



IBM Software Group

IBM WebSphere Application Server V6

Performance Tools



@business on demand.

© 2005 IBM Corporation
Updated May 12, 2015

This presentation, will focus on WebSphere® Application Server's Performance Tools.

Goals

- Introduce Performance Monitoring Infrastructure (PMI)
- Describe Tivoli Performance Viewer (TPV) and its feature set
- Describe Performance Advisors and their use



The goals for this presentation are to discuss the performance tools that WebSphere Application Server provides to analyze its current performance. The goal of the presentation is also to point out any differences between V5 and V6 along the way.

Agenda

- WebSphere Application Server Environment
- Performance Monitoring Infrastructure (PMI)
- Tivoli Performance Viewer
- Performance Advisors
 - ▶ Tivoli Performance Viewer Performance Advisor
 - ▶ Runtime Performance Advisor



The agenda for this presentation is to first discuss a typical WebSphere Application Server environment and then discuss each of the performance tools and their roles in analyzing performance.

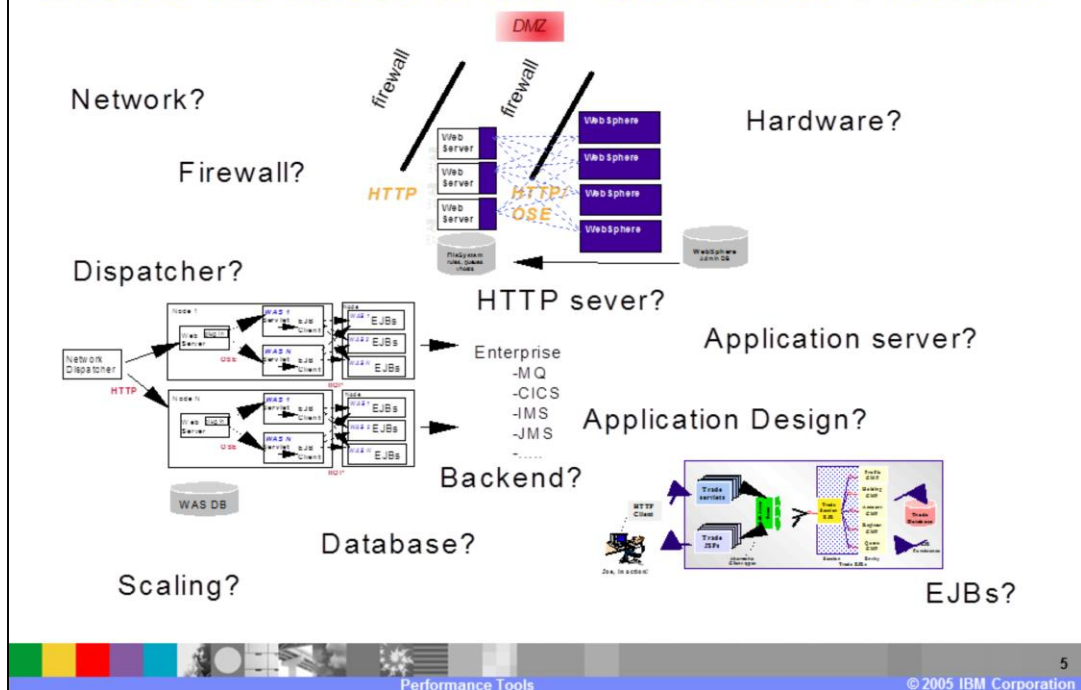
Section

WebSphere Application Server Environment



The next section will discuss a typical WebSphere Application Server environment.

What is the Cause of the Performance Problem?



Where do the performance problems within a business application's environment lie? In most business application environments this question can not be easily answered, because of the complexity of the environment. The problem could actually lie in one or several areas. As seen in the diagram your application's problems may be network or hardware related, backend system related, actual product bugs or, quite often, an application design issue. This is where WebSphere Application Server's performance tools come into the picture. They allow an administrator to visualize and analyze performance data collected from various areas of your business application environment. The administrator can make an informed decision when it comes to diagnosing and performance tuning their environment.

Performance Tools Today

- Data
- Viewer
- Advisors



User View



WebSphere Application Server provides all of these things to the administrator. The Performance Monitoring Infrastructure (PMI) externalizes the performance data. The Tivoli Performance Viewer that is integrated with the Administrative Console, allows the Administrator to view the collected PMI data. And the Performance Advisors provided in the Administrative Console or Tivoli Performance Viewer, provide configuration advice based on the collected PMI data and WebSphere Application Server's current configuration.

Section

Performance Monitoring Infrastructure (PMI)



The next section will discuss the Performance Monitoring Infrastructure or PMI service.

