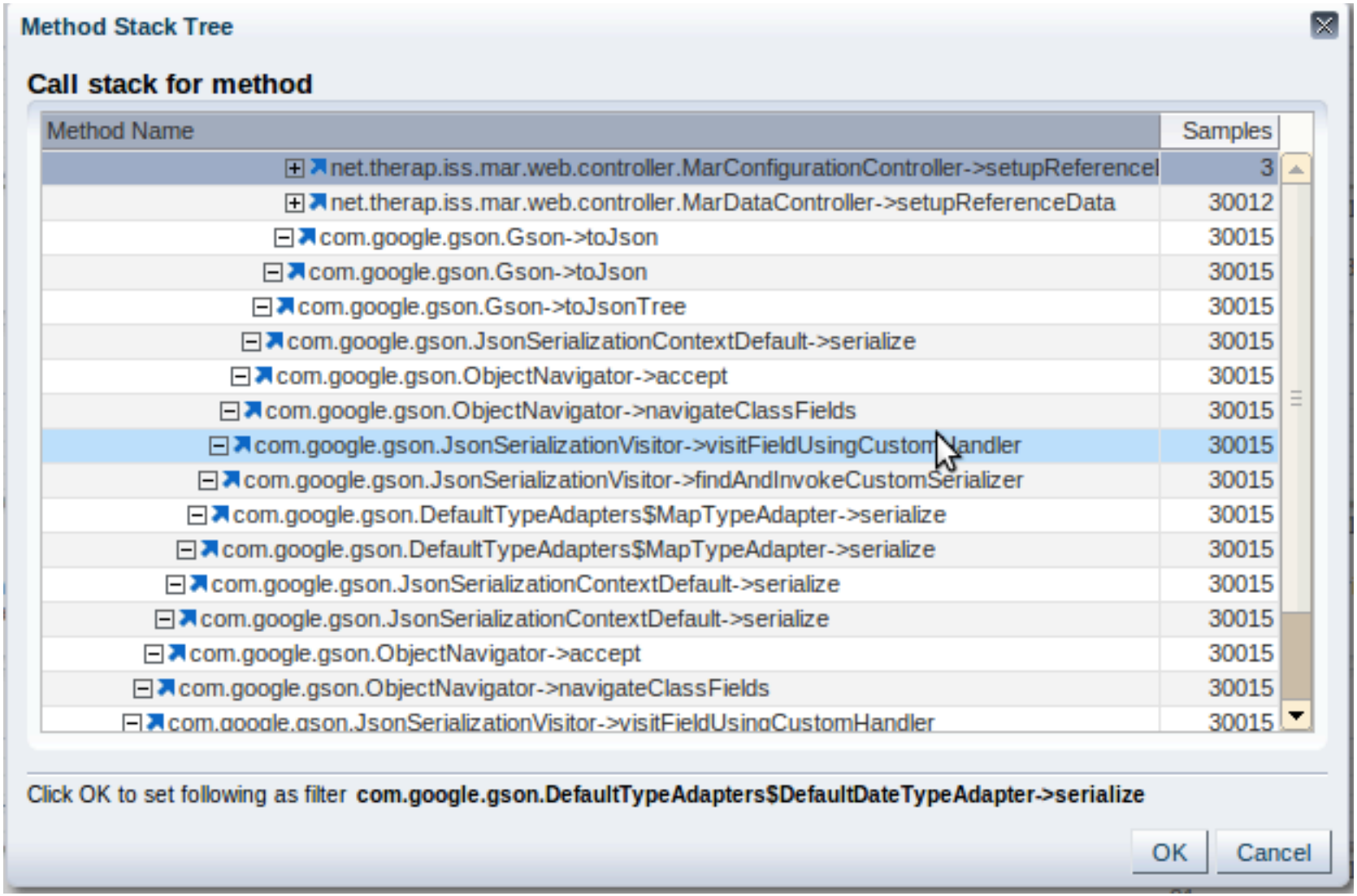
The Slow Library…



The Slow Library…



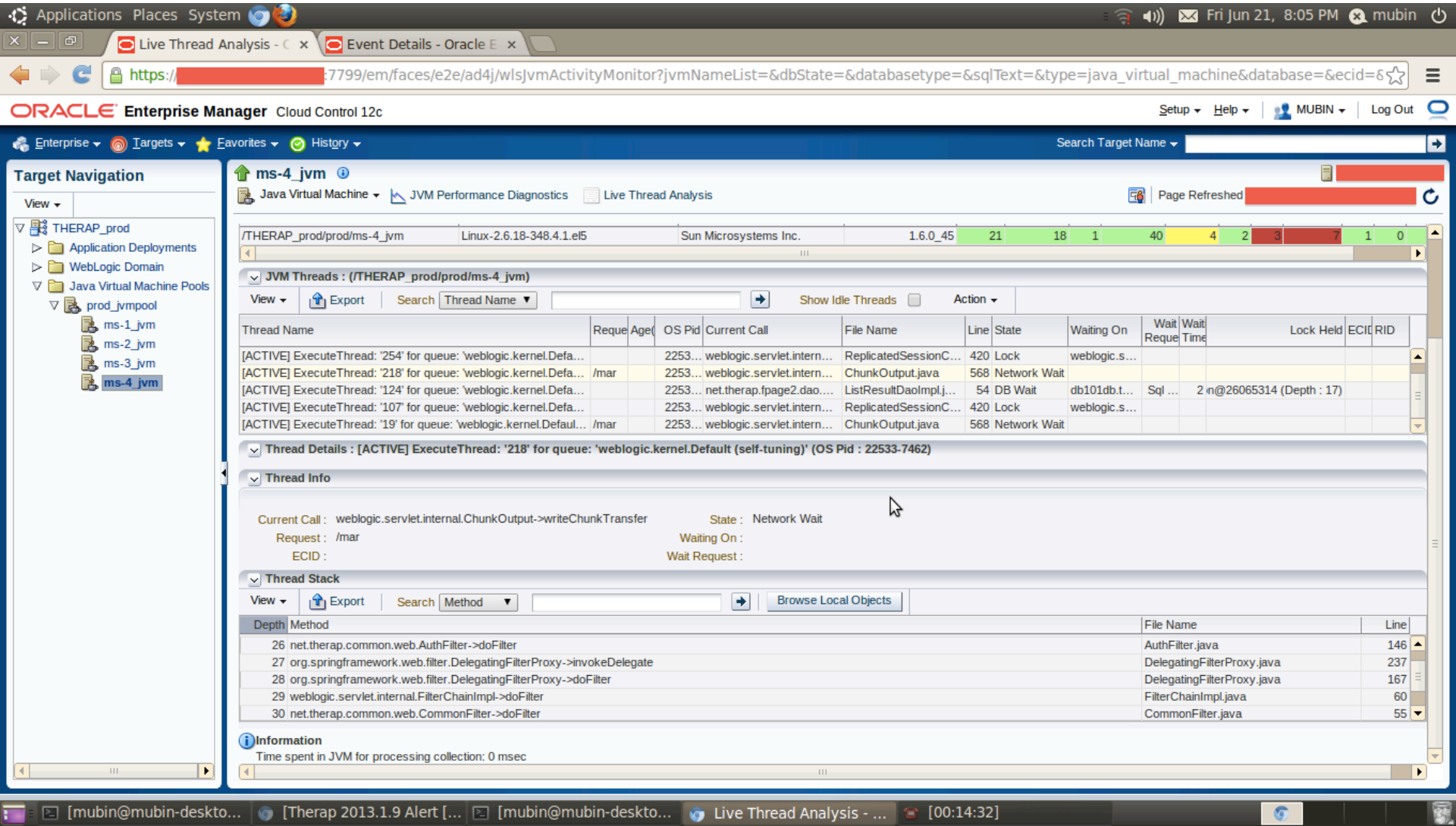
The Slow Library…



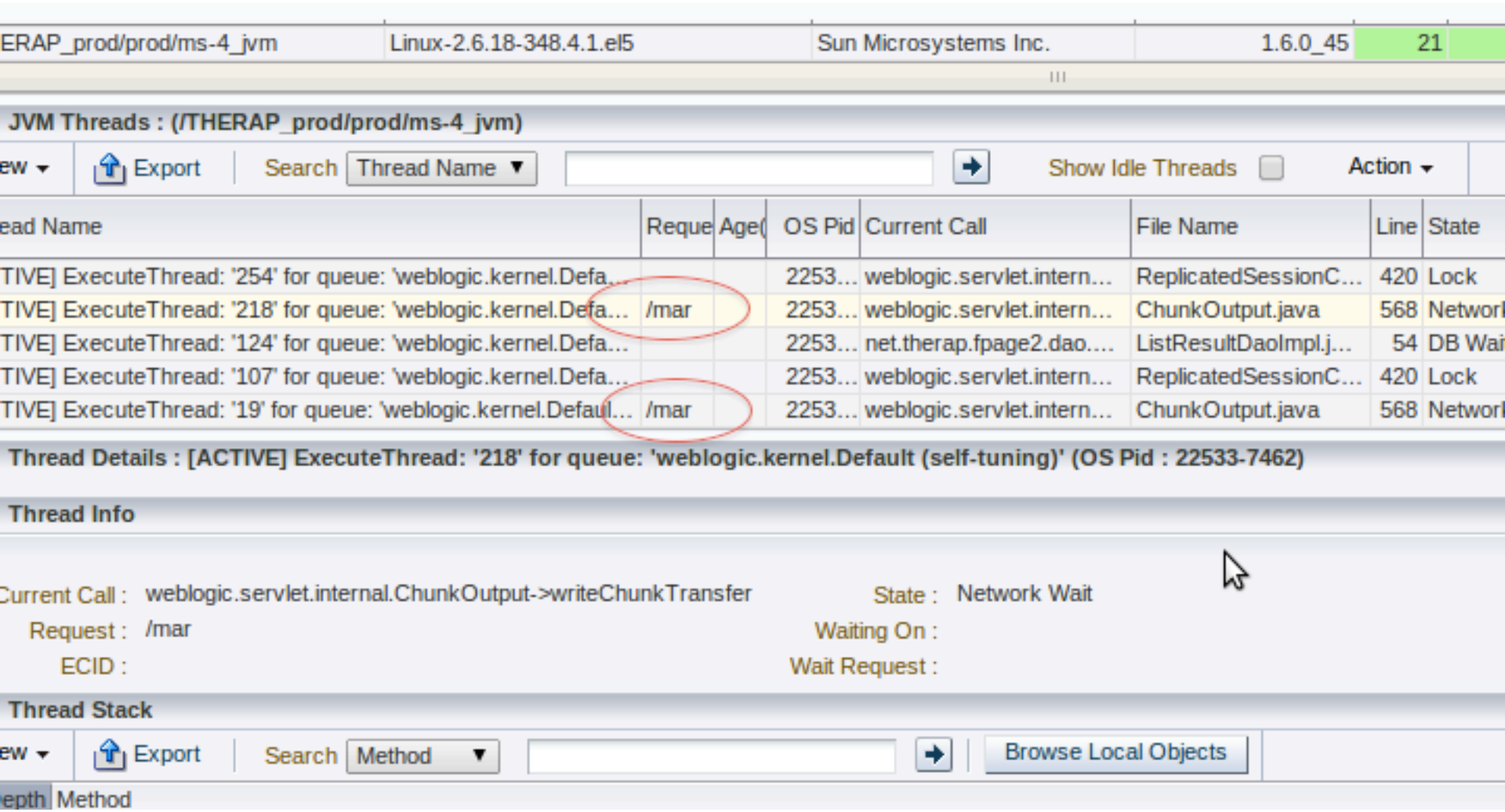
In-efficient Network Write

- Initially discovered in production through JVMD
- There were instances of high network waits
- Methods a certain module in the application showed up in the top list during the high network wait periods
- Discovered a 3 level loop that writes data
- Further inspection through JProfiler confirmed it

In-efficient Network Write…



In-efficient Network Write…



Automatic Thread Snapshots

- Previously, relied on **`kill -3`**
- Manual, missed dumps at crucial moments
- Now, JVMD takes thread snapshots when an abnormal thread state is reached on any WebLogic server
- Combined with auto-restart from WebLogic, eliminated unplanned downtime

Other APM Tools

- New Relic
  - Started with Ruby, covers Java well now
  - Cloud based
  - I recommend highly for small to medium systems
- App Dynamics
  - Probably somewhat more functional than New Relic
  - A lot easier than OEM to setup

Contact

masum ~ AT ~ therapservices.net

Thank You!