Session:

WebSphere Application Server for iSeries Performance Tuning & Analysis Tools

Version 2.0   8/31/2004

What is Performance?
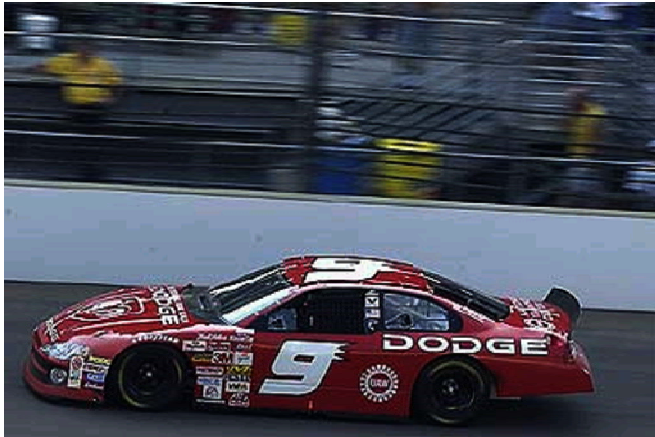
Performance is the measure of **time** and **resource consumption** incurred to complete a set of tasks.
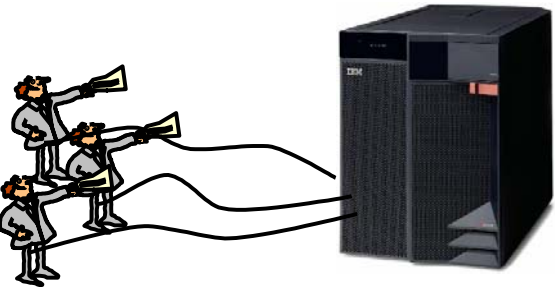
The goal is to minimize both of these, within **constraints** such as cost and available resources.

 Tradeoffs occur – costs, resources and laws of nature are inevitable challenges.
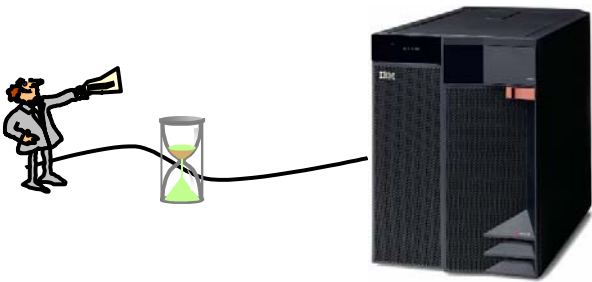
Performance is more complex to conceptualize in a web environment due to the many **cooperating and competing resources** involved.
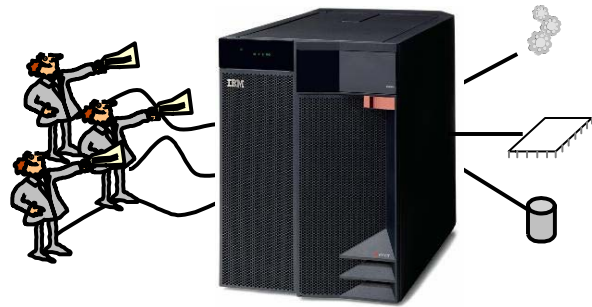
WebSphere sof

Performance Metrics

Throughput is the number of client requests a web environment can service at a given time.
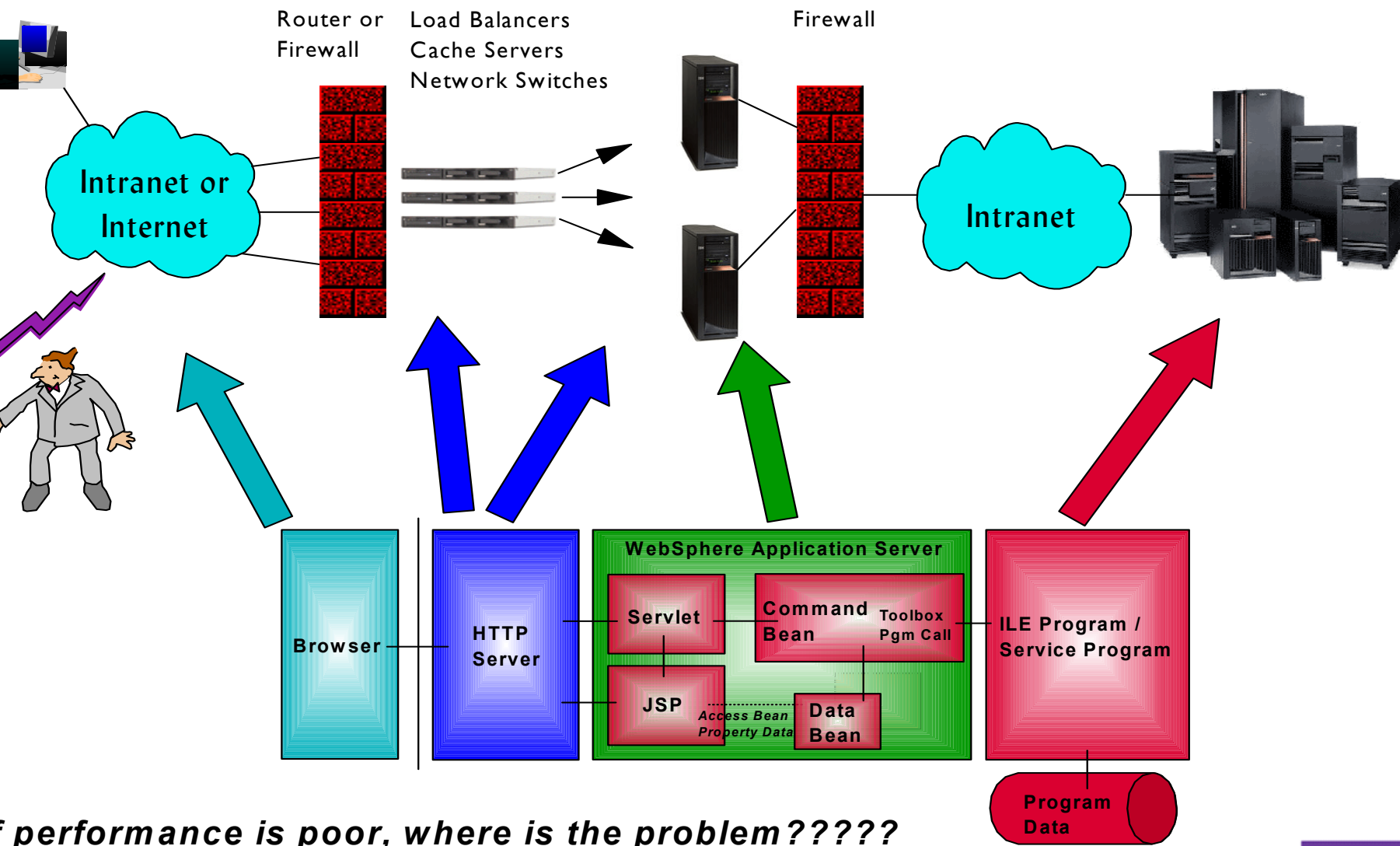
Response Time is elapsed time it takes to service a particular client's request.

