HP OpenView Smart Plug-in for WebSphere Application Server

For HP-UX and Solaris OpenView Operations Management Servers

Software Version: A.04.00

Reference

December 2005

Legal Notices

Warranty

Hewlett-Packard makes no warranty of any kind with regard to this document, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Hewlett-Packard shall not be held liable for errors contained herein or direct, indirect, special, incidental or consequential damages in connection with the furnishing, performance, or use of this material.

A copy of the specific warranty terms applicable to your Hewlett-Packard product can be obtained from your local Sales and Service Office.

Restricted Rights Legend

Copyright Notices

Trademark Notices

Support

Please visit the HP OpenView support web site at:

http://www.hp.com/managementsoftware/support

This web site provides contact information and details about the products, services, and support that HP OpenView offers.

HP OpenView online software support provides customer self-solve capabilities. It provides a fast and efficient way to access interactive technical support tools needed to manage your business. As a valuable support customer, you can benefit by using the support site to:

- Search for knowledge documents of interest

- Submit enhancement requests online

- Download software patches

- Submit and track progress on support cases

- Manage a support contract

- Look up HP support contacts

- Review information about available services

- Enter discussions with other software customers

- Research and register for software training

Most of the support areas require that you register as an HP Passport user and log in. Many also require a support contract.

To find more information about access levels, go to:

http://www.hp.com/managementsoftware/access_level

To register for an HP Passport ID, go to:

http://www.managementsoftware.hp.com/passport-registration.html

Contents

1 WBS-SPI Metrics

This chapter provides detailed and summary listings of the HP OpenView Smart Plug-in for WebSphere Application Server (WBS-SPI) metrics and provides descriptions and other relevant information that help you interpret the incoming WBS-SPI data. The information here should also prove valuable for customizing metric monitor templates. The chapter contains the following sections:

- Metric Summary
- Overhead Generated through Data Collection
- Metric Details (Names/Numbers, Specification Description)

Metric Summary

WBS-SPI metric monitor templates have pre-defined settings that simplify setup tasks for the WBS-SPI. Over time, however, you may want to customize some of those settings. This and the section that follows provide the basic pieces of information you need for those customizations.

For easy reference, the tables on the following pages list all metrics contained in the chapter. Following the metric summary table are individual metric details for every WBS-SPI metric and, when available, its monitor template settings. For metrics used for reporting or graphing only, no monitor settings exist, hence the setting is labeled "N/A" (not applicable).

Metric Summary Sheet: Columns Key

The summary list assists you in quickly finding a metric and its most basic information. Following the summary list are individual metric details which include settings for alarming metrics with parallel monitor templates.

ID (Metric Number)	The number assigned to the metric; for example 25 = I025. Numbers in the 200 range are for drill down metrics that collect values on a single instance of WBS; for example 225 = I225.
Metric Name	The metric name in non-abbreviated form; for example, EJBTranRbPct = EJB Transaction Rollback Percent.
Description	What the collected metric value represents.
Impact	Shows the generated overhead rating of the metric; possible ratings are: H = High M = Medium L = Low

Type	The purpose for which the metric is collected.
	A = Alarming
	R = Reporter reporting
	G = Reporter graphing
Severity	The severity of the exceeded threshold condition.
Area	The logical area to which the metric belongs.

Figure 1   .Summary WBS-SPI Metric Listing (page 1 of 4)

ID	Metric Name	Description	Impact	Type	Severity	Area
1	I001_ServerStatus	Status of a server	L	A	Critical	Availability
2	I002_ServerStatusRep	Status of a server - reporting	L	R		Availability
3	I003_AdminServerStat	Status of the Admin server	L	A	Critical	Availability
4	I004_AdminServerStatusRep	Status of the Admin server - reporting	L	R		Availability
5	I005_JVMMemUtilPct (WBS version 4.x)	% of heap space used in the JVM	L	A	Critical	JVM
5	I005_JVMMemUtilPct (WBS version 5)	% of heap space used in the JVM	H	A	Critical	JVM
210	I210_ThreadPoolActThreads	Ave # of active threads in a pool during collection interval	H	R		Performance
211	I211_ThreadPoolAveSize	Ave # of threads in a pool (active and idle) during collection interval	H	R		Performance
212	I212_ThreadPoolUtilPct	% of threads used in a pool collection interval	H	A	Critical Major Minor	Performance
13	I013_ThrdPoolPctMax	% of time # of threads in pool reached configured maximum size	H	G		Performance
213	I213_ThreadPoolPctMax	% of time # of threads in pool reached configured maximum size (drill down)	H	A	Minor	Performance
14	I014_ThrdPoolCrtRt	# of threads created per minute	L	G		Performance
20	I020_EJBPoolUtil	% of active beans in the pool	H	G		EJB
220	I220_EJBPoolUtil	% of active beans in the pool (drill down)	H	RA	Warning	EJB
221	I221_EJBMethRespTime	Ave response time of an EJB in msec	M	RA	Major Warning	EJB
22	I022_EJBMethCallsRt	# of EJB method calls per minute	L	GR		EJB
222	I222_EJBMethodCallsRt	# of EJB method calls per minute (drill down)	L	RA	Warning	EJB

Figure 2   Summary WBS-SPI Metric Listing (page 2 of 4)

ID	Metric Name	Description	Impact	Type	Severity	Area
223	I223_EJBPoolSize	Ave size of the EJB pool	H	R		EJB
24	I024_EJBEntDatLdStRt	# of times an EJB was written to or loaded from the database per minute	L	GR		EJB
224	I224_EJBEntDataLdStRt	# of times an EJB was written to or loaded from the database per minute (drill down)	L	RA	Warning	EJB
25	I025_EJBPoolMissPct	Ave % of time a call to retrieve an EJB from the pool failed	L	G		EJB
225	I225_EJBPoolMissPct	Ave % of time a call to retrieve an EJB from the pool failed (drill down)	L	R		EJB
26	I026_EJBConcLives	Ave # of bean objects in the pool	H	GA	Warning	EJB
40	I040_ServSessAveLife	Ave servlet session lifetime in msec	M	GA	Warning	Servlets
41	I041_ServSessActSess	# of sessions currently being accessed	H	GRA	Warning	Servlets
42	I042_ServInvSessRt	# of sessions being invalidated per second	L	GA	Warning	Servlets
45	I045_WebAppServReqRt	# of requests for a servlet per second	L	GR		Web Applications
245	I245_WebAppServletReqRt	# of requests for a servlet per second (drill down)	L	RA	Warning	Web Applications
246	I246_WebAppServletRespTime	Ave response time for a servlet in msec	M	RA	Major Warning	Web Applications
47	I047_WebAppServErrRt	# of errors in a servlet per second	L	G		Web Applications
247	I247_WebAppServletErrorRt	# of errors in a servlet per second (drill down)	L	A	Warning	Web Applications

Figure 3   Summary WBS-SPI Metric Listing (page 3 of 4)

ID	Metric Name	Description	Impact	Type	Severity	Area
48	I048_WebAppServLoad	# of servlets currently loaded for a web application	L	GA	Warning	Web Applications
49	I049_WebAppServRelRt	# of servlets reloaded for a web application per minute	L	G		Web Applications
260	I260_JDBCConnPoolSize	Ave # of connections in the connection pool	H	RA	Minor	JDBC
61	I061_JDBCConPoolWait	Ave # of threads waiting for a connection from connection pools	H	G		JDBC
261	I261_JDBCConnPoolWaiters	Ave # of threads waiting for a connection from connection pools (drill down)	H	RA	Major Warning	JDBC
62	I062_JDBConPoolWtTim (WBS version 4.x)	Ave time that a client waited for a connection in msec	M	G		JDBC
62	I062_JDBConPoolWtTim (WBS version 5)	Ave time that a client waited for a connection in msec	H	G		JDBC
262	I262_JDBCConnPoolWaitTime (WBS version 4.x)	Ave time that a client waited for a connection in msec (drill down)	M	RA	Major Warning	JDBC
262	I262_JDBCConnPoolWaitTime (WBS version 5)	Ave time that a client waited for a connection in msec (drill down)	H	RA	Major Warning	JDBC
263	I263_JDBCConnPoolUtil	% of connection pool in use	H	RA	Critical Major	JDBC
264	I264_JDBCConnPoolMaxPct	% of time that all connections in a pool are in use	H	A	Critical Major	JDBC
65	I065_JDBConPoolTimRt	# of times a client timed out waiting for a connection from the pool per minute	L	G		JDBC
265	I265_JDBCConnPoolTimeoutRt	# of times a client timed out waiting for a connection from the pool per minute (drill down)	L	RA	Critical Major	JDBC

Figure 4   Summary WBS-SPI Metric Listing (page 4 of 4)

ID	Metric Name	Description	Impact	Type	Severity	Area
66	I066_JDBConPoolThru	# of connections allocated and returned by applications per second	L	GR		JDBC
266	I266_JDBConnPoolThroughput	# of connections allocated and returned by applications	L	RA	Warning	JDBC
70	I070_TranGlobDur (WBS version 4.x)	Ave duration of global transactions	H	GA	Warning	Transactions
70	I070_TranGlobDur (WBS version 5)	Ave duration of global transactions	M	GA	Warning	Transactions
71	I071_TranLocDur (WBS version 4.x)	Ave duration of local transactions	H	GA	Warning	Transactions
71	I071_TranLocDur (WBS version 5)	Ave duration of local transactions	M	GA	Warning	Transactions
72	I072_TranGlobCommDur	Ave duration of commits for global transactions	M	GA	Warning	Transactions
73	I073_TranLocCommDur	Ave duration of commits for local transactions	M	GA	Warning	Transactions
74	I074_TranRollbackRt	# of global and local transactions rolled back per second	L	GA	Warning	Transactions
75	I075_TranTimeoutRt	# of global and local transactions that timed out per second	L	GA	Warning	Transactions
76	I076_TranCommitRt	# of global and local transactions that were committed per second	L	GA	Warning	Transactions
77	I077_TranThruput	# of global and local transactions that were completed per second	L	R		Transactions
78	I078_TranStartRt	# of global and local transactions that were begun per second	L	GA	Warning	Transactions

Overhead Generated Through Data Collection

All data collection affects performance in some way, with impact varying according to metric (counter). The overhead cost associated with each WBS-SPI metric is represented with a rating of low, medium, or high. A metric with a low rating involves only a minor performance cost since its calculation requires just a single addition or subtraction. Metrics with medium or high ratings have higher performance impacts. The calculations required for the collected data generally require multiplication, division, or both.

WBS-SPI Template Groups (organized according to performance impact): The WBS-SPI metric monitors are grouped for deployment in template groups according to the impact that their data collection has on system performance. WBS-SPI's three group are Low Impact, Medium Impact, and High Impact. The Low Impact group has only low impact metrics; while the Medium has both low and medium. High has all metrics, adding those with high impact levels to the low and medium groups.