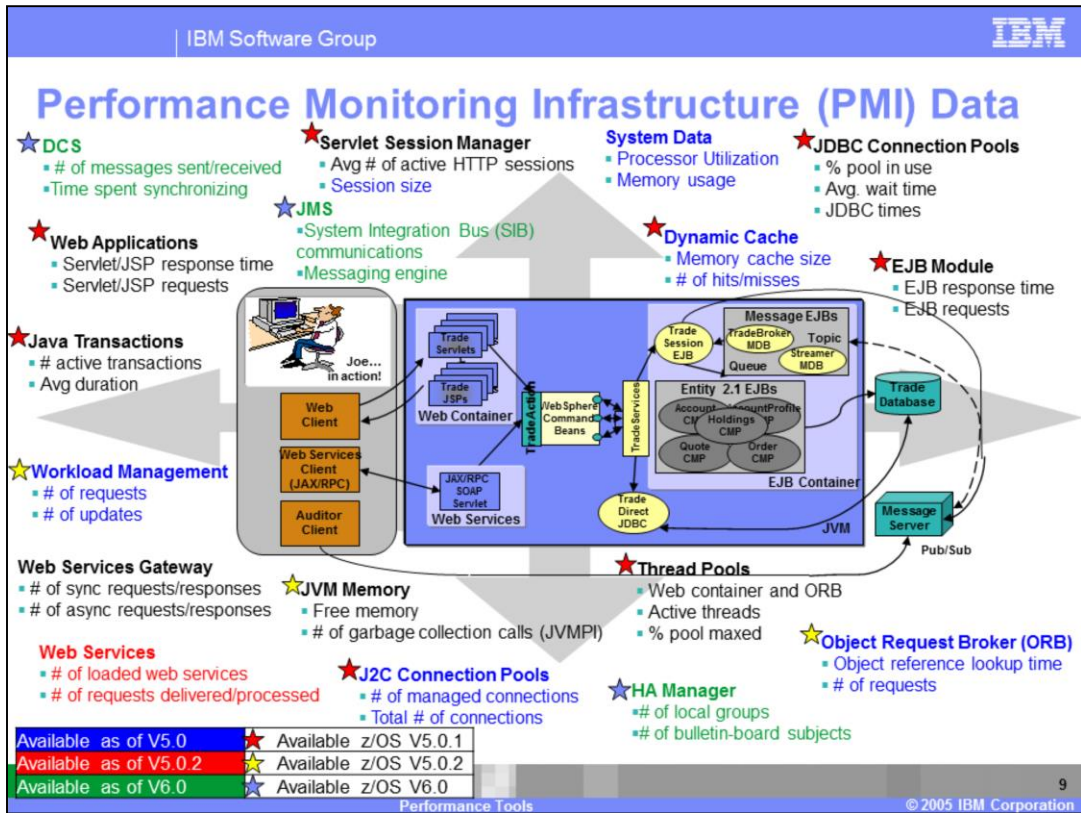
Performance Monitoring Infrastructure (PMI)

- Monitoring infrastructure that collects performance data from a running Application Server
- Performance data collected on
 - ▶ Customer's application resources
 - Example: Custom PMI, EJBs, Servlets/JSPs, Web Services, etc.
 - ▶ Application Server's run-time resources
 - Example: Java™ Virtual Machine (JVM) memory, thread pools, database connection pools, etc.
 - ▶ System resources
 - Example: Processor usage, total free memory, etc.
 - ▶ Detailed list of all metrics listed in Information Center



The Performance Monitoring Infrastructure (PMI) allows the Application Server to externalize its performance data to monitoring tools that can use the data to visualize and analyze its current performance.

The performance data that is generated falls into three general categories. The first category that collected PMI data falls into is customer application performance data. This includes information about your installed enterprise beans, servlets, Web Services, or even custom defined PMI data. Custom PMI data will be further discussed in an upcoming slide. The next category that PMI data falls into is the Application Server's run-time resources. This includes information from the Application Server's JVM, active thread pools and database connection pools and most of the other resources that you have defined within the Application Server. The last category that PMI data falls into are performance data collected from the system where the Application Server is currently running. This includes performance data on the system's processor usage or the total amount of free memory on the system. To see a complete list of all of the PMI data that is available from a WebSphere Application Server, visit the WebSphere Application Server Information Center.



The graphic above gives you a view into the amount of PMI data that is collected by the Application Server. The graphic does not depict the complete list of PMI metrics that are collected by the Application Server environment. The diagram also attempts to depict when PMI data was added across the different releases of the distributed and z/OS® WebSphere platforms. Again for a complete list of available PMI data, visit the WebSphere Application Server Information Center. The Information Center also provides information about the release in which a PMI metric was added.