Load is the amount of main storage, processor, I/O support, etc. needed to satisfy all client requests at a given time.

WebSphere sof

Typical e-Business Application Topology

Router or Firewall

Load Balancers
Cache Servers
Network Switches

Firewall

Intranet or Internet

Intranet

WebSphere Application Server

Browser

HTTP Server

Servlet

Command Bean

Toolbox Pgm Call

ILE Program / Service Program

JSP

Access Bean Property Data

Data Bean

Program Data

If performance is poor, where is the problem?????

WebSphere soft

Basic Performance Contributors

➡️*Processor Speed*
➡️*Memory*
➡️*Other Workload*

➡️*Processor, Memory, Disk, e...*
➡️*Operating System Settings*
➡️*Middleware (app server, etc...*
➡️*Application Design*
➡️*Data & Resource Access*

➡️*Bandwidth*
➡️*Latency*
➡️*Design*

■ **Server Environment** □ **Network/Communications** ■ **Client/Worksttion**

WebSphere soft
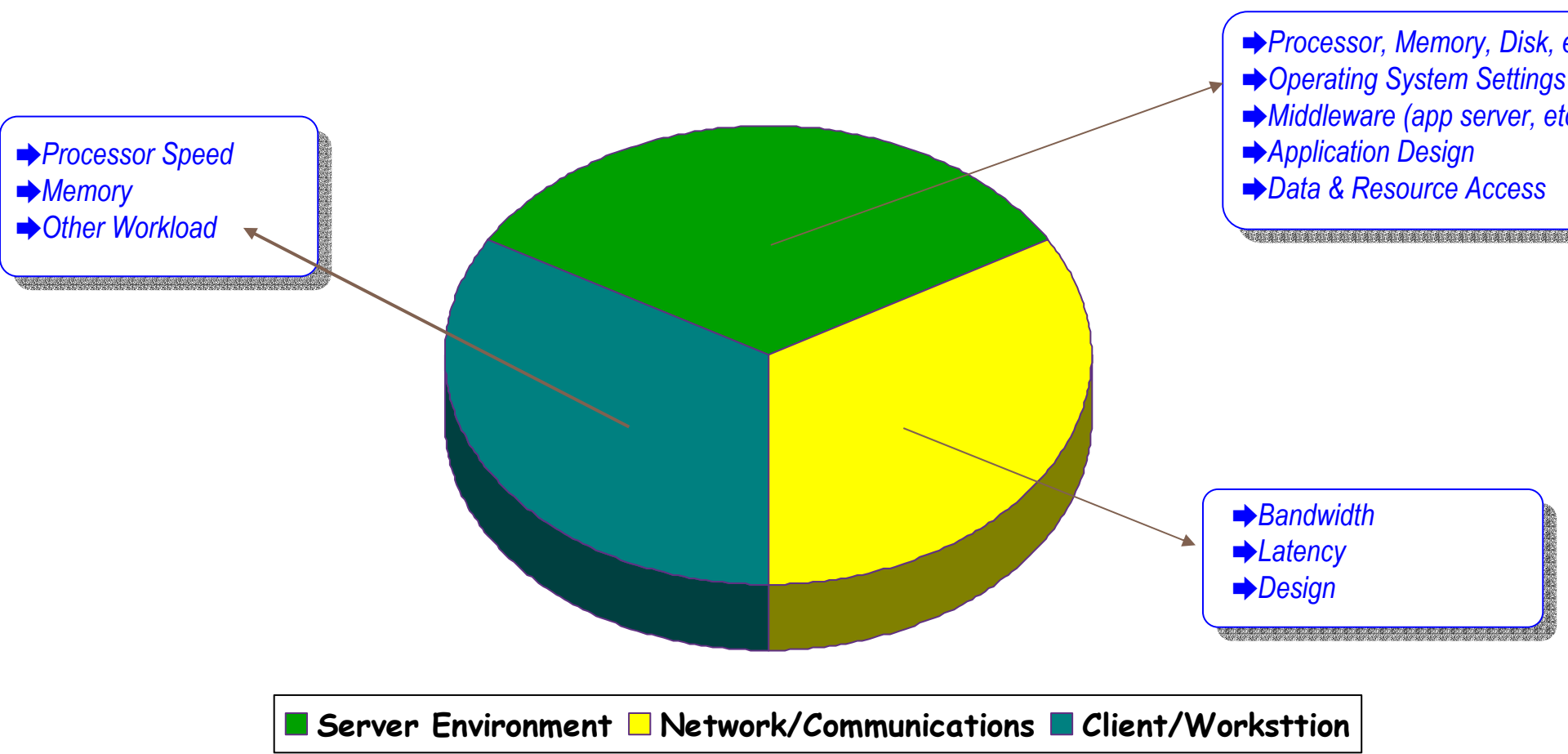
Server Performance Contributors

