



# Oracle SOA Suite 11g Performance Tuning

Matt Brasier  
C2B2 Consulting



# Who am I?

- Head of consulting at C2B2
- 12 years experience with application servers
  - WebLogic 5.1 to present day
- Consulting with C2B2 since 2006
  - BEA Professional services
  - Red Hat Global Services
  - SME start ups to UK Govt
- Author of SOA Suite 11g performance tuning cookbook

# Objectives

- Provide techniques and methodology
- Applicable to any enterprise java application
- Key messages
  - Tune where there is a problem
  - Use the right tools

# Agenda

- Performance and SOA Suite
- A performance tuning methodology for SOA Suite
- Common SOA Suite performance problems
- Questions



Key challenges in SOA Suite performance

# PERFORMANCE AND SOA SUITE



# Why is performance important?

- Fundamental to many organisations SOA infrastructure
  - It needs to work invisibly and quietly
  - Plumbing
- Hard to test
  - Requirements vary significantly between applications
- Limited documentation



# Key Challenges

- Different components have different uses
  - Different customers use different components
  - There is no single answer to tuning such a wide range of products
- Technologies
  - BPM
  - BPEL
  - Mediator
  - BAM
  - OSB

- There is no one size fits all configuration
  - If there was, it would be default and there wouldn't be switches
- We can make some general recommendations though



# Fundamental Nature

- A message processing pipeline
  - Data size and quality has a large impact on performance
    - Frequent small messages vs infrequent large ones
- A Java application
  - Subject to garbage collection and other JVM features

# Goals

- Out of the box, Oracle SOA Suite is tuned for developer productivity.
- Tune to remove bottlenecks
  - Performance tuning where there is no bottleneck has no result
- Maximise performance and availability with existing infrastructure



How do we do this?