Table 1   Low Impact Metrics

LOW IMPACT Metric Number	Metric Name	WebSphere Version	PMI Module
WBSSPI_0001	Server Status	4.x, 5.0	JMX MBean
WBSSPI_0002	Server Status Report	4.x, 5.0	JMX MBean
WBSSPI_0003	Admin Server Status	4.x, 5.0	WAS 4.x Specific
WBSSPI_0004	Admin Server Status Report	4.x, 5.0	WAS 4.x Specific
WBSSPI_0005	JVM Memory Utilization	4.x	jmvRuntimeModule
WBSSPI_0014	Thread Pool Created Rate	4.x, 5.0	threadPoolModule
WBSSPI_0022	EJB Method Calls Rate	4.x, 5.0	beanModule
WBSSPI_0222	EJB Method Calls Rate; Drill Down	4.x, 5.0	beanModule
WBSSPI_0024	EJB Data Loads/Stores Rate	4.x, 5.0	beanModule
WBSSPI_0224	EJB Data Loads/Stores Rate; Drill Down	4.x, 5.0	beanModule
WBSSPI_0025	Web Application Servlet Request Rate	4.x, 5.0	beanModule
WBSSPI_0225	Web Application Servlet Request Rate; Drill Down	4.x, 5.0	beanModule
WBSSPI_0042	Server Invalidated Session Rate	4.x, 5.0	servletSessionsModule
WBSSPI_0045	Web Application Servlet Request Rate	4.x, 5.0	webAppModule
WBSSPI_0245	Web Application Servlet Request Rate; Drill Down	4.x, 5.0	webAppModule
WBSSPI_0047	Web Application Servlet Error Rate	4.x, 5.0	webAppModule
WBSSPI_0247	Web Application Servlet Error Rate	4.x, 5.0	webAppModule
WBSSPI_0048	Web Application Servlet Load Rate	4.x, 5.0	webAppModule
WBSSPI_0049	Web Application Servlet Reload Rate	4.x, 5.0	webAppModule
WBSSPI_0065	JDBC Connection Pool Timeout Rate	4.x, 5.0	connectionPoolModule
WBSSPI_0265	JDBC Connection Pool Timeout Rate; Drill Down	4.x, 5.0	connectionPoolModule

Table 1   Low Impact Metrics (cont'd)

LOW IMPACT Metric Number	Metric Name	WebSphere Version	PMI Module
WBSSPI_0066	JDBC Connection Pool Throughput	4.x, 5.0	connectionPoolModule
WBSSPI_0266	JDBC Connection Pool Throughput; Drill Down	4.x, 5.0	connectionPoolModule
WBSSPI_0074	Transaction Rollback Rate	4.x, 5.0	transactionModule
WBSSPI_0075	Transaction Timeout Rate	4.x, 5.0	transactionModule
WBSSPI_0076	Transaction Commit Rate	4.x, 5.0	transactionModule
WBSSPI_0077	Transaction Throughput	4.x, 5.0	transactionModule
WBSSPI_0078	Transaction Start Rate	4.x, 5.0	transactionModule

Table 2   Medium Impact Metrics

MEDIUM IMPACT Metric Number	Metric Name	WebSphere Version	PMI Module
WBSSPI_0221	EJB Method Response Time	4.x, 5.0	beanModule
WBSSPI_0040	Servlet Session Average Life	4.x, 5.0	servletSessionsModule
WBSSPI_0246	Web Application Servlet Response Time	4.x, 5.0	webAppModule
WBSSPI_0062	JDB Connection Pool Wait Time	4.x	connectionPoolModule
WBSSPI_0262	JDB Connection Pool Wait Time; Drill Down	4.x	connectionPoolModule
WBSSPI_0070	Transaction Global Duration	5.0	transactionModule
WBSSPI_0071	Transaction Local Duration	5.0	transactionModule
WBSSPI_0072	Transaction Global Commit Duration	4.x, 5.0	transactionModule
WBSSPI_0073	Transaction Local Commit Duration	4.x, 5.0	transactionModule

Table 3   High Impact Metrics

HIGH IMPACT Metric Number	Metric Name	WebSphere Version	PMI Module
WBSSPI_0005	JVM Memory Utilization	5.0	jmvRuntimeModule
WBSSPI_0210	Thread Pool Active Threads	4.x, 5.0	threadPoolModule
WBSSPI_0211	Thread Pool Average Size	4.x, 5.0	threadPoolModule
WBSSPI_0212	Thread Pool Utilization Percentage	4.x, 5.0	threadPoolModule
WBSSPI_0013	Thread Pool Percentage Maximum	4.x, 5.0	threadPoolModule
WBSSPI_0213	Thread Pool Percentage Maximum; Drill Down	4.x, 5.0	threadPoolModule
WBSSPI_0020	EJB Pool Utilization	4.x, 5.0	beanModule
WBSSPI_0220	EJB Pool Utilization; Drill Down	4.x, 5.0	beanModule

Table 3   High Impact Metrics (cont'd)

HIGH IMPACT Metric Number	Metric Name	WebSphere Version	PMI Module
WBSSPI_0223	EJB Pool Size	4.x, 5.0	beanModule
WBSSPI_0026	EJB Concurrent Lives	4.x, 5.0	beanModule
WBSSPI_0041	Servlet Session Active Sessions	4.x, 5.0	servletSessionsModule
WBSSPI_0260	JDBC Connection Pool Size	4.x, 5.0	connectionPoolModule
WBSSPI_0061	JDBC Connection Pool Waiters	4.x, 5.0	connectionPoolModule
WBSSPI_0261	JDBC Connection Pool Waiters; Drill Down	4.x, 5.0	connectionPoolModule
WBSSPI_0062	JDB Connection Pool Wait Time	5.0	connectionPoolModule
WBSSPI_0262	JDB Connection Pool Wait Time; Drill Down	5.0	connectionPoolModule
WBSSPI_0263	JDBC Connection Pool Utilization	4.x, 5.0	connectionPoolModule
WBSSPI_0264	JDBC Connection Pool Percentage Maximum	4.x, 5.0	connectionPoolModule
WBSSPI_0070	Transaction Global Duration	4.x	transactionModule
WBSSPI_0071	Transaction Local Duration	4.x	transactionModule

PMI Modules Not Used

The following PMI modules are not used by the SPI (PMI module settings should be set to N, none, for these modules):

- cacheModule
- orbPerfModule
- systemModule
- webServicesModule

Metric Details

The WBS-SPI metrics are identified by a metric name/number. These numbers also appear in the monitor templates and Application Bank (ASCII) reports (if either exists for the parallel metric). The naming/numbering conventions are as follows:

- **metric names/numbers**: The "I" preceding each metric number designates the metric as a WBS-SPI metric. WBS-SPI metrics can then be identified as I*XXX*, where *XXX* represents the number assigned to the metric; for example, I005.

- **metric number ranges**: WBS-SPI numbers range from 0000 to 0999 with ranges covering metrics as follows:

  1 - 4 —Availability
  10-19 —Server Performance
  20-39 —Enterprise Java Beans
  40-44 —Servlets
  45-49 —Web Application
  60-69 —JDBC
  70-79 — transaction

  In addition, for metrics defined by the user (user defined metrics), the 1000 to 1999 range is reserved.

- **Application Bank report names**: If available for a specific WBS-SPI metric, the report name is the metric number followed by an underscore and the abbreviated metric name; for example, I0005_JVMMemUtilPct.

- **monitor template names**: If a monitor template is available for a metric, the monitor template name omits the "I" and begins with WBSSPI followed by an underscore and the metric number. Zeros are used as necessary to total a four-digit number; for example, metric number I005 = monitor template WBSSPI_0005

Monitor Template Name	Begins always with "WBSSPI," followed by the metric number. Within the monitor template you can change settings as described in the definition; for example, threshold value, severity, etc.
Metric Name	The name assigned to the metric.
Metric Type	Shows how the metric is used, whether for: Alarming (using monitor template settings), Reporting (within a report of the separately purchased HP OpenView Reporter), and/or Graphing (within a graph of the separately purchased HP OpenView Performance Manager).
Description	What the metric represents.
Impact	Overhead cost rating: H (High): Data counters (metrics) with the highest impact on system performance. M (Medium): Data counters with moderate impact on system performance. L (Low): Data counters with nominal impact on system performance.
PMI Module	PMI module mapped to the metric.
Severity: Condition	The severity of the exceeded threshold condition. (Critical, Major, Minor, Warning, Normal). If multiple conditions—for example, graduated thresholds—are defined within the metric, severity levels are identified according to the specific condition.
Collection Interval	How often the metric is collected and analyzed. (5 min, 15 min, 1 hour, 1 time daily)
OVO Min/Max Threshold	Because this setting is the same for all WBS-SPI metrics, which have maximum thresholds, it is omitted.
Default OVO Threshold	Shows the default OVO threshold for metrics with parallel monitor templates. (*=Metrics that should have been assigned a threshold value of 0 are set at 0.5 because OVO alarms must occur at <= or >= values. Since a 0 value would always trigger an alarm, the threshold is set to 0.5)
OVO Threshold Type	Because this setting is the same for all WBS-SPI metrics, which are without reset, it is omitted.
Message Group	To what OVO message group does the metric belong? (WBSSPI = conditions occurring in the WBS-SPI; and WebSphere = conditions occurring in the WebSphere application server)
Message Text	The message displayed for each condition.
Instruction Text	Problem-solving information (Probable causes, Potential impact, Suggested actions, and Reports)
Application Bank Report	Indicates whether or not an ASCII report is available and whether or not an Automatic Action is associated with it. Note: All the reports that are automatic actions or operator actions are also in the OVO Application Bank. However, metrics that are OpenView Performance Agent (also known as MeasureWare Agent) metrics (no alarms) do not have an OVO template for Operator or Auto actions, so they are ONLY in the Application Bank. N/A means that no report is planned.
Area	The logical area to which the metric belongs. (Availability, JVM, Performance, Servlets, EJB, Servlets, Web Applications, J2C, JDBC, Transactions)

Metric I001_ServerStatus

Monitor Template Name	WBSSPI_0001
Metric Name	I001_ServerStatus
Metric Type	Alarming
Description	Status of a server, monitors whether running or not.
Impact	Low (WebSphere version 4.x, 5)
PMI Module	JMX MBean
Severity: Condition with threshold	WBSSPI-0001.1: Critical threshold, 4.5
Collection Interval	5m
Message Group	WebSphere
Message Text	WBSSPI-0001.1: Server status is down
Instruction Text	
Application Bank Report	N/A
Area	Availability

Metric I002_ServerStatusRep

Monitor Template Name	N/A—Used to generate a report
Metric Name	I002_ServerStatusRep
Metric Type	Reporting
Description	Status of server—reporting
Impact	Low (WebSphere version 4.x, 5)
PMI Module	JMX MBean
Severity: Condition with threshold	N/A
Collection Interval	5m
Message Text	N/A
Instruction Text	N/A
Application Bank Report	N/A
Area	Availability

Metric I003_AdminServer Stat

Monitor Template Name	WBSSPI_0003
Metric Name	I003_AdminServerStat
Metric Type	Alarming
Description	Status of the Admin server.
Impact	Low (WebSphere version 4.x, 5)
PMI Module	WAS 4.x Specific
Severity: Condition with threshold	WBSSPI-0003.1: Critical threshold, 4.5
Collection Interval	5m
Message Group	WebSphere
Message Text	WBSSPI-0003.1: Admin Server status is down
Instruction Text	
Application Bank Report	No
Area	Availability

Metric I004_AdminServerStatusRep

Monitor Template Name	N/A—Used to generate a report
Metric Name	I004_AdminServerStatusRep
Metric Type	Reporting
Description	Status of the Admin server - reporting
Impact	Low (WebSphere version 4.x, 5)
PMI Module	WAS 4.x Specific
Severity: Condition with threshold	N/A
Collection Interval	5m
Message Group	WebSphere
Message Text	N/A
Instruction Text	N/A
Application Bank Report	N/A
Area	Availability

Metric I005_JVMMemUtilPct

Monitor Template Name	WBSSPI_0005
Metric Name	I005_JVMMemUtilPct
Metric Type	Alarming
Description	Percentage of heap space used in the JVM.
Impact	Low (WebSphere version 4.x); High (WebSphere version 5)
PMI Module	jvmRuntimeModule
Severity: Condition with threshold	WBSSPI-0005.1: Critical threshold, 98 WBSSPI-0005.2: Major threshold, 95
Collection Interval	15m
Message Group	WebSphere
Message Text	WBSSPI-0005.1:% of heap space used (<$VALUE>%) too high (>=<$THRESHOLD>%)

Instruction Text	The JVM is running out of available heap space.
	Suggested action: Open the Performance Tuner Wizard by clicking Console → Wizards → Performance Tuner from the Administrative Console.
	<u>Java Virtual Machine (JVM) Heap Size</u>
	The Java Virtual Machine (JVM) Heap Size settings influence garbage collection of Java objects. If you increase the heap size, garbage collection occurs less frequently, but takes longer. These settings depend strongly on your application and on the amount of physical memory available. Consider:
	• whether the JVM Heap for the selected application server shares physical memory with other application server JVM Heaps on the same machine.
	• specifying JVM Heaps to reside in physical memory and prevent swapping to disk.
	• setting the starting JVM Heap Size to one quarter of the maximum JVM Heap Size.
	• setting the maximum JVM Heap Size to the following, if you have only one application server on the machine:
	    128 MB, for small systems with less than 1 GB of memory
	    256 MB, for systems with 2 GB of memory
	    512 MB, for larger systems
	A value of 0 or blank indicates that no starting or maximum heap size is passed when initializing the JVM. On OS/400, the JVM Heap Size is quite different and you should never set the maximum heap size.