Processor, Memory, Disk, L2 cache
System Configuration
Memory Pools & Subsystems

Garbage Collector
Request Queuing
Data caching

Object Creation
Connection Pooling
Data Access

■ **Environment Config** □ **Application Implementation** ■ **Hardware Config**
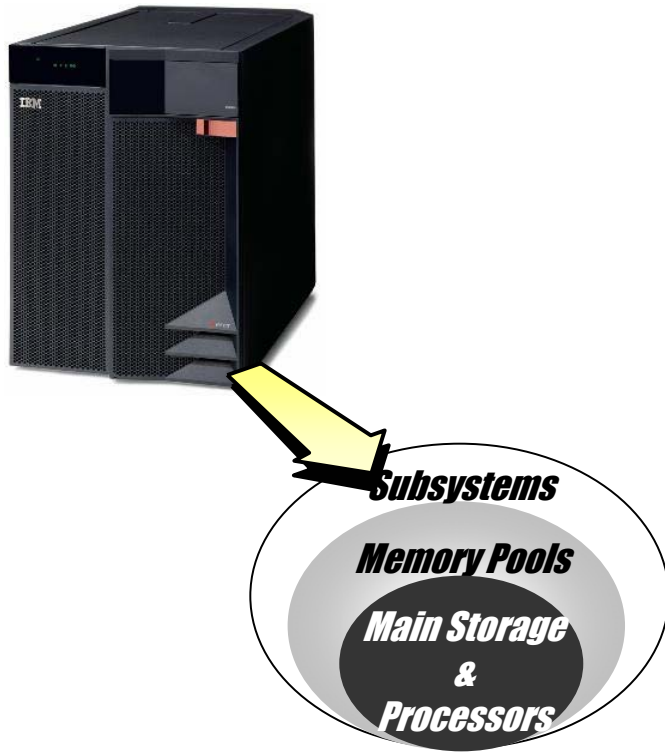
WebSphere soft
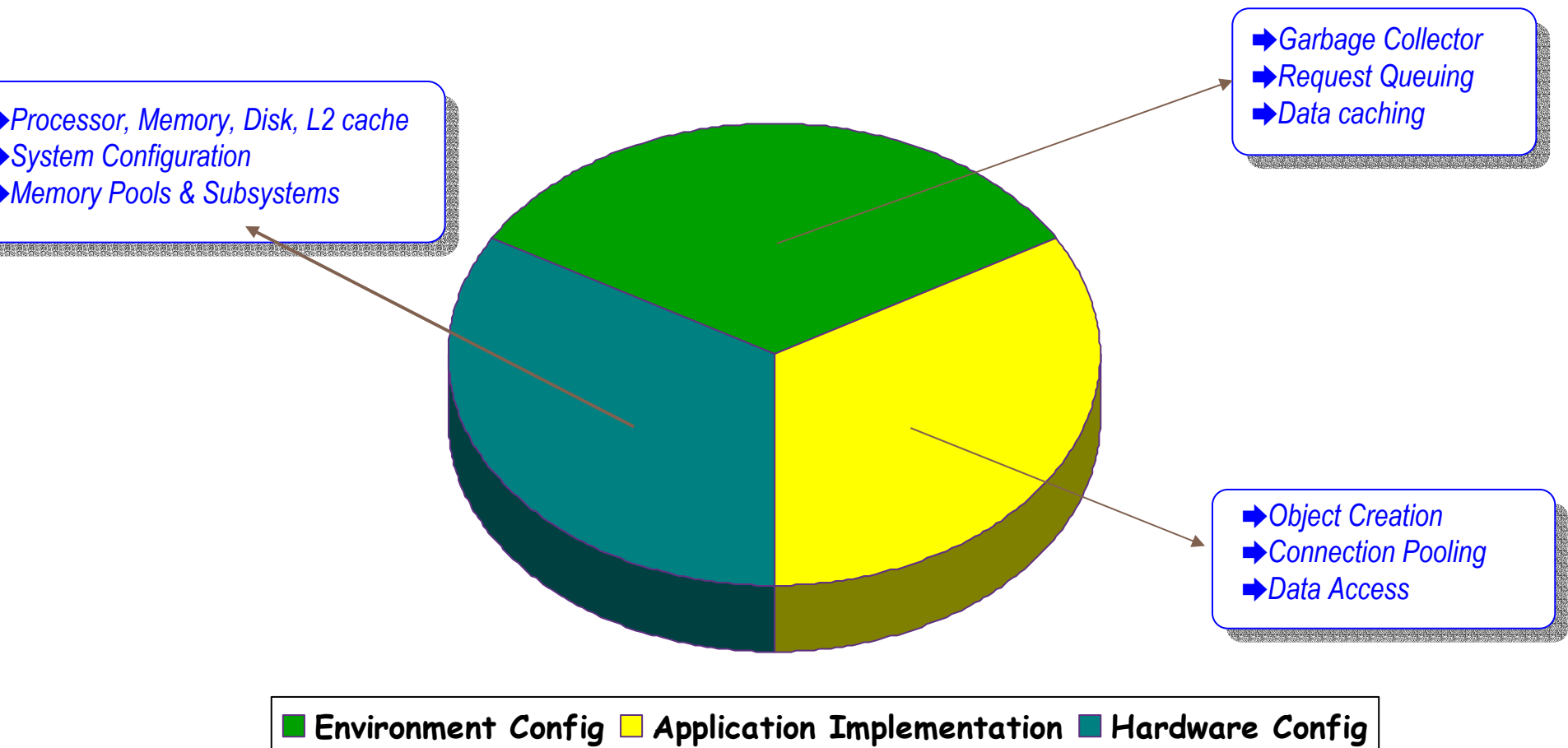
Server & System Performance

Application performance is affected by the amount and distribution of available iSeries resources.