# A PERFORMANCE TUNING METHODOLOGY FOR SOA SUITE

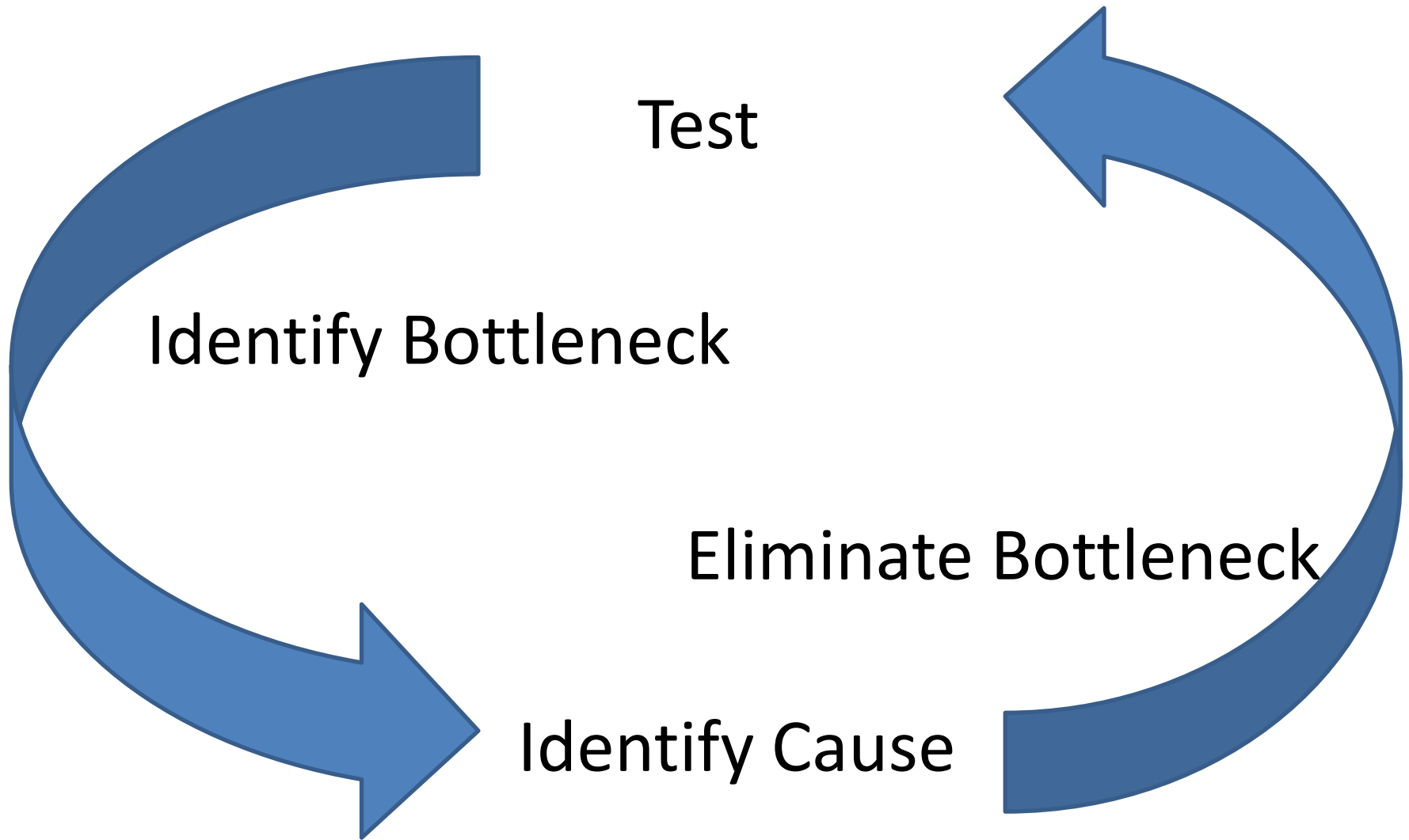


# Methodology

- Treat it the same as any other Java application



# A Science and an Art



# Testing

- Use a proper test harness
- Design users tests that accurately simulate user behaviour
  - All of it, at once
  - 10000 users logging in, then out of a site, over and over is not representative of your load
- Use a sufficiently powerful load generation capability
  - Don't test the performance of your load generator

# Monitoring

- Goes hand-in-hand with testing
- Testing is not a pass/fail activity
  - How close was the system to failing?
  - What happens if I double the load?
  - Which components are likely to fail first?
- Do not view your system as a black box

# Monitoring

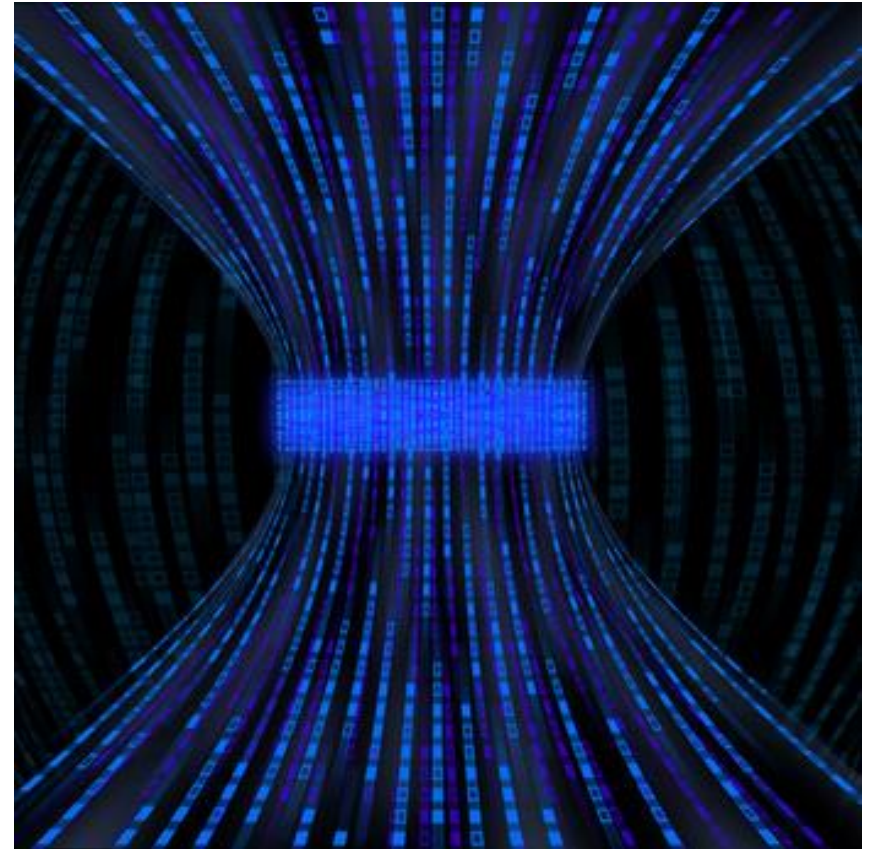
- A JMX based monitoring framework is essential
  - Memory stats (inc GC)
  - Connection pool stats
  - JMS queue stats
  - Transaction stats
  - Data
- Not just CPU% and Memory%





# Identify Bottleneck

- Use the monitoring as a guide
- What resource is constrained?
  - CPU
  - Memory
  - Locks
  - Bandwidth
  - Storage
  - Resource pools



# Identify Cause

- Now you know what is bottlenecked, but why?
  - Bad code
  - Sub-optimal setting
  - Data overload
  - Data structure
  - Insufficient resources

# Useful tools

- WebLogic console
- Enterprise Manager
- JStack
- JStat
- JVisualVM
- JRockit Mission Control (JRMC)



# Identify Solution

- The solution depends on the cause
  - Configuration changes
  - Code changes
  - More hardware!\*
  
- \* Beware anyone who starts out with this assumption



# Implement solution

- Implement the solution
  - In an environment where the problem could be reproduced
- Test again
  - Has the bottleneck gone away?
  - Is there a new bottleneck?