V6 Enhancements



- Implements J2EE 1.4 Performance Data Framework
 - ▶ PMI Client API deprecated (around for next 3 releases)
- Custom PMI
 - ▶ Allows customer to add own performance metrics
- Fine-grained control
 - ▶ V5 and earlier used performance impact levels (none, low, medium, high, maximum)
 - ▶ V6 eliminates monitoring levels – enable/disable individual performance counters



There have been a number of improvements to the usability and functionality of the PMI service in V6. WebSphere Application Server now implements the J2EE 1.4 Performance Data Framework. With the addition of the new standardized API, the older, proprietary PMI client API that has existed since WebSphere Application Server V3.5.5 will be deprecated starting in V6. This means that the API will be available to WebSphere customers for the next 3 releases of the product.

Additionally the customer will have the ability to add their own custom PMI counters to the Application Server in V6. This functionality allows a customer to define their own PMI counters based on their application. For example, a PMI counter that computes the shares of stock sold per day could be added to the Application Server for a brokerage application. Care should be taken, however, to not redefine PMI counters that are available by default through the Application Server. It is important to keep in mind that the more performance counters that are defined within the Application Server, the larger the performance impact on your Application Server.

The ability to select individual PMI counters has been added in V6. Past versions of WebSphere Application Server used performance impact levels to enable or disable groups of PMI counters. Remember that performance impact levels were the low, medium, high and maximum PMI levels that could be set for the PMI service. WebSphere Application Server V6 provides the ability to enable or disable individual PMI counters instead of groups of PMI counters. The fine-grained control is available through both the Administrative Console and the command line.

V6 Enhancements (cont.)



- PMI grouping has changed
 - ▶ V5 and earlier grouped PMI metrics according to performance impact levels
 - Ex: Setting PMI monitoring to High enabled collection of PMI counters with performance impact level of high or less
 - ▶ V6 organizes PMI metrics into categories
 - Basic – J2EE components, processor usage, HTTP session info
 - Extended – Basic + additional Application Server resources (WLM, Dynamic Cache, etc.)
 - Custom – Fine-grained control
- Sequential update
 - ▶ Causes all PMI counters to be updated sequentially
 - ▶ Enabling adds additional overhead



The way in which groups of PMI counters are selected has also changed in WebSphere Application Server V6. Previous versions of WebSphere Application Server used performance impact levels to enable or disable groups of PMI counters. WebSphere Application Server V6 has redefined the organization of PMI counters into different categories. In the Administrative Console you can either enable individual PMI counters through the Custom setting or you can enable or disable groups of PMI metrics using the defined categories. The categories that exist are Basic, Extended and All. The Basic setting enables the collection of performance data on the J2EE components, processor usage and HTTP session information. The Extended setting extends the information collected by the Basic setting by additionally collecting performance data on Application Server resources.

You can now additionally enable whether all PMI counters are updated sequentially. Sequential updates of counters provides more accurate performance data, but at the same time places additional overhead on the server. By default the Application Server only sequentially updates the RangeStatistic and BoundedRangeStatistic. When sequential update is enabled, all Performance Data Framework statistic types are updated sequentially. For more information on the sequential update option visit the WebSphere Application Server Information Center.

Performance Impacts

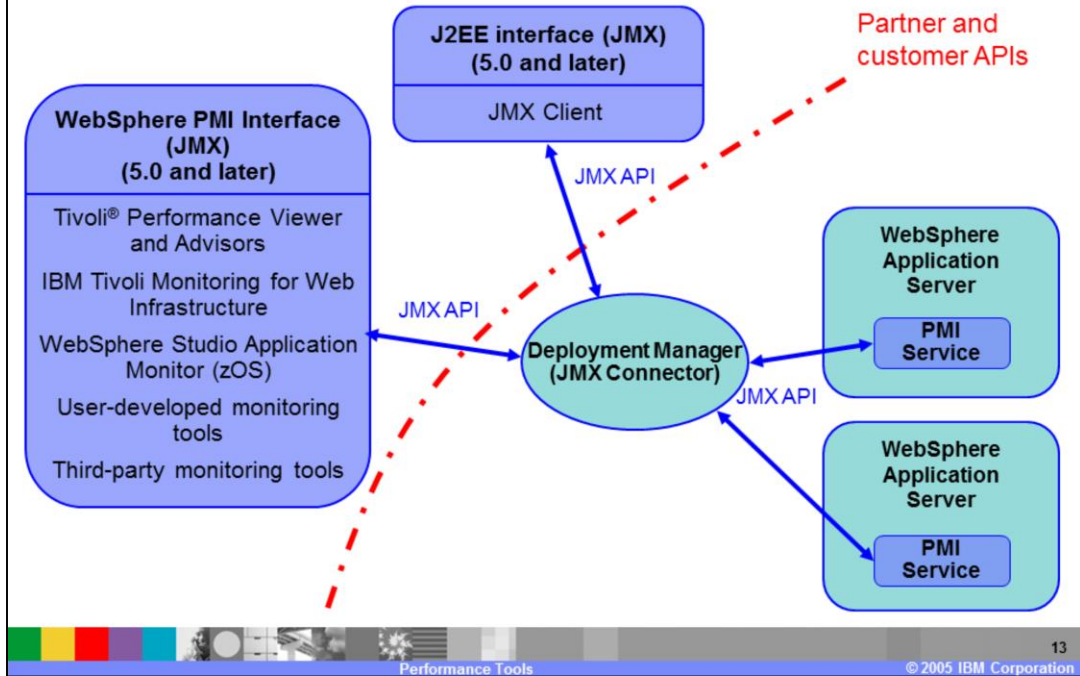
- Varies depending on counter
 - ▶ All data collection impacts performance in some way
- Basic setting has overhead of < 2%
 - ▶ Basic setting is enabled by default
- Extended setting has overhead of < 3%
- All setting has overhead of < 6%
- Collection of JVMPi data adds additional overhead
 - ▶ Collecting just GC and Thread data adds < 2%
 - ▶ Collecting all JVMPi data adds 20-25%