Meeting performance requirements means:

(a) Do we have enough hardware resources installed?
(b) Have we allocated them efficiently?

Consult the **OS/400 Work Management Guide** in the iSeries Information Center for a work management overview.

Subsystems

Memory Pools

Main Storage & Processors

WebSphere sof

Server & System Performance
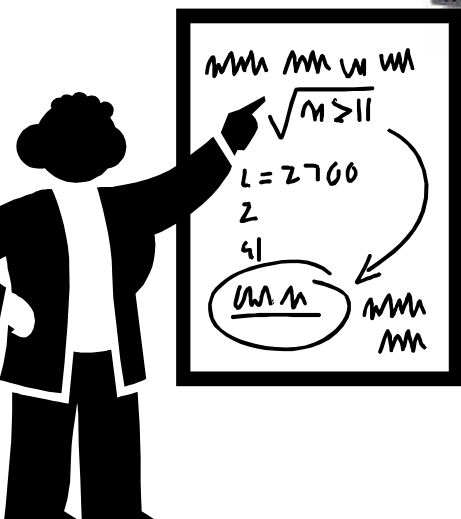
Key Concept - Make sure that you have enough system capacity.

Java & WAS applications require more processor and memory resources than traditional iSeries applications.

Ensure you have enough processor, memory, disk and other resources for Java/WAS and traditional workload on your system.

Newer iSeries servers and operating system releases have better performance than older servers and releases.

Use the **IBM Workload Estimator Tool (WLE)** or obtain professional services.

http://www-912.ibm.com/wle/EstimatorServlet

WebSphere sof

IBM Workload Estimator

The **IBM eServer Workload Estimator (WLE)** requires that you understand many aspects of the target iSeries environment, for example:

- Application design (Java™ Servlet, JSP™, and EJB™ complexity)
- Web traffic throughput (hits/second)
- RAID configuration

OS/400 Memory Pool Considerations

WebSphere Application Server - Express for iSeries

```
                Display Pool Definitions

System:
 Subsystem description:   QASE51        Status:   ACTIVE


Pool         Storage      Activity
 ID          Size (K)      Level
  1            *BASE
```

Activity level (of private pools)
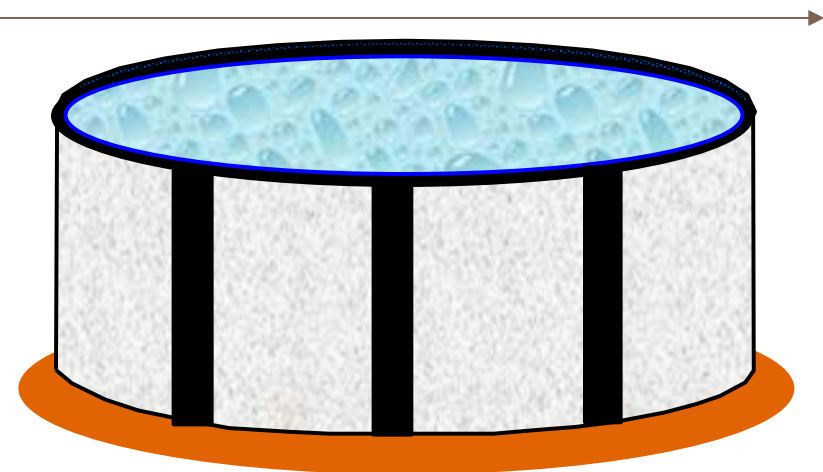
*Storage is allocated from the *BASE shared pool*

Subsystem description pool identifier

WebSphere Application Server v5 for iSeries

```
                Display Pool Definitions
                                             System:
 Subsystem description:   QEJBAS51      Status:   ACTIVE


Pool         Storage      Activity
 ID          Size (K)      Level
  1            *BASE
```

OS/400 subsystems use memory pools for allocating resources to their jobs.

WebSphere Application Server subsystems use the *BASE storage pool by default.

Private memory pool(s) can be defined exclusive for WAS usage, but you must evaluate the pro's and con's of doing this.

WebSphere sof

OS/400 Memory Pool Considerations

Data that is not contained in a memory pool (a ***page fault***) must be read from disk (***paging***), which can negatively affect performance

Paging has a larger performance impact on systems without enough disk arms.

The goal is for each memory pool to be large enough to keep paging at "reasonable" levels without severely impacting the performance of applications running in other pools.

Tuning can only partially compensate for inadequate memory resources on the server.

System Pool	Pool Size (M)	Reserved Size (M)	Max Active	-----DB----- Fault	Pages	---Non-DB--- Fault	Pages
1	507.58	272.60	+++++	.0	.0	9.0	9.0
2	10547.85	2.87	556	12.1	12.1	110.6	187.8
3	15.25	.00	5	.0	.0	.0	.0
4	897.61	.00	150	.0	.0	4.5	4.5

WebSphere soft

OS/400 Memory Pool Considerations

```
                    Display System Value

System value . . . . . :   QPFRADJ
Description  . . . . . :    Performance adjustment


Performance adjustment . . . :   2          0=No adjustment
                                            1=Adjustment at IPL
                                            2=Adjustment at IPL and automatic
                                             adjustment
                                            3=Automatic adjustment




Press Enter to continue.

F3=Exit   F12=Cancel
```

Memory pool size and activity level can be adjusted manually or automatically.

Automatic system performance adjustment is enabled by setting the **Performance Adjustment** system value (QPFRADJ). OS/400 attempts to allocate sufficient memory and activity level for executing storage pools.