# Applying the methodology

- Bottom up
  - Work from the CPU upwards to find bottleneck
- Treat SOA Suite as any other Java application at this stage
  - Key suspects such as Garbage Collection and locking
- Stop when you are meeting your targets
  - You can always go faster
    - Subject to the speed of light in copper cable



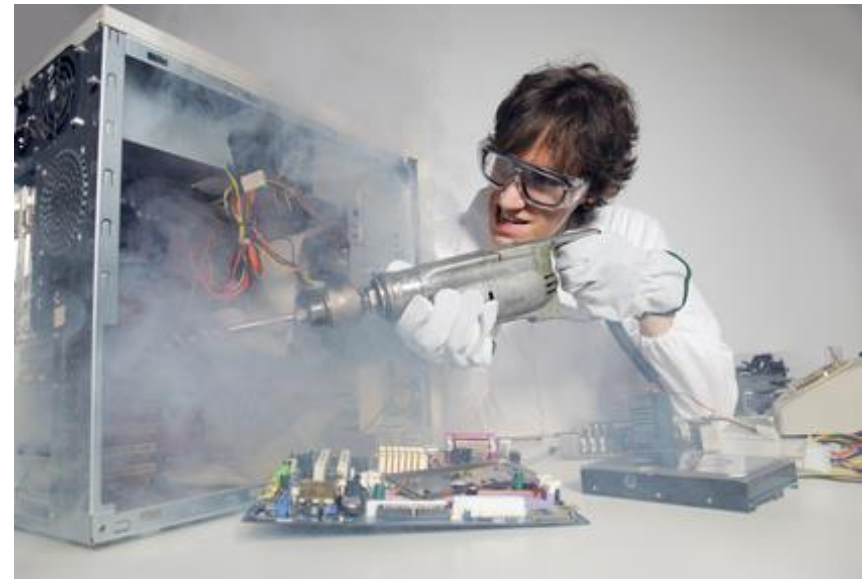
Where to start?

# COMMON SOA SUITE PERFORMANCE PROBLEMS



# 3 Common Problems

- Excessive garbage collection
- Thread pools
- Database access





# Garbage Collection

- SOA Suite applications vary in how much data they have to process and how fast



- However it is often the case that they process lots of XML
  - XML is object creation intensive
  - Results in garbage
- Garbage collection is based on the JVM
  - Hotspot/JRockit

# Garbage Collection

- Use tools like visualvm, jrmc or jstat to identify garbage collection problems
  - Garbage collection pause times of  $> 1s$  per minute
- Tune the garbage collector
  - Parallel is a good starting point
  - Reduce the number and duration of full GC pauses
- Tune the memory pools
  - Generational memory

# Tools

- Jstat
  - Jstat –gccause <pid>
- JVisualVM

# Thread Pools

- There are a lot of thread pools
  - Use a tool to see which thread pools are exhausted
    - JStack
    - JVisualVM
    - Etc
- WebLogic thread pool



# Thread Pools

- BPEL/BPMN thread pools
  - Dispatcher Invoke threads
    - Process requests from external sources
  - Dispatcher engine threads
    - Process asynchronous callback responses
  - Dispatcher System threads
    - Housekeeping tasks

# Database access

- Oracle SOA Suite makes heavy use of the database
  - Persistence of data
  - Metadata
  - Auditing and logging



# Database access

- Database access can be exponential with load
  - Each new user adds persistence load, auditing load, metadata lookups etc.
- Reduce the amount of database access to improve performance
  - Reduce auditing
  - Reduce persistence of in-flight processes



# Database access

- Reduce persistence
  - InMemoryOptimisation (transient processes)
  - Reduce audit threshold
  - Reduce completion persist level
  - Increase audit trail size threshold
  - Increase large document threshold
  
  - Many many more

# Database access

- Tune the database
  - Purge old BPEL processes
  - General database tuning



# Summary

- Treat SOA Suite as any other java EE application
  - One that processes a lot of XML
- Reduce database access
- Manage memory correctly
- Tune thread pools



You can't make it faster if it is not a bottleneck

# FOCUS TUNING ONLY WHERE IT IS REQUIRED

# Questions?

- Thankyou for your time



## Oracle SOA Suite 11g Performance Tuning Cookbook

Discover how you can get the best performance from your Oracle SOA Suite 11g infrastructure.

Matthew Brasier Nicholas Wright **[PACKT]** enterprise  
PUBLISHING professional expertise defined

[mbrasier@c2b2.co.uk](mailto:mbrasier@c2b2.co.uk) @c2b2consulting, @mbrasier, [www.c2b2.co.uk](http://www.c2b2.co.uk)