Application Bank Report	Yes
Area	JVM

Metric I210_ThreadPoolActThreads

Monitor Template Name	N/A—Used to generate a report
Metric Name	I210_ThreadPoolActThreads
Metric Type	Reporting
Description	Average number of active threads in a pool during collection interval.
Impact	High (WebSphere version 4.x, 5)
PMI Module	threadPoolModule
Severity: Condition with threshold	N/A
Collection Interval	15m
Message Group	N/A
Message Text	N/A
Instruction Text	N/A
Application Bank Report	N/A
Area	Performance

Metric I211_ThreadPoolAveSize

Monitor Template Name	N/A—Used to generate a report
Metric Name	I211_ThreadPoolAveSize
Metric Type	Reporting
Description	Average number of threads (active and idle) in a pool during collection interval
Impact	High (WebSphere version 4.x, 5)
PMI Module	threadPoolModule
Severity: Condition with threshold	N/A
Collection Interval	15m
Message Group	N/A
Message Text	N/A
Instruction Text	N/A
Application Bank Report	N/A
Area	Performance

Metric I212_ThreadPoolUtilPct

Monitor Template Name	WBSSPI_0212
Metric Name	I212_ThreadPoolUtilPct
Metric Type	Alarming
Description	Percentage of threads used in a pool during collection interval.
Impact	High (WebSphere version 4.x, 5)
PMI Module	threadPoolModule
Severity: Condition with threshold	WBSSPI-0212.1: Critical threshold, 90 WBSSPI-0212.2: Major threshold, 85 WBSSPI-0212.3: Minor threshold, 80
Collection Interval	15m
Message Group	WebSphere
Message Text	WBSSPI-0212.1: % of threads used (<$VALUE>%) too high (>=<$THRESHOLD>%)
Instruction Text	The percent of threads in use in a pool has exceeded a threshold value **Suggested action**: Open the Performance Tuner Wizard by clicking Console → Wizards→ Performance Tuner from the Administrative Console. Web Container Pool Update the pool size within the normal range by adjusting the slider. Edit the numeric field to update the pool size within, or outside the normal range. Think about specifying the maximum number of Web container threads less than the number of connections accepted by the Web server. Consider setting the Web container maximum threads number significantly smaller than the number of connections accepted by the Web server for a site with many static pages returned directly by the Web server. ORB Properties You can tune the following object request broker properties: *Pass by Reference* This option can provide better performance. Select Pass by Reference only if appropriate for your application. Selecting this option can break remote transparency, since you can modify objects passed to an EJB method. Know your application before using this option. *ORB Threads Pool Size* A thread is needed for each EJB request. Enterprise beans are typically invoked from servlets in another JVM using RMI/IIOP and remote EJB client applications using RMI/IIOP. The ORB thread pool size should accommodate both request sources.
Application Bank Report	Yes
Area	Performance

Metric I013_ThrdPoolPctMax

Monitor Template Name	N/A—Used to generate a graph
Metric Name	I013_ThrdPoolPctMax
Metric Type	Graphing
Description	Percentage of time Number of threads in pool reached configured maximum size.
Impact	High (WebSphere version 4.x, 5)
PMI Module	threadPoolModule
Severity: Condition with threshold	N/A
Collection Interval	15m
Message Group	N/A
Message Text	N/A
Instruction Text	N/A
Application Bank Report	N/A
Area	Performance

Metric I213_ThreadPoolPctMax

Monitor Template Name	WBSSPI_0213
Metric Name	I213_ThreadPoolPctMax
Metric Type	Alarming
Description	Percentage of time Number of threads in pool reached configured maximum size (drill down).
Impact	High (WebSphere version 4.x, 5)
PMI Module	threadPoolModule
Severity: Condition with threshold	WBSSPI-0213.1: Minor threshold, 10
Collection Interval	15m
Message Group	WebSphere
Message Text	WBSSPI-0213.1: % of time # of threads reached configured maximum (<$VALUE>%) too high (>=<$THRESHOLD>%)
Instruction Text	The percent of threads in use in a pool has exceeded a threshold value **Suggested action**: Open the Performance Tuner Wizard by clicking Console → Wizards→ Performance Tuner from the Administrative Console. Web Container Pool Update the pool size within the normal range by adjusting the slider. Edit the numeric field to update the pool size within, or outside the normal range. Think about specifying the maximum number of Web container threads less than the number of connections accepted by the Web server. Consider setting the Web container maximum threads number significantly smaller than the number of connections accepted by the Web server, for a site with many static pages returned directly by the Web server. ORB Properties You can tune the following object request broker properties: *Pass by Reference* This option can provide better performance. Select Pass by Reference, only if appropriate for your application. Selecting this option can break remote transparency, since you can modify objects passed to an EJB method. Know your application before using this option. *ORB Threads Pool Size* A thread is needed for each EJB request. Enterprise beans are typically invoked from servlets in another JVM, using RMI/IIOP and remote EJB client applications, using RMI/IIOP. The ORB thread pool size should accommodate both request sources.
Application Bank Report	Yes
Area	Performance

Metric I014_ThrdPoolCrtRt

Monitor Template Name	N/A—Used to generate a graph
Metric Name	I014_ThrdPoolCrtRt
Metric Type	Graphing
Description	Number of threads created per minute.
Impact	Low (WebSphere version 4.x, 5)
PMI Module	threadPoolModule
Severity: Condition with threshold	N/A
Collection Interval	15m
Message Group	N/A
Message Text	N/A
Instruction Text	N/A
Application Bank Report	N/A
Area	Performance

Metric I020_EJBPoolUtil

Monitor Template Name	N/A—Used to generate a graph
Metric Name	I020_EJBPoolUtil
Metric Type	Graphing
Description	Percentage of active beans in the pool.
Impact	High (WebSphere version 4.x, 5)
PMI Module	beanModule
Severity: Condition with threshold	N/A
Collection Interval	1h
Message Group	N/A
Message Text	N/A
Instruction Text	N/A
Application Bank Report	N/A
Area	EJB

Metric I220_EJBPoolUtil

Monitor Template Name	WBSSPI_0220
Metric Name	I220_EJBPoolUtil
Metric Type	Alarming and Reporting
Description	Percentage of active beans in the pool (drill down).
Impact	High (WebSphere version 4.x, 5)
PMI Module	beanModule
Severity: Condition with threshold	WBSSPI-0220.1: Warning threshold, 90
Message Group	WebSphere
Collection Interval	1h
Message Text	WBSSPI-0220.1: % of EJBs in the pool in use (<$VALUE>%) too high (>=<$THRESHOLD>%)

Instruction Text	The utilization of the EJB cache has exceeded a threshold value.
	
	Suggested action:
	
	EJB Container: Object Request Broker thread pool size
	Short description: Size of the thread pool.
	How to see or set:
	
	1 In the administrative console, click the appropriate application server.
	2 Click the Services tab.
	3 Select Object Request Broker and then Edit Properties. The thread pool size is on the General Properties panel.
	
	Cache settings
	Short description: To determine a rough approximation of the cache absolute limit, multiply the number of enterprise beans active in any given transaction by the total number of concurrent transactions expected. Then add the number of active session bean instances.
	
	Use the Resource Analyzer to view bean performance information.
	How to see or set:
	Edit the EJB container service properties for the application server you are tuning.
	Default value:
	
	    Cache Size = 2047
	    Cache preferred limit = 2000
	    Cache clean-up interval = 1000

Monitor Template Name	WBSSPI_0220
Instruction Text (cont.)	*Deployment descriptors* Short description: When creating deployment descriptors for your entity beans, pay close attention to the beans' functions and define your descriptors accordingly. When it is appropriate for the requirements of an application, set an entity bean's method to read-only in the deployment descriptor. How to see or set: Setting an entity bean's method to read-only can be done both in VisualAge for Java and in the Application Assembly Tool (AAT). In the AAT, within the methods extensions of the bean, set the access intent to 'read.' For each enterprise bean, the commit options are configured using the 'activate at' and 'load at' settings: *Commit Option A (exclusive database access)* This option improves performance by caching entity bean data in memory. It requires that the EJB container has exclusive access to the database used by the bean (and therefore, the only copy of a bean's persistent state), or that the bean's data is accessed as read-only at all times. For this option, use 'Activate at Once' and 'Load at Activation' settings. *Commit Option B (shared database access)* This option reloads the bean state from the database at the beginning of each transaction. If an enterprise bean contains a significant number of calls to the enterprise bean, Activate function, using option B is beneficial, because the required object is already in the cache. Otherwise, this option does not provide significant benefit over option A. For this option, use the 'Activate at Once' and 'Load at Transaction' settings. *Commit Option C (shared database access)* Entity beans are not cached across transactions. This is the default and is necessary whenever the database is shared with other processes or EJB containers. To configure this option, use the 'Activate at Transaction' and either 'Load at Transaction' or 'Load at Activation' settings.
Application Bank Report	Yes
Area	EJB

Metric I221_EJBMethRespTime

Monitor Template Name	WBSSPI_0221
Metric Name	I221_EJBMethRespTime
Metric Type	Alarming and Reporting
Severity: Condition with threshold	WBSSPI-0221.1: Major threshold, 5000
Description	Average EJB response time in milliseconds.
Impact	Medium (WebSphere version 4.x, 5)
PMI Module	beanModule
Message Group	WebSphere
Collection Interval	5m
Message Text	WBSSPI-0221.1: Ave. EJB response time (<$VALUE>ms) too high (>=<$THRESHOLD>ms)

Instruction Text	The average response time of an EJB has exceeded a threshold value.
	
	Suggested Action:
	
	EJB Container
	
	Object Request Broker thread pool size
	Short description: Size of the thread pool.
	How to see or set:
	1 In the administrative console, click the appropriate application server.
	2 Click the Services tab.
	3 Select Object Request Broker and then Edit Properties. The thread pool size is on the General Properties panel.
	
	Cache settings
	Short description: To determine a rough approximation of the cache absolute limit, multiply the number of enterprise beans active in any given transaction by the total number of concurrent transactions expected. Then add the number of active session bean instances.
	
	Use the Resource Analyzer to view bean performance information.
	
	How to see or set:
	Edit the EJB container service properties for the application server you are tuning.
	
	Default value:
	
	    Cache Size = 2047
	    Cache preferred limit = 2000
	    Cache clean-up interval = 1000

Monitor Template Name	WBSSPI_0221
Instruction Text (cont.)	Deployment descriptors *Short description*: When creating deployment descriptors for your entity beans, pay close attention to the beans' functions and define your descriptors accordingly. When it is appropriate for the requirements of an application, set an entity bean's method to read-only in the deployment descriptor. How to see or set: Setting an entity bean's method to read-only can be done both in VisualAge for Java and in the Application Assembly Tool (AAT). In the AAT, within the methods extensions of the bean, set the access intent to 'read.' For each enterprise bean, the commit options are configured using the 'activate at' and 'load at' settings: *Commit Option A* (exclusive database access) This option improves performance by caching entity bean data in memory. It requires that the EJB container have exclusive access to the database used by the bean (and therefore, the only copy of a bean's persistent state), or that the bean's data is accessed as read-only at all times. For this option, use 'Activate at Once' and 'Load at Activation' settings. *Commit Option B* (shared database access) This option reloads the bean state from the database at the beginning of each transaction. If an enterprise bean contains a significant number of calls to the enterprise bean, Activate function, using option B is beneficial, because the required object is already in the cache. Otherwise, this option does not provide significant benefit over option A. For this option, use the 'Activate at Once' and 'Load at Transaction' settings. *Commit Option C* (shared database access) Entity beans are not cached across transactions. This is the default and is necessary whenever the database is shared with other processes or EJB containers. To configure this option, use the 'Activate at Transaction' and either 'Load at Transaction' or 'Load at Activation' settings.
Application Bank Report	Yes
Area	EJB

Metric I022_EJBMethCallsRt

Monitor Template Name	N/A—Used to generate a report and graph
Metric Name	I022_EJBMethCallsRt
Metric Type	Graphing and Reporting
Description	Number of EJB method calls per minute.
Impact	Low (WebSphere version 4.x, 5)
PMI Module	beanModule
Severity: Condition with threshold	N/A
Collection Interval	5m
Message Group	N/A
Message Text	N/A
Instruction Text	N/A
Application Bank Report	N/A
Area	EJB

Metric I222_EJBMethodCallsRt

Monitor Template Name	WBSSPI_0222
Metric Name	I222_EJBMethodCallsRt
Metric Type	Alarming and Reporting
Description	Number of EJB method calls per minute (drill down).
Impact	Low (WebSphere version 4.x, 5)
PMI Module	beanModule
Severity: Condition with threshold	WBSSPI-0222.1: Warning threshold, 10
Collection Interval	5m
Message Group	WebSphere
Message Text	WBSSPI-0222.1: # of EJB method calls per minute (<$VALUE>/min) too high (>=<$THRESHOLD>/min)
Instruction Text	The number of EJB method calls per minute has exceeded a threshold value.