Option (2) automatically adjusts at IPL and dynamically at regular intervals, however, this will overwrite any manual settings.

Option 3 'remembers' your last settings and uses them at IPL.

WebSphere sof

WebSphere Performance – Stay Current

|	OS/400 V5R1M0	OS/400 V5R2M0	OS/400 V5R3M0
WAS - Express 5.1 for iSeries	SF99273	SF99274	SF99275
WAS 5.1 for iSeries	SF99276	SF99277	SF99285
WAS Network Deployment 5.1 for iSeries	SF99278	SF99279	SF99286

Key Concept - Maintain currency with latest WebSphere Application Server (WAS) Versions and Group PTF levels.

WAS fixes for iSeries are delivered as Group PTFs, which include other iSeries Group PTF's (such as Java and HTTP server). They also frequently provide performance improvements.

Monitor the WebSphere for iSeries homepage for PTF availability and requirements.

http://www-912.ibm.com/s_dir/sline003.NSF/GroupPTFs?OpenView&view=GroupPTFs

WebSphere soft

Java Heap Size and Garbage Collection
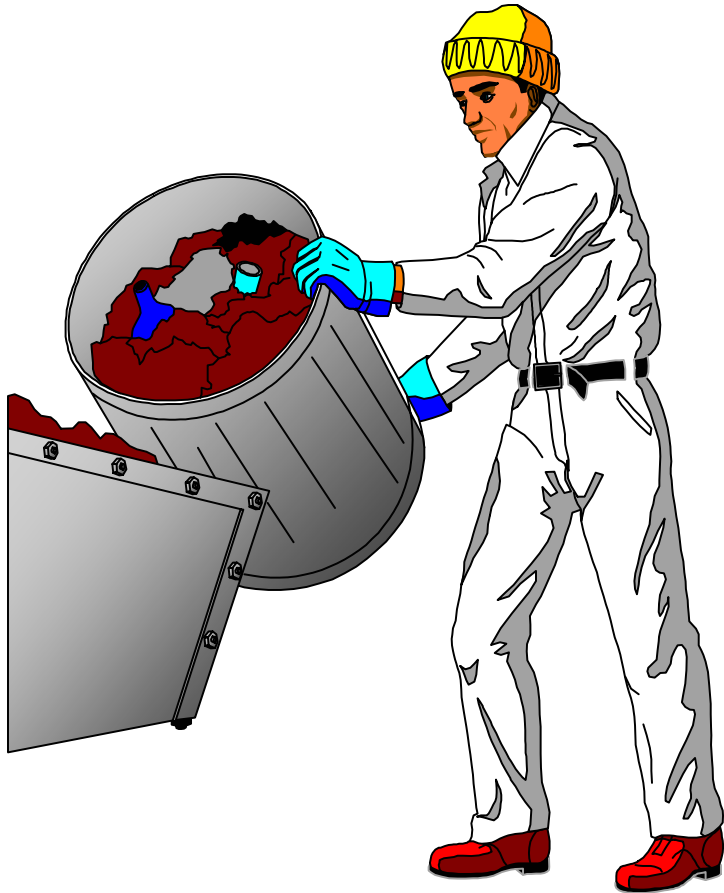
Java^{TM} objects are dynamically created in a portion of main storage allocated by the Java^{TM} Virtual Machine (JVM) called the Java^{TM} Heap. When all references to a particular object move out of scope, the object remains in the heap, taking up memory.

A **Garbage Collector** process periodically runs within the JVM to remove objects that are no longer referenced. Although OS/400 has a very efficient Garbage Collector, its execution can consume resources unnecessarily if not tuned properly.

WebSphere sof

Setting the Heap Size

- ## Set the initial GC size to:
  - 96MB for 1-2 way systems
    - The default for WebSphere
  - 256MB for 4-8 way systems
  - 768MB for 12 way and above
- ## Rule of Thumb: set initial GC size for a 3 Way and above to 64MB per processor, then increase as needed
- ## This is twice the recommended initial GC size for non-WebSphere applications.

pplication Servers > std001 > Process Definition >

Java Virtual Machine

Advanced Java virtual machine settings. [i]

**For more information on OS/400 on setting heap size visit:
www-1.ibm.com/servers/eserver/iseries/perfmgmt/pdf/tuninggc.pdf**

		verbose JNI.
Initial Heap Size	* 96	[i] Specifies the available to the J
Maximum Heap Size		[i] Specifies the available to the J default is 256.
Run HProf	☐	[i] Specifies wh

WebSphere sof

The HTTP Server (powered by Apache) local cache can store static content such as images and static web pages, in main storage. Configuration data includes the caching method, the files to cache, and the amount of main storage to use for the cache.

There are three caching methods:
- Copy an entire file into memory.
- Keep files descriptor open (for ASCI stream files only), reducing memory load.
- Memory map of file

What to cache?
- Setting the dynamic cache directive to "on" instructs the server to cache the most frequently accessed files.
- Specify if the cache is updated when a cached file is modified. Disabling this option is the optimum setting for performance.