Instruction Text (continued):

The number of EJB method calls per minute has exceeded a threshold value.

Suggested action:

EJB Container

Object Request Broker thread pool size

Short description: Size of the thread pool.

How to see or set:

1 In the administrative console, click the appropriate application server.
2 Click the Services tab.
3 Select Object Request Broker and then Edit Properties. The thread pool size is on the General Properties panel.

Cache settings
Short description: To determine a rough approximation of the cache absolute limit, multiply the number of enterprise beans active in any given transaction by the total number of concurrent transactions expected. Then add the number of active session bean instances.

Use the Resource Analyzer to view bean performance information.

How to see or set:
Edit the EJB container service properties for the application server you are tuning.

Default value:
Cache Size = 2047
Cache preferred limit = 2000
Cache clean-up interval = 1000

Monitor Template Name	WBSSPI_0222
Instruction Text (cont.)	Deployment descriptors *Short description*: When creating deployment descriptors for your entity beans, pay close attention to the beans' functions and define your descriptors accordingly. When it is appropriate for the requirements of an application, set an entity bean's method to read-only in the deployment descriptor. How to see or set: Setting an entity bean's method to read-only can be done both in VisualAge for Java and in the Application Assembly Tool (AAT). In the AAT, within the methods extensions of the bean, set the access intent to 'read.' For each enterprise bean, the commit options are configured using the 'activate at' and 'load at' settings: *Commit Option A* (exclusive database access) This option improves performance by caching entity bean data in memory. It requires that the EJB container has exclusive access to the database used by the bean (and therefore, the only copy of a bean's persistent state), or that the bean's data is accessed as read-only at all times. For this option, use 'Activate at Once' and 'Load at Activation' settings. *Commit Option B* (shared database access) This option reloads the bean state from the database at the beginning of each transaction. If an enterprise bean contains a significant number of calls to the enterprise bean, Activate function, using option B is beneficial, because the required object is already in the cache. Otherwise, this option does not provide significant benefit over option A. For this option, use the 'Activate at Once' and 'Load at Transaction' settings. *Commit Option C* (shared database access) Entity beans are not cached across transactions. This is the default and is necessary whenever the database is shared with other processes or EJB containers. To configure this option, use the 'Activate at Transaction' and either 'Load at Transaction' or 'Load at Activation' settings.
Application Bank Report	Yes
Area	EJB

Metric I223_EJBPoolSize

Monitor Template Name	N/A—Used to generate a report
Metric Name	I223_EJBPoolSize
Metric Type	Reporting
Description	Average size of the EJB pool.
Impact	High (WebSphere version 4.x, 5)
PMI Module	beanModule
Severity: Condition with threshold	N/A
Collection Interval	5m
Message Group	N/A
Message Text	N/A
Instruction Text	N/A
Application Bank Report	N/A
Area	EJB

Metric I024_EJBEntDatLdStRt

Monitor Template Name	N/A—Used to generate a report and graph
Metric Name	I024_EJBEntDatLdStRt
Metric Type	Graphing and Reporting
Description	Number of times an EJB was written to or loaded from the database per minute.
Impact	Low (WebSphere version 4.x, 5)
PMI Module	beanModule
Severity: Condition with threshold	N/A
Collection Interval	5m
Message Group	N/A
Message Text	N/A
Instruction Text	N/A
Application Bank Report	N/A
Area	EJB

Metric I224_EJBEntDataLdStRt

Monitor Template Name	WBSSPI_0224
Metric Name	I224_EJBEntDataLdStRt
Metric Type	Alarming and Reporting
Description	Number of times an EJB was written to or loaded from the database per minute (drill down).
Impact	Low (WebSphere version 4.x, 5)
PMI Module	beanModule
Severity: Condition with threshold	WBSSPI-0224.1: Warning threshold
Collection Interval	15m
Message Group	WebSphere
Message Text	WBSSPI-0224.1: # of times EJB data was written to or loaded from the database per minute (<$VALUE>/min) too high (>=<$THRESHOLD>/min)
Instruction Text	The number of times an EJB was written to or loaded from the database per minute has exceeded a threshold value. **Suggested action**: EJB Container Object Request Broker thread pool size Short description: Size of the thread pool. How to see or set: 1 In the administrative console, click the appropriate application server. 2 Click the Services tab. 3 Select Object Request Broker and then Edit Properties. The thread pool size is on the General Properties panel. Cache settings Short description: To determine a rough approximation of the cache absolute limit, multiply the number of enterprise beans active in any given transaction by the total number of concurrent transactions expected. Then add the number of active session bean instances. Use the Resource Analyzer to view bean performance information. How to see or set: Edit the EJB container service properties for the application server you are tuning. Default value: Cache Size = 2047 Cache preferred limit = 2000 Cache clean-up interval = 1000

Monitor Template Name	WBSSPI_0224
Instruction Text (cont.)	Deployment descriptors *Short description*: When creating deployment descriptors for your entity beans, pay close attention to the beans' functions and define your descriptors accordingly. When it is appropriate for the requirements of an application, set an entity bean's method to read-only in the deployment descriptor. How to see or set: Setting an entity bean's method to read-only can be done both in VisualAge for Java and in the Application Assembly Tool (AAT). In the AAT, within the methods extensions of the bean, set the access intent to 'read.' For each enterprise bean, the commit options are configured using the 'activate at' and 'load at' settings: *Commit Option A* (exclusive database access) This option improves performance by caching entity bean data in memory. It requires that the EJB container has exclusive access to the database used by the bean (and therefore, the only copy of a bean's persistent state), or that the bean's data is accessed as read-only at all times. For this option, use 'Activate at Once' and 'Load at Activation' settings. *Commit Option B* (shared database access) This option reloads the bean state from the database at the beginning of each transaction. If an enterprise bean contains a significant number of calls to the enterprise bean, Activate function, using option B is beneficial, because the required object is already in the cache. Otherwise, this option does not provide significant benefit over option A. For this option, use the 'Activate at Once' and 'Load at Transaction' settings. *Commit Option C* (shared database access) Entity beans are not cached across transactions. This is the default and is necessary whenever the database is shared with other processes or EJB containers. To configure this option, use the 'Activate at Transaction' and either 'Load at Transaction' or 'Load at Activation' settings.
Application Bank Report	Yes
Area	EJB

Metric I025_EJBPoolMissPct

Monitor Template Name	N/A—Used to generate a graph
Metric Name	I025_EJBPoolMissPct
Metric Type	Graphing
Description	Average Percentage of time a call to retrieve an EJB from the pool failed.
Impact	Low (WebSphere version 4.x, 5)
PMI Module	beanModule
Severity: Condition with threshold	Warning: WBSSPI-0025.1, threshold 10.
Collection Interval	5m
Message Group	N/A
Message Text	N/A
Instruction Text	N/A
Application Bank Report	N/A
Area	EJB

Metric I225_EJBPoolMissPct

Monitor Template Name	N/A—Used to generate a report
Metric Name	I225_EJBPoolMissPct
Metric Type	Reporting
Description	Average Percentage of time a call to retrieve an EJB from the pool failed (drill down).
Impact	Low (WebSphere version 4.x, 5)
PMI Module	beanModule
Severity: Condition with threshold	N/A
Collection Interval	5m
Default OVO Threshold	10
Message Group	WebSphere
Message Text	N/A
Instruction Text	N/A
Application Bank Report	N/A
Area	EJB

Metric I026_EJBConcLives

Monitor Template Name	WBSSPI_0026
Metric Name	I026_EJBConcLives
Metric Type	Alarming and Graphing
Description	Average Number of bean objects in the pool.
Impact	High (WebSphere version 4.x, 5)
PMI Module	beanModule
Severity: Condition with threshold	WBSSPI-0026.1: Warning threshold, 1000
Collection Interval	5m
Default OVO Threshold	10
Message Group	WebSphere
Message Text	WBSSPI-0026.1: Ave. # of bean objects in the pool (<$VALUE>) too high (>=<$THRESHOLD>)
Instruction Text	The average number of bean objects in the pool has exceeded a threshold value.
	Suggested action:
	EJB Container
	Object Request Broker thread pool size Short description: Size of the thread pool. How to see or set:
	1 In the administrative console, click the appropriate application server.
	2 Click the Services tab.
	3 Select Object Request Broker and then Edit Properties.
	4 The thread pool size is on the General Properties panel.
	Cache settings
	Short description: To determine a rough approximation of the cache absolute limit, multiply the number of enterprise beans active in any given transaction by the total number of concurrent transactions expected. Then add the number of active session bean instances.
	Use the Resource Analyzer to view bean performance information.
	How to see or set:
	Edit the EJB container service properties for the application server you are tuning.
	Default value: Cache Size = 2047 Cache preferred limit = 2000 Cache clean-up interval = 1000

Monitor Template Name	WBSSPI_0026
Instruction Text (cont.)	Deployment descriptors
	Short description: When creating deployment descriptors for your entity beans, pay close attention to the beans' functions and define your descriptors accordingly. When it is appropriate for the requirements of an application, set an entity bean's method to read-only in the deployment descriptor.
	How to see or set: Setting an entity bean's method to read-only can be done both in VisualAge for Java and in the Application Assembly Tool (AAT). In the AAT, within the methods extensions of the bean, set the access intent to 'read.'
	For each enterprise bean, the commit options are configured using the 'activate at' and 'load at' settings:
	Commit Option A (exclusive database access) This option improves performance by caching entity bean data in memory. It requires that the EJB container has exclusive access to the database used by the bean (and therefore, the only copy of a bean's persistent state), or that the bean's data is accessed as read-only at all times. For this option, use 'Activate at Once' and 'Load at Activation' settings.
	Commit Option B (shared database access) This option reloads the bean state from the database at the beginning of each transaction. If an enterprise bean contains a significant number of calls to the enterprise bean, Activate function, using option B is beneficial, because the required object is already in the cache. Otherwise, this option does not provide significant benefit over option A. For this option, use the 'Activate at Once' and 'Load at Transaction' settings.
	Commit Option C (shared database access) Entity beans are not cached across transactions. This is the default and is necessary whenever the database is shared with other processes or EJB containers. To configure this option, use the 'Activate at Transaction' and either 'Load at Transaction' or 'Load at Activation' settings.
Application Bank Report	Yes
Area	EJB

Metric I040_ServSessAveLife

Monitor Template Name	WBSSPI_0040
Metric Name	I040_ServSessAveLife
Metric Type	Alarming and Graphing
Description	Average servlet session lifetime in milliseconds.
Impact	Medium (WebSphere version 4.x, 5)
PMI Module	servletSessionModule
Severity: Condition with threshold	WBSSPI-0040.1: Warning threshold, 1000
Collection Interval	1h
WebSphere	Message Group
Message Text	WBSSPI-0040.1: Ave. servlet session lifetime (<$VALUE>ms) too high (>=<$THRESHOLD>ms)
Instruction Text	The average servlet session lifetime has exceeded a threshold value. **Suggested action**: Web Containers To route servlet requests from the Web server to the Web containers, the product establishes a transport queue between the Web server plug-in and each Web container. Web container maximum thread size Short description: Use the maximum thread size parameter to specify the number of connections to use for the communications channel between the Web server and a Web container. Each connection represents a request for a servlet. How to see or set: 1 In the administrative console, select the application server you are tuning and then click the Services tab. 2 Click Web Container Service and then click Edit Properties. 3 In the Web Container Service window, click the General tab. 4 Specify the value in the Maximum Thread Size field. 5 Click Apply after returning to the Services panel, to ensure that the changes are saved. 6 Stop and restart the application server. Default value: 50 Note: For Linux systems, the recommended value is 25, not the default of 50. Related parameters: See Adjusting WebSphere's System Queues and Prepared Statement Cache Size

Monitor Template Name	WBSSPI_0040
Instruction Text (cont.)	URL invocation cache Short description: The Invocation Cache holds information for mapping request URLs to servlet resources. A cache of the requested size is created for each thread/process. The number of threads/processes is determined by the Web container maximum thread size setting. Note the following considerations when increasing this cache size: A larger cache uses more of the Java heap, so you might also need to increase maximum Java heap size. For example, if each cache entry requires 2KB, maximum thread size is set to 25, and the URL Invocation cache size is 100; then 5MB of Java heap are required. When to try adjusting: If you have more than 50 unique URLs actively being used (each JSP page is a unique URL), increase this parameter. How to see or set: The size of the cache can be specified for the application server along with other JDK parameters by: 1 In the administrative console, click the application server you are tuning. 2 Click the JVM Setting tab. 3 On the same panel, click Add in the System Properties section. 4 Add the name -DinvocationCacheSize and a value of 50. 5 Click Apply to ensure that the changes are saved. 6 Stop and restart the application server. Default value: 50
Application Bank Report	Yes
Area	Servlets

Metric I041_ServSessActSess

Monitor Template Name	WBSSPI_0041
Metric Name	I041_ServSessActSess
Metric Type	Alarming, Graphing, and Reporting
Description	Number of sessions currently being accessed.
Impact	High (WebSphere version 4.x, 5)
PMI Module	servletSessionModule
Severity: Condition	WBSSPI-0041.1: Warning threshold, 10000
Collection Interval	1h
Message Group	WebSphere
Message Text	WBSSPI-0041.1: # of sessions currently being accessed (<$VALUE>) too high (>=<$THRESHOLD>)
Instruction Text	The number of sessions currently being accessed has exceeded a threshold value. **Suggested action**: <u>Web Containers</u> To route servlet requests from the Web server to the Web containers, the product establishes a transport queue between the Web server plug-in and each Web container. Web container maximum thread size Short description: Use the maximum thread size parameter to specify the number of connections to use for the communications channel between the Web server and a Web container. Each connection represents a request for a servlet. How to see or set: 1 In the administrative console, select the application server you are tuning and then click the Services tab. 2 Click Web Container Service and then click Edit Properties. 3 In the Web Container Service window, click the General tab. 4 Specify the value in the Maximum Thread Size field. 5 Click Apply after returning to the Services panel, to ensure that the changes are saved. 6 Stop and restart the application server. Default value: 50 Note: For Linux systems, the recommended value is 25, not the default of 50. Related parameters: See Adjusting WebSphere's System Queues and Prepared Statement Cache Size

Monitor Template Name	WBSSPI_0041
Instruction Text (cont.)	URL invocation cache
	Short description: The Invocation Cache holds information for mapping request URLs to servlet resources.
	A cache of the requested size is created for each thread/process. The number of threads/processes is determined by the Web container maximum thread size setting.
	Note the following considerations when increasing this cache size:
	A larger cache uses more of the Java heap, so you might also need to increase maximum Java heap size. For example, if each cache entry requires 2KB, maximum thread size is set to 25, and the URL Invocation cache size is 100; then 5MB of Java heap are required.
	When to try adjusting: If you have more than 50 unique URLs actively being used (each JSP page is a unique URL), increase this parameter.
	How to see or set: The size of the cache can be specified for the application server along with other JDK parameters by:
	1 In the administrative console, click the application server you are tuning.
	2 Click the JVM Setting tab.
	3 On the same panel, click Add in the System Properties section.
	4 Add the name -DinvocationCacheSize and a value of 50.
	5 Click Apply to ensure that the changes are saved.
	6 Stop and restart the application server.
Application Bank Report	Yes
Area	Servlets

Metric I042_ServInvSessRt

Monitor Template Name	WBSSPI_0042
Metric Name	I042_ServInvSessRt
Metric Type	Alarming and Graphing
Description	Number of sessions being invalidated per second.
Impact	Low (WebSphere version 4.x, 5)
PMI Module	servletSessionModule
Severity: Condition	WBSSPI-0042.1: Warning threshold, 10000
Collection Interval	1h
Message Group	WebSphere
Message Text	WBSSPI-0042.1: # of sessions timed out per second (<$VALUE>/sec) too high (>=<$THRESHOLD>/sec)
Instruction Text	The number of sessions being invalidated per second has exceeded a threshold value.
	Suggested action:
	Web Containers
	To route servlet requests from the Web server to the Web containers, the product establishes a transport queue between the Web server plug-in and each Web container.
	Web container maximum thread size Short description: Use the maximum thread size parameter to specify the number of connections to use for the communications channel between the Web server and a Web container. Each connection represents a request for a servlet.
	How to see or set:
	1 In the administrative console, select the application server you are tuning and then click the Services tab.
	2 Click Web Container Service and then click Edit Properties.
	3 In the Web Container Service window, click the General tab.
	4 Specify the value in the Maximum Thread Size field.
	5 Click Apply after returning to the Services panel, to ensure that the changes are saved.
	6 Stop and restart the application server.
	Default value: 50
	Note: For Linux systems, the recommended value is 25, not the default of 50. Related parameters: See Adjusting WebSphere's System Queues and Prepared Statement Cache Size

Monitor Template Name	WBSSPI_0042
Instruction Text (cont.)	URL invocation cache Short description: The Invocation Cache holds information for mapping request URLs to servlet resources. A cache of the requested size is created for each thread/process. The number of threads/processes is determined by the Web container maximum thread size setting. Note the following considerations when increasing this cache size: A larger cache uses more of the Java heap, so you might also need to increase maximum Java heap size. For example, if each cache entry requires 2KB, maximum thread size is set to 25, and the URL Invocation cache size is 100; then 5MB of Java heap are required. When to try adjusting: If you have more than 50 unique URLs actively being used (each JSP page is a unique URL), increase this parameter. How to see or set: The size of the cache can be specified for the application server along with other JDK parameters by: 1 In the administrative console, click the application server you are tuning. 2 Click the JVM Setting tab. 3 On the same panel, click Add in the System Properties section. 4 Add the name -DinvocationCacheSize and a value of 50. 5 Click Apply to ensure that the changes are saved. 6 Stop and restart the application server. Default value: 50
Application Bank Report	Yes
Area	Servlets

Metric I045_WebAppServReqRt

Monitor Template Name	WBSSPI_0245
Metric Name	I045_WebAppServReqRt
Metric Type	Graphing and Reporting
Description	Number of requests for a servlet per second.
Impact	Low (WebSphere version 4.x, 5)
PMI Module	webAppModule
Severity: Condition	N/A
Collection Interval	1h
Message Group	N/A
Message Text	N/A
Instruction Text	N/A
Application Bank Report	N/A
Area	Web Applications

Metric I245_WebAppServletReqRt

Monitor Template Name	WBSSPI_0245
Metric Name	I245_WebAppServletReqRt
Metric Type	Alarming and Reporting
Description	Number of requests for a servlet per second (drill down)./
Impact	Low (WebSphere version 4.x, 5)
PMI Module	webAppModule
Severity: Condition	WBSSPI-0245.1: Warning threshold, 1000
Collection Interval	1h
Message Group	N/A
Message Text	WBSSPI-0245.1: Ave. request rate for a web application servlet (<$VALUE>/sec) too high (>=<$THRESHOLD>/sec)
Instruction Text	The number of requests for a servlet per second has exceeded a threshold value. **Suggested action**: Web Applications You can also set parameters specific to each Web application you deploy. The settings can affect performance. Servlet Reload Interval and Reloading Enabled Short description: WebSphere application server offers an auto reload capability. The default automatically reloads servlets in the Web application when the class files change. The auto reload capability can simplify the testing and management of your Web site's applications by enabling you to quickly modify your site without restarting the WebSphere application server. (Be sure that your Reload Interval is short). However, this ability to reload servlets dynamically and the associated polling affects performance negatively. When the application's resources (such as servlets and enterprise beans) are fully deployed, it is not as necessary to aggressively reload these resources as during development. When to try adjusting: When you are in a stable production mode, you need to either set a long Reload Interval or disable Reloading. For a production system, it is common to reload resources only a few times a day. How to see or set: The Reload Interval and Reloading Enabled can be set for your application by using the Application Assembler from the administrative console. When creating a new Web module, these parameters can be configured by selecting the IBM Extensions and 1 Unchecking the Reloading Enabled box. 2 Updating the Reload Interval field. Default value: Reload Interval = three seconds Reloading Enabled=true
Application Bank Report	Yes
Area	Web Applications

Metric I246_WebAppServletRespTime

Monitor Template Name	WBSSPI_0246
Metric Name	I246_WebAppServletRespTime
Metric Type	Alarming and Reporting
Description	Average response time for a servlet in msec.
Impact	Medium (WebSphere version 4.x, 5)
PMI Module	webAppModule
Severity: Conditions	Condition/WBSSPI-0246.1: Major, 10000 Condition/WBSSPI-0246.2: Warning, 2000
Collection Interval	1h
Message Group	WebSphere
Message Text	WBSSPI-0246.1: Ave. response time for a web application servlet (<$VALUE>ms) too high (>=<$THRESHOLD>ms)
Instruction Text	The average response time for servlet in msec has exceeded a threshold value. **Suggested action**: Web Applications You can also set parameters specific to each Web application you deploy. The settings can affect performance. Servlet Reload Interval and Reloading Enabled Short description: WebSphere application server offers an auto reload capability. The default automatically reloads servlets in the Web application when the class files change. The auto reload capability can simplify the testing and management of your Web site's applications by enabling you to quickly modify your site without restarting the WebSphere application server. (Be sure that your Reload Interval is short). However, this ability to reload servlets dynamically and the associated polling affects performance negatively. When the application's resources (such as servlets and enterprise beans) are fully deployed, it is not as necessary to aggressively reload these resources as during development. When to try adjusting: When you are in a stable production mode, you need to either set a long Reload Interval or disable Reloading. For a production system, it is common to reload resources only a few times a day. How to see or set: The Reload Interval and Reloading Enabled can be set for your application by using the Application Assembler from the administrative console. When creating a new Web module, these parameters can be configured by selecting the IBM Extensions and 1 Unchecking the Reloading Enabled box. 2 Updating the Reload Interval field. Default value: Reload Interval = three seconds Reloading Enabled=true
Application Bank Report	Yes
Area	Web Applications

Metric I047_WebAppServErrRt

Monitor Template Name	N/A—Used to generate a graph
Metric Name	I047_WebAppServErrRt
Metric Type	Graphing
Description	Number of errors in a servlet per second.
Impact	Low (WebSphere version 4.x, 5)
PMI Module	webAppModule
Severity: Condition	N/A
Collection Interval	1h
Message Group	N/A
Message Text	
Instruction Text	
Application Bank Report	N/A
Area	Web Applications

Metric I247_WebAppServletErrorRt

Monitor Template Name	WBSSPI_0247
Metric Name	WI247_WebAppServletErrorRt
Metric Type	Alarming
Description	Number of errors in a servlet per second (drill down).
Impact	Low (WebSphere version 4.x, 5)
PMI Module	webAppModule
Severity: Condition	WBSSPI-0247.1: Warning, 100
Collection Interval	1h
Message Group	WebSphere
Message Text	WBSSPI-0247.1: # of errors for a web application servlet per second (<$VALUE>/sec) too high (>=<$THRESHOLD>/sec)
Instruction Text	The number of errors in a servlet per second has exceeded a threshold value. **Suggested action**: Web Applications You can also set parameters specific to each Web application you deploy. The settings can affect performance. Servlet Reload Interval and Reloading Enabled Short description: WebSphere application server offers an auto reload capability. The default automatically reloads servlets in the Web application when the class files change. How to see or set: The Reload Interval and Reloading Enabled can be set for your application by using the Application Assembler from the administrative console. When creating a new Web module, these parameters can be configured by selecting the IBM Extensions and 1 Unchecking the Reloading Enabled box. 2 Updating the Reload Interval field. Default value: Reload Interval = three seconds Reloading Enabled=true The auto reload capability can simplify the testing and management of your Web site's applications by enabling you to quickly modify your site without restarting the WebSphere application server. (Be sure that your Reload Interval is short). However, this ability to reload servlets dynamically and the associated polling affects performance negatively. When the application's resources (such as servlets and enterprise beans) are fully deployed, it is not as necessary to aggressively reload these resources as during development. When to try adjusting: When you are in a stable production mode, you need to either set a long Reload Interval or disable Reloading. For a production system, it is common to reload resources only a few times a day.