System Resources ⑦

Highly Available Server **Denial of Service** **Advanced**
HTTP Connections **Caching** **FRCA File Cache**

Maximum cache storage size: 2000   Kilobytes ⌄   or... ⌄ ⑦

Files to cache when server is started: ⑦

	File path and name	Cache method
Example	*/www/welcome.html*	*Copy into memory*
Example	*/www/cached/*.html*	*Keep file descriptor open*
Example	*/www/icons/*.gif*	*Memory map of file*
⊙	/www/mydirectory/*.gif	Copy into memory ⌄

Copy into memory
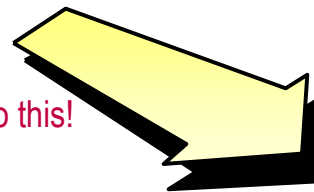Keep file descriptor open
Memory map of file

Add **Remove** **Move up** **Move down**

What to cache: ⑦

☐ Dynamically cache files based on file usage
☑ Update cache when files are modified

OK   **Apply**   **Cancel**

WebSphere sof
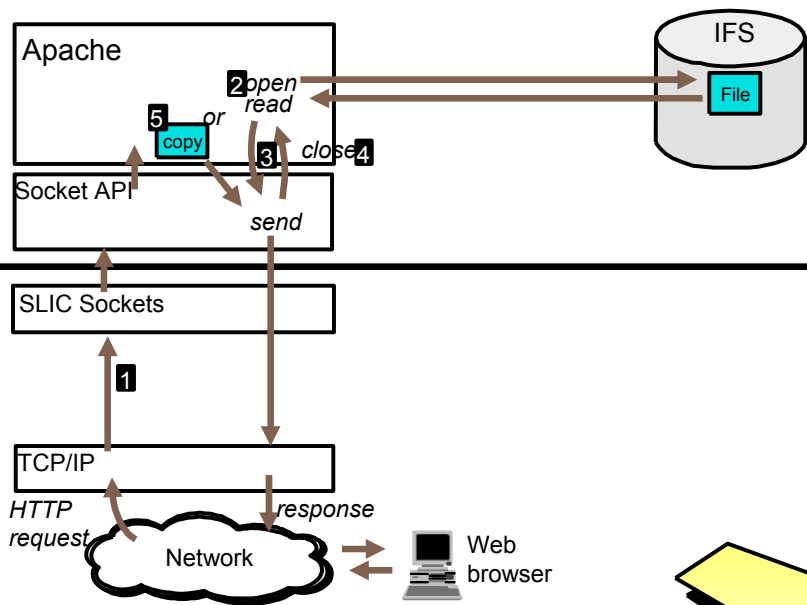
Fast Response Cache Accelerator (FRCA), available in OS/400 V5R2M0, provides a system API set and framework for socket applications. It is a caching function from within the Licensed Internal Code to reduce request processing time, thus accelerating file serving performance for the HTTP server.

Go from this to this!

The HTTP Server (powered by Apache) does not check for authorization on content served from FRCA. Use FRCA to cache content that does not need to be secured or accessed through specific validation. FRCA also does not perform code-page conversions. Use local caching for those requirements.