Monitor Template Name	WBSSPI_0247
Instruction Text (cont.)	How to see or set: The Reload Interval and Reloading Enabled can be set for your application by using the Application Assembler from the administrative console. When creating a new Web module, these parameters can be configured by selecting the IBM Extensions and 1 Unchecking the Reloading Enabled box. 2 Updating the Reload Interval field. Default value: Reload Interval = three seconds Reloading Enabled=true
Application Bank Report	Yes
Area	Web Applications

Metric I048_WebAppServLoad

Monitor Template Name	N/A—Used to generate a graph
Metric Name	I048_WebAppServLoad
Metric Type	Alarming and Graphing
Severity: Condition	WBSSPI-0048.1: Warning threshold, 100
Description	Number of servlets currently loaded for a web application.
Impact	Low (WebSphere version 4.x, 5)
PMI Module	webAppModule
Collection Interval	1h
Message Group	N/A
Message Text	WBSSPI-0048.1: # of servlets currently loaded for a web application (<$VALUE>) too high (>=<$THRESHOLD>)
Instruction Text	The number of servlets currently loaded for a web application has exceeded a threshold value. **Suggested action**: <u>Web Applications</u> You can also set parameters specific to each Web application you deploy. The settings can affect performance. Servlet Reload Interval and Reloading Enabled Short description: WebSphere application server offers an auto reload capability. The default automatically reloads servlets in the Web application when the class files change. The auto reload capability can simplify the testing and management of your Web site's applications by enabling you to quickly modify your site without restarting the WebSphere application server. (Be sure that your Reload Interval is short). However, this ability to reload servlets dynamically and the associated polling affects performance negatively. When the application's resources (such as servlets and enterprise beans) are fully deployed, it is not as necessary to aggressively reload these resources as during development. When to try adjusting: When you are in a stable production mode, you need to either set a long Reload Interval or disable Reloading. For a production system, it is common to reload resources only a few times a day. How to see or set: The Reload Interval and Reloading Enabled can be set for your application by using the Application Assembler from the administrative console. When creating a new Web module, these parameters can be configured by selecting the IBM Extensions and: (1) Unchecking the Reloading Enabled box. (2) Updating the Reload Interval field. Default value: Reload Interval = three seconds Reloading Enabled=true
Application Bank Report	Yes
Area	Web Applications

Metric I049_WebAppServRelRt

Monitor Template Name	N/A—Used to generate a graph
Metric Name	I049_WebAppServRelRt
Metric Type	Graphing
Description	Number of servlets reloaded for a web application per minute.
Impact	Low (WebSphere version 4.x, 5)
PMI Module	webAppModule
Severity: Condition	N/A
Collection Interval	1h
Message Group	WebSphere
Message Text	N/A
Instruction Text	N/A
Application Bank Report	N/A
Area	Web Applications

Metric I260_JDBCConnPoolSize

Monitor Template Name	WBSSPI_0260
Metric Name	I260_JDBCConnPoolSize
Metric Type	Alarming and Reporting
Description	Average Number of connections in the connection pool.
Impact	High (WebSphere version 4.x, 5)
PMI Module	connectionPoolModule
Severity: Condition	Warning: WBSSPI-0260.1, threshold 10000
Collection Interval	5m
Message Group	WebSphere
Message Text	WBSSPI-0260.1: Ave. # of connections in the connection pool (<$VALUE>) too high (>=<$THRESHOLD>)
Instruction Text	The average number of connections in the connection pool has exceeded a threshold value. **Suggested action**: Open the Performance Tuner Wizard by clicking Console → Wizards → Performance Tuner from the Administrative Console. Data Source: Connection Pool Size Each data source contains a pool of connections to the corresponding database. An upper bound for your application comes from the sum of the number of Web container threads and ORB threads. The servlet contribution to the connection pool size can be significantly smaller than the Web container maximum threads, if only a small percentage of servlet requests use these database connections. Data Source: Prepared Statement Cache Size Make your prepared statement cache large enough for all prepared statements, by setting the cache size to the product of: The number of SQL prepared statements in your application The maximum number of configured data source connections Database: (DB2 Only) This panel is only available for DB2 databases. Tune the database after this wizard has completed, by selecting Tune database and entering the DB2 SYSADM ID and password. If you provided a cataloged database alias name when configuring the data source and this alias name differs from the real database name, enter the real database name in the field provided.

Monitor Template Name	WBSSPI_0260
Instruction Text (cont.)	After database tuning is selected, and you have completed the remaining panels, click Finish on the Summary panel. The tuning wizard then calls the DB2SmartGuide API, to tune the DB2 database associated with the data source. Stop and restart the database instance, for the DB2SmartGuide changes to take effect.
	Note: DB2SmartGuide tuning works better if the database is already populated. It is not necessary or recommended to tune the repository database (WAS).
	Before tuning a database, you might want to use the DB2 DBA utility to back up (db2cfexp) the database configuration. If the tuning fails, you can then restore (db2cfimp) your database configuration. You can also use:
	DB2 RESET DATABASE CONFIGURATION FOR database-name to restore the database to default valuesDB2 RESET DATABASE MANAGER CONFIGURATION to restore the DBM to default values
	The database being tuned must reside on a DB2 Server at Version 7.2.1, or higher. This tuning option is not available in DB2 servers for OS/390, OS/400, VM or VSE.
Application Bank Report	Yes
Area	JDBC

Metric I061_JDBCConPoolWait

Monitor Template Name	N/A—Used to generate a graph
Metric Name	I061_JDBCConPoolWait
Metric Type	Graphing
Description	Average number of threads waiting for a connection from connection pools.
Impact	High (WebSphere version 4.x, 5)
PMI Module	connectionPoolModule
Severity: Condition	Warning: WBSSPI-0061.1, threshold 100
Collection Interval	5m
Message Group	WebSphere
Message Text	N/A
Instruction Text	N/A
Application Bank Report	Yes
Area	JDBC

Metric I261_JDBCConnPoolWaiters

Monitor Template Name	WBSSPI_0261
Metric Name	I261_JDBCConnPoolWaiters
Metric Type	Alarming and Reporting
Description	Average Number of threads waiting for a connection from connection pools (drill down).
Impact	High (WebSphere version 4.x, 5)
PMI Module	connectionPoolModule
Severity: Condition with threshold	WBSSPI-0261.1: Major, 10 WBSSPI-0261.2: Warning, 1
Collection Interval	5m
Message Group	WebSphere
Message Text	WBSSPI-0261.1: Ave. # of threads waiting for a connection from connection pools (<$VALUE>) too high (>=<$THRESHOLD>)
Instruction Text	The average number of threads waiting for a connection from the connection pool has exceeded a threshold value. Suggested action: Open the Performance Tuner Wizard by clicking Console →Wizards → Performance Tuner from the Administrative Console. Data Source: Connection Pool Size Each data source contains a pool of connections to the corresponding database. An upper bound for your application comes from the sum of the number of Web container threads and ORB threads. The servlet contribution to the connection pool size can be significantly smaller than the Web container maximum threads, if only a small percentage of servlet requests use these database connections. Data Source: Prepared Statement Cache Size Make your prepared statement cache large enough for all prepared statements, by setting the cache size to the product of: The number of SQL prepared statements in your application The maximum number of configured data source connections *Database: (DB2 Only)* This panel is only available for DB2 databases. Tune the database after this wizard has completed, by selecting Tune database and entering the DB2 SYSADM ID and password. If you provided a cataloged database alias name when configuring the data source and this alias name differs from the real database name, enter the real database name in the field provided.

Monitor Template Name	WBSSPI_0261
Instruction Text (cont.)	After database tuning is selected, and you have completed the remaining panels, click Finish on the Summary panel. The tuning wizard then calls the DB2SmartGuide API, to tune the DB2 database associated with the data source. Stop and restart the database instance, for the DB2SmartGuide changes to take effect. Note: DB2SmartGuide tuning works better if the database is already populated. It is not necessary or recommended to tune the repository database (WAS). Before tuning a database, you might want to use the DB2 DBA utility to back up (db2cfexp) the database configuration. If the tuning fails, you can then restore (db2cfimp) your database configuration. You can also use: DB2 RESET DATABASE CONFIGURATION FOR database-name to restore the database to default values DB2 RESET DATABASE MANAGER CONFIGURATION to restore the DBM to default values The database being tuned must reside on a DB2 Server at Version 7.2.1, or higher. This tuning option is not available in DB2 servers for OS/390, OS/400, VM or VSE.
Application Bank Report	Yes
Area	JDBC

Metric I062_JDBConPoolWtTim

Monitor Template Name	N/A—Used to generate a graph
Metric Name	I062_JDBConPoolWtTim
Metric Type	Graphing
Description	Average time that a client waited for a connection in milliseconds.
Impact	Medium (WebSphere version 4.x); High (WebSphere version 5)
PMI Module	connectionPoolModule
Severity: Condition	N/A
Collection Interval	5m
Message Group	N/A
Message Text	NA
Instruction Text	N/A
Application Bank Report	N/A
Area	JDBC

Metric I262_JDBCConnPoolWaitTime

Monitor Template Name	WBSSPI_0262
Metric Name	I262_JDBCConnPoolWaitTime
Metric Type	Alarming and Reporting
Description	Average time that a client waited for a connection in msec (drill down).
Impact	Medium (WebSphere version 4.x); High (WebSphere version 5)
PMI Module	connectionPoolModule
Severity: Condition	WBSSPI-0262.1: Major, 50 WBSSPI-0262.2: Warning
Collection Interval	5m
Message Group	WebSphere
Message Text	WBSSPI-0262.1: Ave. time a client waited for a connection (<$VALUE>ms) too high (>=<$THRESHOLD>ms)
Instruction Text	The average time that a client waited for a connection in has exceeded a threshold value. Suggested action: Open the Performance Tuner Wizard by clicking Console → Wizards → Performance Tuner from the Administrative Console. Data Source: Connection Pool Size Each data source contains a pool of connections to the corresponding database. An upper bound for your application comes from the sum of the number of Web container threads and ORB threads. The servlet contribution to the connection pool size can be significantly smaller than the Web container maximum threads, if only a small percentage of servlet requests use these database connections. Data Source: Prepared Statement Cache Size Make your prepared statement cache large enough for all prepared statements, by setting the cache size to the product of: The number of SQL prepared statements in your application. The maximum number of configured data source connections *Database: (DB2 Only)* This panel is only available for DB2 databases. Tune the database after this wizard has completed, by selecting Tune database and entering the DB2 SYSADM ID and password. If you provided a cataloged database alias name when configuring the data source and this alias name differs from the real database name, enter the real database name in the field provided. After database tuning is selected, and you have completed the remaining panels, click Finish on the Summary panel. The tuning wizard then calls the DB2SmartGuide API, to tune the DB2 database associated with the data source. Stop and restart the database instance, for the DB2SmartGuide changes to take effect.

Monitor Template Name	WBSSPI_0262
Instruction Text (cont.)	Note: DB2SmartGuide tuning works better if the database is already populated. It is not necessary or recommended to tune the repository database (WAS).
	Before tuning a database, you might want to use the DB2 DBA utility to back up (db2cfexp) the database configuration. If the tuning fails, you can then restore (db2cfimp) your database configuration. You can also use:
	DB2 RESET DATABASE CONFIGURATION FOR database-name to restore the database to default values DB2 RESET DATABASE MANAGER CONFIGURATION to restore the DBM to default values
	The database being tuned must reside on a DB2 Server at Version 7.2.1, or higher. This tuning option is not available in DB2 servers for OS/390, OS/400, VM or VSE.
Application Bank Report	Yes
Area	JDBC

Metric I263_JDBCConnPoolUtil

Monitor Template Name	WBSSPI_0263
Metric Name	I263_JDBCConnPoolUtil
Metric Type	Alarming and Reporting
Description	Percentage of connection pool in use.
Impact	High (WebSphere version 4.x, 5)
PMI Module	connectionPoolModule
Severity: Condition	WBSSPI-0263.1: Critical, 98 WBSSPI-0263.2: Major, 95
Collection Interval	5m
Message Group	WebSphere
Message Text	WBSSPI-0263.1: % utilization of a connection pool (<$VALUE>%) too high (>=<$THRESHOLD>%)
Instruction Text	The percent utilization of the connection pool has exceeded a threshold value. Suggested action: Open the Performance Tuner Wizard by clicking Console → Wizards → Performance Tuner from the Administrative Console. Data Source: Connection Pool Size Each data source contains a pool of connections to the corresponding database. An upper bound for your application comes from the sum of the number of Web container threads and ORB threads. The servlet contribution to the connection pool size can be significantly smaller than the Web container maximum threads, if only a small percentage of servlet requests use these database connections. Data Source: Prepared Statement Cache Size Make your prepared statement cache large enough for all prepared statements, by setting the cache size to the product of: The number of SQL prepared statements in your application The maximum number of configured data source connections Database: (DB2 Only) This panel is only available for DB2 databases. Tune the database after this wizard has completed, by selecting Tune database and entering the DB2 SYSADM ID and password. If you provided a cataloged database alias name when configuring the data source and this alias name differs from the real database name, enter the real database name in the field provided. After database tuning is selected, and you have completed the remaining panels, click Finish on the Summary panel. The tuning wizard then calls the DB2SmartGuide API, to tune the DB2 database associated with the data source. Stop and restart the database instance, for the DB2SmartGuide changes to take effect.

Monitor Template Name	WBSSPI_0263
Instruction Text (cont.)	Note: DB2SmartGuide tuning works better if the database is already populated. It is not necessary or recommended to tune the repository database (WAS).
	Before tuning a database, you might want to use the DB2 DBA utility to back up (db2cfexp) the database configuration. If the tuning fails, you can then restore (db2cfimp) your database configuration. You can also use:
	DB2 RESET DATABASE CONFIGURATION FOR database-name to restore the database to default values DB2 RESET DATABASE MANAGER CONFIGURATION to restore the DBM to default values
	The database being tuned must reside on a DB2 Server at Version 7.2.1, or higher. This tuning option is not available in DB2 servers for OS/390, OS/400, VM or VSE.
Application Bank Report	Yes
Area	JDBC

Metric I264_JDBCConnPoolMaxPct

Monitor Template Name	WBSSPI_0264
Metric Name	I264_JDBCConnPoolMaxPct
Metric Type	Alarming
Description	Percentage of time that all connections in a pool are in use.
Impact	High (WebSphere version 4.x, 5)
PMI Module	connectionPoolModule
Severity: Condition	WBSSPI-0264.1: Critical, 98 WBSSPI-0264.2: Major, 95
Collection Interval	5m
Message Group	WebSphere
Message Text	WBSSPI-0264.1: % of time all connections in a pool are in use (<$VALUE>%) too high (>=<$THRESHOLD>%)
Instruction Text	The percent of time that all connections in a pool are in use has exceeded a threshold value.
	Suggested action: Open the Performance Tuner Wizard by clicking Console → Wizards → Performance Tuner from the Administrative Console.
	Data Source: Connection Pool Size
	Each data source contains a pool of connections to the corresponding database. An upper bound for your application comes from the sum of the number of Web container threads and ORB threads.
	The servlet contribution to the connection pool size can be significantly smaller than the Web container maximum threads, if only a small percentage of servlet requests use these database connections.
	Data Source: Prepared Statement Cache Size
	Make your prepared statement cache large enough for all prepared statements, by setting the cache size to the product of:
	The number of SQL prepared statements in your application The maximum number of configured data source connections
	Database: (DB2 Only) This panel is only available for DB2 databases. Tune the database after this wizard has completed, by selecting Tune database and entering the DB2 SYSADM ID and password. If you provided a cataloged database alias name when configuring the data source and this alias name differs from the real database name, enter the real database name in the field provided.
	After database tuning is selected, and you have completed the remaining panels, click Finish on the Summary panel. The tuning wizard then calls the DB2SmartGuide API, to tune the DB2 database associated with the data source. Stop and restart the database instance, for the DB2SmartGuide changes to take effect.

Monitor Template Name	WBSSPI_0264
Instruction Text (cont.)	Note: DB2SmartGuide tuning works better if the database is already populated. It is not necessary or recommended to tune the repository database (WAS). Before tuning a database, you might want to use the DB2 DBA utility to back up (db2cfexp) the database configuration. If the tuning fails, you can then restore (db2cfimp) your database configuration. You can also use: DB2 RESET DATABASE CONFIGURATION FOR database-name to restore the database to default values DB2 RESET DATABASE MANAGER CONFIGURATION to restore the DBM to default values. The database being tuned must reside on a DB2 Server at Version 7.2.1, or higher. This tuning option is not available in DB2 servers for OS/390, OS/400, VM or VSE.
Application Bank Report	Yes
Area	JDBC

Metric I065_JDBCConPoolTimRt

Monitor Template Name	N/A—Used to generate a graph
Metric Name	I065_JDBCConPoolTimRt
Metric Type	Graphing
Description	Number of times a client timed out waiting for a connection from the pool per minute.
Impact	Low (WebSphere version 4.x, 5)
PMI Module	connectionPoolModule
Severity: Condition with threshold	N/A
Collection Interval	5m
Message Group	N/A
Message Text	N/A
Instruction Text	N/A
Application Bank Report	Yes
Area	JDBC

Metric I265_JDBCConnPoolTimeoutRt

Monitor Template Name	WBSSPI_0265
Metric Name	I265_JDBCConnPoolTimeoutRt
Metric Type	Alarming and Reporting
Description	Number of times a client timed out waiting for a connection from the pool (drill down) per minute.
Impact	Low (WebSphere version 4.x, 5)
PMI Module	connectionPoolModule
Severity: Condition	WBSSPI-0265.1: Critical, 98 WBSSPI-0265.2: Major, 95
Collection Interval	5m
Message Group	WebSphere
Message Text	WBSSPI-0265.1: # of times a client timed out waiting for a connection per minute (<$VALUE>/min) too high (>=<$THRESHOLD>/min)
Instruction Text	The number of times a client timed out waiting for a connection from the connection pool has exceeded a threshold value. Suggested action: Open the Performance Tuner Wizard by clicking Console → Wizards → Performance Tuner from the Administrative Console. Data Source: Connection Pool Size Each data source contains a pool of connections to the corresponding database. An upper bound for your application comes from the sum of the number of Web container threads and ORB threads. The servlet contribution to the connection pool size can be significantly smaller than the Web container maximum threads, if only a small percentage of servlet requests use these database connections. Data Source: Prepared Statement Cache Size Make your prepared statement cache large enough for all prepared statements, by setting the cache size to the product of: The number of SQL prepared statements in your application The maximum number of configured data source connections *Database: (DB2 Only)* This panel is only available for DB2 databases. Tune the database after this wizard has completed, by selecting Tune database and entering the DB2 SYSADM ID and password. If you provided a cataloged database alias name when configuring the data source and this alias name differs from the real database name, enter the real database name in the field provided. After database tuning is selected, and you have completed the remaining panels, click Finish on the Summary panel. The tuning wizard then calls the DB2SmartGuide API, to tune the DB2 database associated with the data source. Stop and restart the database instance, for the DB2SmartGuide changes to take effect.

Monitor Template Name	WBSSPI_0265
Instruction Text (cont.)	Note: DB2SmartGuide tuning works better if the database is already populated. It is not necessary or recommended to tune the repository database (WAS).
	Before tuning a database, you might want to use the DB2 DBA utility to back up (db2cfexp) the database configuration. If the tuning fails, you can then restore (db2cfimp) your database configuration. You can also use: DB2 RESET DATABASE CONFIGURATION FOR database-name to restore the database to default values DB2 RESET DATABASE MANAGER CONFIGURATION to restore the DBM to default values.
	The database being tuned must reside on a DB2 Server at Version 7.2.1, or higher. This tuning option is not available in DB2 servers for OS/390, OS/400, VM or VSE.
Application Bank Report	Yes
Area	JDBC

Metric I066_JDBConPoolThru

Monitor Template Name	N/A—Used to generate a report and graph
Metric Name	I066_JDBConPoolThru
Metric Type	Graphing and Reporting
Description	Number of connections allocated and returned by applications per second.
Impact	Low (WebSphere version 4.x, 5)
PMI Module	connectionPoolModule
Severity: Condition	N/A
Collection Interval	5m
Message Group	N/A
Message Text	N/A
Instruction Text	N/A
Application Bank Report	N/A
Area	JDBC

Metric I266_JDBConnPoolThroughput

Monitor Template Name	WBSSPI_0266
Metric Name	I266_JDBConnPoolThroughput
Metric Type	Alarming and Reporting
Description	Number of connections allocated and returned by applications per second (drill down).
Impact	Low (WebSphere version 4.x, 5)
PMI Module	connectionPoolModule
Severity: Condition	WBSSPI-0266.1: Warning threshold, 10000
Collection Interval	5m
Message Group	WebSphere
Message Text	WBSSPI-0266.1: # of connections allocated and returned by applications (<$VALUE>/sec) too high (>=<$THRESHOLD>/sec)
Instruction Text	The number of connections allocated and returned by applications per second has exceeded a threshold value. **Suggested action**: Open the Performance Tuner Wizard by clicking Console → Wizards → Performance Tuner from the Administrative Console. Data Source: Connection Pool Size Each data source contains a pool of connections to the corresponding database. An upper bound for your application comes from the sum of the number of Web container threads and ORB threads. The servlet contribution to the connection pool size can be significantly smaller than the Web container maximum threads, if only a small percentage of servlet requests use these database connections. Data Source: Prepared Statement Cache Size Make your prepared statement cache large enough for all prepared statements, by setting the cache size to the product of: The number of SQL prepared statements in your application The maximum number of configured data source connections Database: (DB2 Only) This panel is only available for DB2 databases. Tune the database after this wizard has completed, by selecting Tune database and entering the DB2 SYSADM ID and password. If you provided a cataloged database alias name when configuring the data source and this alias name differs from the real database name, enter the real database name in the field provided. After database tuning is selected, and you have completed the remaining panels, click Finish on the Summary panel. The tuning wizard then calls the DB2SmartGuide API, to tune the DB2 database associated with the data source. Stop and restart the database instance, for the DB2SmartGuide changes to take effect.