WebSphere sof
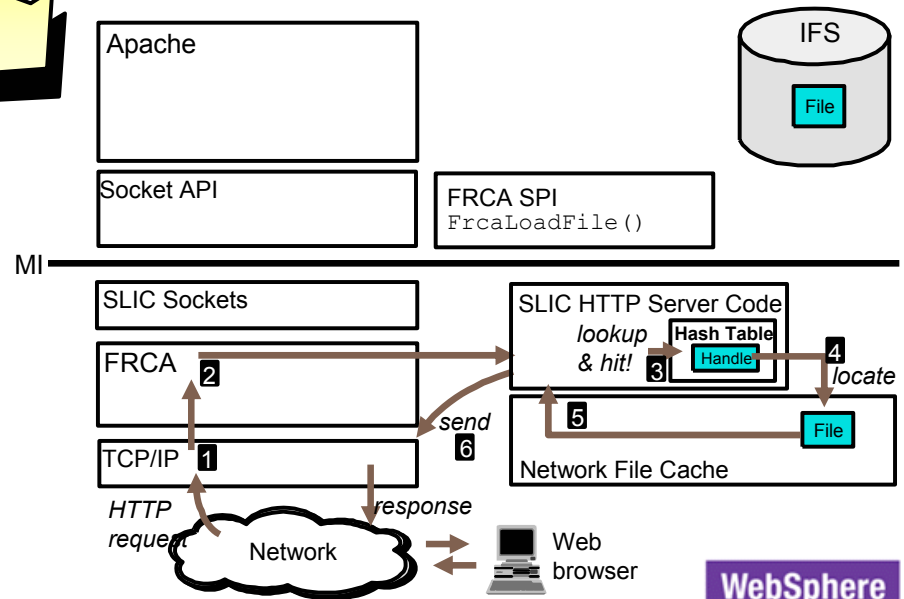
1. The Network file Cache, used by TCP/IP, is first configured using the CHGTCPA command

```
                    Change TCP/IP Attributes (CHGTCPA)

Type choices, press Enter.

IP time to live (hop limit)  . .   64              1-255, *SAME, *DFT
IP QoS enablement  . . . . . . .   *NO             *SAME, *TOS, *YES, *NO
IP QoS datagram batching . . . .   *NORMAL         *SAME, *NORMAL, *MINDELAY
IP QoS timer resolution  . . . .   100             5-5000, *SAME, *DFT
IP dead gateway detection:
  Enablement . . . . . . . . . .   *YES            *SAME, *DFT, *NO, *YES
   Interval . . . . . . . . . .   2               1-60
ARP cache timeout  . . . . . . .   15              1-1440, *SAME, *DFT
Network file cache:
  Enablement . . . . . . . . . .   *YES            *DFT, *CLEAR, *SAME, *YES, *NO
  Cached file timeout  . . . . .   300             *NOMAX, 30-604800 sec (1week)
  Cache size . . . . . . . . . .   10              10-100000 megabytes
Log protocol errors  . . . . . .   *NO             *SAME, *YES, *NO



                                                       Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```
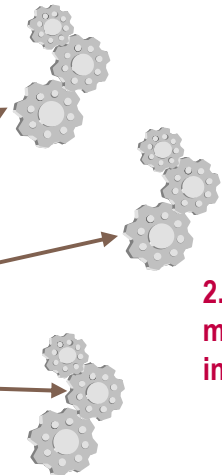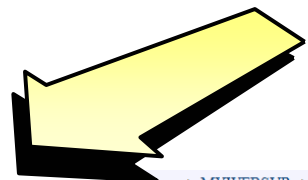
2. The size of this cache can be partitioned among multiple IBM HTTP Server (powered by Apache) instances.

3. FRCA is configured for a specific server instance using the IBM HTTP Server (powered by Apache) administrative instance.

All servers -> MYWEBSVR -> FRCA

FRCA

FRCA Logs   FRCA Reverse Proxy Cache
General Settings   FRCA File Cache

FRCA file cache capabilities: Enabled

Maximum cache size: 2000   Kilobytes
Maximum file size to cache: 90   Kilobytes

Files to cache during server startup:

File paths and names
Example /www/welcome.html
Example /www/cached/*.html
Example /www/icons/*.gif
⊙ /www/hello/*.gif

Add   Remove   Move up   Move down   Continue

Files to cache during server runtime:

File paths and names
Example /www/welcome.html
Example /www/cached/*.html
Example /www/icons/*.gif

Add

OK   Apply   Cancel

All servers -> MYWEBSVR -> FRCA

FRCA

FRCA Logs   FRCA Reverse Proxy Cache
General Settings   FRCA File Cache

Server IP addresses and ports to listen on:

	IP address		Port	FRCA
Example	All IP addresses		80	Disabled
⊙	192.168.1.1	or...	80	Enabled

Add   Remove   Move up   Move down   Continue

OK   Apply   Cancel

WebSphere sof

WebSphere ByteCode Verification Cache

- Unique to iSeries
- Improves application startup time by eliminating ByteCode verification and creation of temporary JVAPGM objects
- Most useful for very large applications such as WebSphere Commerce and Portal

Administrative Console

ogout | Help |

Application Servers > **WASEXPRESS** > **Process Definition** > **Java Virtual Machine** >

Custom Properties

Specifies arbitrary name/value pairs of data, where the name is a property key and the value is a string value which can be used to set internal system configuration properties. ℹ

Total: 3

⊞ Filter

⊞ Preferences

New | Delete

	Name ⇅	Value ⇅	Description ⇅
☐	os400.define.class.cache.file	/QIBM/ProdData/Java400/MyCache.jar	Class Loader Cache File for MyApp
☐	os400.define.class.cache.hours	500	
☐	os400.define.class.cache.maxpgms	500	

WebSphere sof

When Things Go Wrong or You Need Help

Key Concept - Use the appropriate tool(s) to analyze all relevant performance metrics.

Design and build performance into the WAS solution during development and test stages, not afterwards.

Establish system resource consumption baseline measurements to help plan for peak conditions and ensure adequate capacity will be available to handle future growth.

Find the 'hot spots' and focus your efforts most efficiently.

WebSphere sof

Tool	Where found….
OS/400 System Activity Commands	Included within OS/400 V5R2M0
verboseGC	Included within OS/400 V5R2M0
Dump Java^{TM} Virtual Machine (DMPJVM)	Included within OS/400 V5R2M0
Analyze Java^{TM} Virtual Machine (ANZJVM)	Included within OS/400 V5R2M0
OS/400 Collection Services	Included within OS/400 V5R2M0
PEX with Performance Trace Data Visualizer (PTDV)	Included within OS/400 V5R2M0 and free from IBM alphaWorks (www.alphaworks.ibm.com)
iDoctor	Free and fee-based form IBM eServer iSeries Support (http://www-912.ibm.com/i_dir/idoctor.nsf)
Tivoli Performance Monitor	Included in WebShere Application Server v5 for iSeries and WebSphere Application Server Network Deployment, v5 for iSeries
WebSphere Profiling Tool	WebSphere Development Studio Client V5 for iSeries

WebSphere sof

Work with Active Jobs (WRKACTJOB)	Allows you to review and change the attributes and resource utilization of the jobs running on your system.
Work with Disk Status (WRKDSKSTS)	Display the performance information and attributes for system disk units.
Work with System Status (WRKSYSSTS)	Provides an overview of current system activity. Specifically, it displays the number of jobs on the system and storage pool utilization information.
Work with System Activity WRKSYSACT	Work with jobs and tasks on your system. This command is part of the Performance Tools licensed program (PT1).
Work with Object Locks (WRKOBJLCK)	Work with and display locks on a specified object, including locks waiting to be applied
Work with Shared Storage Pools (WRKSHRPOOL)	Display the utilization information and change attributes of shared storage pools, including machine and base pool.

OS/400 includes a number of **System Activity Commands** that can allow you to perform real-time monitoring of performance data from the character-based interface. You can use these commands to answer specific questions about system performance and to help you tune your system.

WebSphere sof

verboseGC is a parameter to the JVM that provides a simple way to monitor garbage collector behavior, and check for object leaks. Enable it with the -verboseGC option on Java^{TM} command line, or within the WebSphere Application Server Administrative Console.
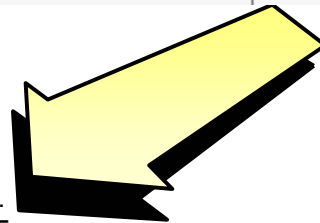
Application Servers > std001 > Process Definition >

Java Virtual Machine

Advanced Java virtual machine settings. ⓘ

		defau loadin
Verbose garbage collection	☑	ⓘ Sp debug defau collec
Verbose JNI	☐	ⓘ Sp debug

GC 5: starting collection, threshold allocation reached.
GC 5: live objects 31739457; collected objects 33663346; collected(KB) 4177772.
GC 5: queued for finalization 0; total soft references 622; cleared soft references 5.
GC 5: current heap(KB) 9066464; current threshold(KB) 2097152.
GC 5: collect (milliseconds) 9232.
GC 5: current cycle allocation(KB) 950219; previous cycle allocation(KB) 4194338.
GC 5: total weak references 3987; cleared weak references 0.
GC 5: total final references 118763; cleared final references 2267.
GC 5: total phantom references 0; cleared phantom references 0.
GC 5: total old soft references 0; cleared old soft references 0.
GC 5: total JNI global weak references 0; cleared JNI global weak references 0.

WebSphere sof

The Dump Java Virtual Machine (DMPJVM) command (intrusively) provides 'on-the-fly' information about the JVM per a specific job at a specific point in time - the classpath, garbage collection, and thread status.

Use DMPJVM to gain real time lock information to detect deadlocks and for garbage collection statistics to detect memory leaks!

- ●

```
                    Dump Java Virtual Machine (DMPJVM)

Type choices, press Enter.

Job name . . . . . . . . . . . > SERVER1      Name
  User . . . . . . . . . . . . >   QEJBSVR     Name
  Number . . . . . . . . . . .     025783      000000-999999
Stack frames . . . . . . . . .    10          0000-9999, *ALL




                                                              Bottom
F3=Exit   F4=Prompt   F5=Refresh   F10=Additional parameters   F12=Cancel
F13=How to use this display       F24=More keys
```

**Sphere
ication Server
D**

....
. Thread information
....
Information for 20 thread(s) of 20 thread(s) proces
Thread: 0000000E Thread-0
 TDE: B002100009149000
 Thread priority: 5
 Thread status: Destroy wait
 Thread group: main
 Runnable: java/lang/Thread
 Stack:
  None
 Locks:
  None
Thread: 0000001D Alarm : 0
 TDE: B00210000AB83000

. Garbage Collection
....
 Garbage collector
parameters
   Initial size: 32768 K
   Max size: 240000000 K
 Current values
   Heap size: 171232 K
   Garbage collections: 26
 Additional values
   JIT heap size: 10880 K
   JVM heap size: 62092 K
   Last GC cycle time: 0 ms

or 07 14:25:31 2003
rtual Machine Information 025783/QEJBSVR/SERVER1
....
path   .
....
rsion=1.3
ProdData/WebAS5/Base/java/ext/ibmorb.jar:/QIBM/ProdData/WebAS5/Base/j
/ibmext.jar:/QIBM/ProdData/OS400/Java400/jdk/lib/jdkptf13.zip:/QIBM/
ta:/OS400/Java400/jdk/lib/rt.jar:/QIBM/ProdData/OS400/Java400/jdk/lib/
:/QIBM/ProdData/OS400/Java400/jdk/lib/sunrsasign.jar:/QIBM/ProdData
/Java400/ext/IBMmisc.jar:/QIBM/ProdData/OS400/Java400/ext/jssl.jar:/Q
odData/OS400/Java400/ext/ibmjssl.jar:/QIBM/ProdData/OS400/Java400/:/Q
odData/Java400/:/QIBM/UserData/WebAS5/Base/default/properties:/QIBM/P
a/WebAS5/Base/properties:/QIBM/ProdData/WebAS5/Base/lib/bootstrap.jar
ProdData/WebAS5/Base/lib/j2ee.jar:/QIBM/ProdData/WebAS5/Base/lib/lmp

WebSphere soft

	General		System Data			JVM Data						WebSphere Data			
	Cross Platform	Intrusive Data Collection	Paging Rate	Disk Usage	CPU Usage	Garbage Collection Events	Heap Size/ Free Memory	Memory Usage by Object	Object Lock/ Unlock events	Object Method Elapsed Time	Stack Trace	Connection Pool Data	Web / ORB / EJB Pool Data	Session Data	Component (Web / EJB) Processing Data
OS/400 System Activity Commands			X	X	X										
verboseGC Parameter	X					X	X								
DMPJVM Command		X				X	X	X^2							
ANZJVM Command		X					X	X^2							
OS/400 Collection Services			X	X	X										
PEX with PTDV1		X				X			X	X	X^3				
iDoctor			X	X	X	X	X	X	X^5		X				
Tivoli Performance Viewer	X	X^4			X	X	X	X^2				X	X	X	X
WDSc Advanced Edition v5 for iSeries	X	X				X	X	X	X	X	X				

1 For profiling data, use PEX with Performance Tools/400 (57xx-PT1)

2 Provides object counts only - no per object memory usage reported

3 Provides stack depth to 5 levels only

4 The impact depends on the number of counters enabled, the type of counters enabled and the monitoring level set

for the counters.

5 Job Watcher provides some seize/lock analysis capabilities.

This is not an exhaustive list of the performance data returned by each of the identified tools.
It is only provided for high level comparitive purposes.

WebSphere sof

What tool to use?

- Use profiling tools to optimize the application during the development process.

- Start with OS/400 commands that return high-level data to narrow the scope of the performance problem area, then proceed with other tools to obtain low-level data.

- Use tools such as Collection Services and Tivoli Performance Viewer to monitor the day to day ups and downs, plus establish baseline data.

- Many low-level tools, such as PTDV, required advanced performance skills to managed and interpret the data returned.

- Consider tool platform support when working within a heterogeneous environment.

- Also consider the effect on application performance a particular tool has as it collects performance data.

Additional Resources

- WebSphere for iSeries performance homepage
http://www-1.ibm.com/servers/eserver/iseries/software/websphere/wsappserver/product/PerformanceConsiderations.html

- iSeries Information Center
http://publib.boulder.ibm.com/was400/51/english/index.htm?info/rzaiz/51/was.htm

- IBM Redbooks (www.redbooks.ibm.com)
  - Java & WebSphere Performance on iSeries (SG24-6256)
  - DB2 UDB & WebSphere V5 Performance & Tuning Guide (SG24-7068)
  - WebSphere V5 Performance, Scalability & High Availability (SG24-6198)
  - Measuring e-business Web Usage, Performance & Availability (SG24-6931)

- WebSphere, Java & DB2 for iSeries performance
http://www-1.ibm.com/servers/enable/education/i/index.html

WebSphere soft

Questions?

WebSphere sof