Monitor Template Name	WBSSPI_0266
Instruction Text (cont.)	Note: DB2SmartGuide tuning works better if the database is already populated. It is not necessary or recommended to tune the repository database (WAS).
	Before tuning a database, you might want to use the DB2 DBA utility to back up (db2cfexp) the database configuration. If the tuning fails, you can then restore (db2cfimp) your database configuration. You can also use: DB2 RESET DATABASE CONFIGURATION FOR database-name to restore the database to default values DB2 RESET DATABASE MANAGER CONFIGURATION to restore the DBM to default values
	The database being tuned must reside on a DB2 Server at Version 7.2.1, or higher. This tuning option is not available in DB2 servers for OS/390, OS/400, VM or VSE.
Application Bank Report	Yes
Area	JDBC

Metric I070_TranGlobDur

Monitor Template Name	WBSSPI_0070
Metric Name	I070_TranGlobDur
Metric Type	Alarming and Graphing
Description	Average duration of global transactions.
Impact	High (WebSphere version 4.x); Medium (WebSphere version 5)
PMI Module	transactionModule
Severity: Condition	WBSSPI-0070.1: Warning threshold, 1000
Collection Interval	5m
Message Group	WebSphere
Message Text	WBSSPI-0070.1: Ave. duration of a global transaction (<$VALUE>ms) too high (>=<$THRESHOLD>ms)
Instruction Text	The average duration of global transactions has exceeded a threshold value. **Suggested action**: Use this metric to monitor the server load over time. Slower transaction durations may indicate increased server load and/or increased resource contention. WebSphere keeps transaction performance data separately for global and local transactions. Local transactions are limited to a single server and its associated resource manager. Global transactions are controlled by an external transaction manager and can span multiple servers.
Application Bank Report	Yes
Area	Transactions

Metric I071_TranLocDur

Monitor Template Name	WBSSPI_0071
Metric Name	I071_TranLocDur
Metric Type	Alarming and Graphing
Description	Average duration of local transactions.
Impact	High (WebSphere version 4.x); Medium (WebSphere version 5)
PMI Module	transactionModule
Severity: Condition	WBSSPI-0071.1: Warning threshold, 1000
Collection Interval	5m
Message Group	WebSphere
Message Text	WBSSPI-0071.1: Ave. duration of a local transaction (<$VALUE>ms) too high (>=<$THRESHOLD>ms)
Instruction Text	The average duration of local transactions has exceeded a threshold value. **Suggested action**: Use this metric to monitor the server load over time. Slower transaction durations may indicate increased server load and/or increased resource contention. WebSphere keeps transaction performance data separately for global and local transactions. Local transactions are limited to a single server and its associated resource manager. Global transactions are controlled by an external transaction manager and can span multiple servers.
Application Bank Report	Yes
Area	Transactions

Metric I072_TranGlobCommDur

Monitor Template Name	WBSSPI_0072
Metric Name	I072_TranGlobCommDur
Metric Type	Alarming and Graphing
Description	Average duration of commits for global transactions.
Impact	Medium (WebSphere version 4.x, 5)
PMI Module	transactionModule
Severity: Condition	WBSSPI-0072.1: Warning threshold,
Collection Interval	5m
Message Group	WebSphere
Message Text	WBSSPI-0072.1: Ave. duration of a commit for a global transaction (<$VALUE>ms) too high (>=<$THRESHOLD>ms)
Instruction Text	The average duration of commits for global transactions has exceeded a threshold value. **Suggested action**: Use this metric to monitor the server load over time. Slower transaction durations may indicate increased server load and/or increased resource contention. WebSphere keeps transaction performance data separately for global and local transactions. Local transactions are limited to a single server and its associated resource manager. Global transactions are controlled by an external transaction manager and can span multiple servers.
Application Bank Report	Yes
Area	Transactions

Metric I073_TranLocCommDur

Monitor Template Name	WBSSPI_0073
Metric Name	I073_TranLocCommDur
Metric Type	Alarming and Graphing
Description	Average duration of commits for local transactions.
Impact	Medium (WebSphere version 4.x, 5)
PMI Module	transactionModule
Severity: Condition	WBSSPI-0073.1: Warning threshold, 1000
Message Group	WebSphere
Collection Interval	5m
Message Text	WBSSPI-0073.1: Ave. duration of a commit for a local transaction (<$VALUE>ms) too high (>=<$THRESHOLD>ms)
Instruction Text	The average duration of commits for local transactions has exceeded a threshold value. **Suggested action**: Use this metric to monitor the server load over time. Slower transaction durations may indicate increased server load and/or increased resource contention. WebSphere keeps transaction performance data separately for global and local transactions. Local transactions are limited to a single server and its associated resource manager. Global transactions are controlled by an external transaction manager and can span multiple servers.
Application Bank Report	Yes
Area	Transactions

Metric I074_TranRollbackRt

Monitor Template Name	WBSSPI_0074
Metric Name	I074_TranRollbackRt
Metric Type	Alarming and Graphing
Description	Number per second of global and local transactions rolled back.
Impact	Low (WebSphere version 4.x, 5)
PMI Module	transactionModule
Severity: Condition	WBSSPI-0074.1: Warning threshold, 1000
Collection Interval	5m
Message Group	WebSphere
Message Text	WBSSPI-0074.1: # of global and local transactions rolled back (<$VALUE>/sec) too high (>=<$THRESHOLD>/sec)
Instruction Text	The number of global and local transactions rolled back per second has exceeded a threshold value. **Suggested action**: This metric includes both global and local transactions. Local transactions are limited to a single server and its associated resource manager. Global transactions are controlled by an external transaction manager and can span multiple servers. When a transaction commits, all actions associated with that transaction are written to a log. In the event of system problems, those actions are repeated if necessary when the system's recovery mechanism replays the log. When a transaction aborts, any changes made by the transaction are undone. After a transaction is undone (rolled back), the only remaining evidence of the transaction is in the transaction processing system's log. Timeouts associated with transactions usually prevent any one transaction from holding resources at a server for too long. For example, if two transactions are competing for the same resource (one holds a lock on a resource and the other is requesting that lock, and the lock modes conflict), timeouts will eventually abort one of the transactions. The idle timeout will abort a transaction that is inactive too long, and the operation timeout will abort an active transaction that is taking too long.
Application Bank Report	Yes
Area	Transactions

Metric I075_TranTimeoutRt

Monitor Template Name	WBSSPI_0075
Metric Name	I075_TranTimeoutRte
Metric Type	Alarming and Graphing
Description	Number per second of timed out global and local transactions.
Impact	Low (WebSphere version 4.x, 5)
PMI Module	transactionModule
Severity: Condition	WBSSPI-0075.1: Warning threshold, 1000
Collection Interval	5m
Message Group	WebSphere
Message Text	N/A
Instruction Text	The number of global and local transactions that timed out per second has exceeded a threshold value. **Suggested action**: This metric includes both global and local transactions. Local transactions are limited to a single server and its associated resource manager. Global transactions are controlled by an external transaction manager and can span multiple servers. When a transaction commits, all actions associated with that transaction are written to a log. In the event of system problems, those actions are repeated if necessary when the system's recovery mechanism replays the log. When a transaction aborts, any changes made by the transaction are undone. After a transaction is undone (rolled back), the only remaining evidence of the transaction is in the transaction processing system's log. Timeouts associated with transactions usually prevent any one transaction from holding resources at a server for too long. For example, if two transactions are competing for the same resource (one holds a lock on a resource and the other is requesting that lock, and the lock modes conflict), timeouts will eventually abort one of the transactions. The idle timeout will abort a transaction that is inactive too long, and the operation timeout will abort an active transaction that is taking too long.
Application Bank Report	Yes
Area	Transactions

Metric I076_TranCommitRt

Monitor Template Name	WBSSPI_0076
Metric Name	I076_TranCommitRt
Metric Type	Alarming and Graphing
Description	Number per second of global and local transactions that were committed.
Impact	Low (WebSphere version 4.x, 5)
PMI Module	transactionModule
Severity: Condition	WBSSPI-0076.1: Warning threshold, 1000
Collection Interval	5m
Message Group	WebSphere
Message Text	WBSSPI-0076.1: # of global and local transactions that were committed (<$VALUE>/sec) too high (>=<$THRESHOLD>/sec)
Instruction Text	The number of global and local transactions that were committed per second has exceeded a threshold value. **Suggested action**: This metric indicates the rate (number per second) of transactions that are successfully committed on the server. Use this information for capacity planning. This metric includes both global and local transactions. Local transactions are limited to a single server and its associated resource manager. Global transactions are controlled by an external transaction manager and can span multiple servers.
Application Bank Report	Yes
Area	Transactions

Metric I077_TranThruput

Monitor Template Name	WBSSPI_0077
Metric Name	I077_TranThruput
Metric Type	Reporting
Description	Number per second of global and local transactions that were committed.
Impact	Low (WebSphere version 4.x, 5)
PMI Module	transactionModule
Severity: Condition	N/A
Collection Interval	5m
Message Group	N/A
Message Text	N/A
Instruction Text	N/A
Application Bank Report	N/A
Area	Transactions

Metric I078_TranStartRt

Monitor Template Name	WBSSPI_0078
Metric Name	I078_TranStartRt
Metric Type	Alarming and Graphing
Description	Number per second of global and local transactions that were begun.
Impact	Low (WebSphere version 4.x, 5)
PMI Module	transactionModule
Severity: Condition	WBSSPI-0078.1: Warning threshold
Collection Interval	5m
Message Group	WebSphere
Message Text	WBSSPI-0078.1: # of global and local transactions that were begun (<$VALUE>/sec) too high (>=<$THRESHOLD>/sec)
Instruction text	The number of global and local transactions that were begun per second has exceeded a threshold value. **Suggested action**: This metric indicates the rate (number per second) of transactions that are begun on this server. Use this information for capacity planning.
Application Bank Report	Yes
Area	Transactions

2 WBS-SPI Log File and Configuration File Templates

This section shows the HP OpenView Smart Plug-in for WebSphere Application Server (WBS-SPI) templates that monitor logged information and modifications that occur in both WebSphere and the WBS-SPI files. These templates detect error messages internal to the WBS-SPI, as well as changes made to the WebSphere XML configuration files.

WBSSPI-Error Log

Description	Monitors the WBS-SPI error log and captures critical errors, which it sends to the Message Browser.
Severity	Critical
Message Group	WBSSPI
Help Text	Available for each error as detected: WASSPI-1 through WASSPI-233. See the *HP OpenView Smart Plug-in for WebSphere Application Server Configuration Guide* for detailed help text for each error with a WASSPI prefix.

WebSphere Config Files

Description	Detects changes in the WebSphere xml configuration file.
Severity	Critic Warningal
Message Group	WebSphere
Help Text	One of the WebSphere properties files has been saved. Very likely, the file contents has changed. The name of the updated configuration file is listed in field 'Object' of this OVO message. **Probable Cause**: Re-configuration of WebSphere. **Potential Impact**: When starting WebSphere, the new configuration will be used. If some of the changes are not correct, this might lead to problems. **Suggested Action**: Review the updated properties file. If the changes to the configuration file are correct, then you can acknowledge this message.

WebSphere Log Template

Description	Catches critical errors and warnings in the WebSphere log file.
Severity	Critical Warning
Message Group	WebSphere
Help Text	**Probable Cause**: A message with the indicator 'Emergency' or 'Critical' was detected in the WebSphere logfile. OR A message with the indicator 'Notice,' 'Error' or 'Alert' was detected in the WebSphere logfile. **Suggested Action**: Examine the error and use the WebSphere documentation (manuals or online Help) to determine the exact cause and action to take.

Index

E

EJB
active in pool, 28
active in pool, graphing, 29
average number in pool, 42
database rate, 37, 38
I020_EJBPoolUtil, 28
I022_EJBMethCallsRt, 33
I024_EJBEntDatLdStRt, 37
I025_EJBPoolMissPct, 40
I026_EJBConcLives, 42
I220_EJBPoolUtil, 29
I221_EJBMethRespTime, 31
I222_EJBMethodCallsRt, 34
I223_EJBPoolSize, 36
I224_EJBEntDataLdStRt, 38
I225_EJBPoolMissPct, 41
method call rate, 33, 34
pool retrieval failure, 40, 41
pool size, 36
response time, 31

errors
WBSSPI-Error Log, 85

G

graphing
EJB active in pool, 28
EJB average number in pool, 42
EJB database rate, 37
EJB method call rate, 33
EJB pool retrieval failure, 40
global commit duration, 78
global transaction duration, 76
I013_ThreadPoolPctMax, 25
I014_ThrdPoolCrtRt, 27
I020_EJBPoolUtil, 28
I022_EJBMethCallsRt, 33
I024_EJBEntDatLdStRt, 37
I025_EJBPoolMissPct, 40
I026_EJBConcLives, 42
I040_ServSessAverageLife, 44
I041_ServSessActSess, 46
I042_ServInvSessRt, 48
I045_WebAppServReqRt, 50
I047_WebAppServErrRt, 53
I048_WebAppServLoad, 56
I049_WebAppServRelRt, 57
I061_JDBCConPoolWait, 60
I062_JDBCConPoolWtTime, 63
I065_JDBCConPoolTimRt, 70
I066_JDBCConPoolThru, 73
I070_TranGlobDur, 76
I071_TranLocDur, 77
I072_TranGlobCommDur, 78
I073_TranLocCommDur, 79
I074_TranRollbackRt, 80
I075_TranTimeoutRt, 81
I076_TranCommRt, 82
I078_TranStartRt, 84
invalid sessions, 48
JDBC connections allocated, 73
JDBC time out, 70
JDBC wait time, 63
local commit duration, 79
local transaction duration, 77
maximum threads reached, 25
servlet average life, 44
servlet errors, 53
servlet loaded, 56
servlet reloaded, 57
servlet requests, 50
servlet sessions accessed, 46
threads created, 27
threads waiting, 60
transactions begun, 84
transactions committed, 82
transactions rolled back, 80
transactions time out, 81