A performance impact is incurred on the Application Server when PMI data is enabled. The extent of the performance impact is determined by the amount and type of PMI counters that are collected. Counters that add up the amount of times a method is called will cause less of a performance impact than a counter that averages the amount of time spent executing a method call.

You can expect a performance impact of less than 2% when you have the Basic PMI setting enabled. It is also important to note, that the PMI service is enabled by default and set to the Basic setting. This is a change from previous versions of WebSphere Application Server. A performance impact of less than 3% can be expected for the Extended setting. And when all PMI counters are being collected you can expect a performance impact of less than 6%. When collecting additional PMI data from the Application Server's JVM using the Java Virtual Machine Profiler Interface (JVMPi), you can expect additional overhead over the value stated above. Collecting just garbage collection and thread PMI data adds an additional performance impact of less than 2% to your Application Server. This means that if you are already collecting PMI data using the Basic setting and you additionally collect JVM garbage collection and thread data, your Application Server would incur a performance impact of approximately 4%. Additionally if you collect all JVMPi PMI data, your Application Server will experience an additional performance impact of approximately 20 to 25%. Because of the JVMPi PMI module's high performance impact on your Application Server it should only be used in a development environment or in extreme cases in your production environment.

Performance Monitoring Applications



Performance data can be retrieved from the Application Server using the JMX interface. This functionality has remained the same from WebSphere Application Server V5. Also note that the picture does not show the PMI client interface. Remember that the PMI client interface will be available for the next 3 releases of WebSphere Application Server. Performance monitoring products should be developed against the JMX interface.

The performance monitoring solutions that were available for previous releases of WebSphere Application Server remain the same for this release. Take note, however, that the products are being moved to the JMX interface for compliance with WebSphere Application Server.


Section

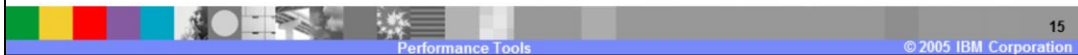
Tivoli Performance Viewer (TPV)



The next section will discuss the Tivoli Performance Viewer or TPV that is provided with WebSphere Application Server.

Tivoli Performance Viewer (TPV)

- Displays PMI data collected from local/remote Application Servers
 - ▶ Summary reports show key areas of contention
 - ▶ Graphical/tabular views of raw PMI data
 - ▶ Optionally save collected PMI data to logs
- Provides configuration advice via Performance Advisor section
 - ▶ Tuning advice formulated from gathered PMI and configuration data
- Integrated into Administrative Console 



The Tivoli Performance Viewer or TPV is a PMI client that is provided with WebSphere Application Server. TPV is responsible for collecting and visualizing the PMI data that is being generated by the Application Server. TPV's functionality has not changed from WebSphere Application Server V5. TPV still allows the user to view the summary views that show key areas of the Application Server such as the web container. It also allows the user to visualize the specific PMI counters in graph or table form. And lastly it allows a user to record performance data to a performance log. TPV also provides a second Performance Advisor. The TPV Performance Advisor will be discussed in an upcoming section.

The key thing to note about TPV in V6 is that it is now included in the Administrative Console. The thin-client version of the Tivoli Performance Viewer has been removed from WebSphere Application Server and incorporated into the Administrative Console. This gives users the capability to monitor their Application Server's performance from any computer that has a web browser.

Tivoli Performance Viewer

Access TPV through the Administrative Console

View panel for TPV – displays summary reports, graphs, and tables

Navigation panel for TPV

Select	Name	Application
<input type="checkbox"/>	ole.environment/about.jsp	nconsole#adminconsole.var
<input type="checkbox"/>	le.perf.pmi/pmiConfig60.jsp	nconsole#adminconsole.var
<input type="checkbox"/>	le.perf.pmi/pmiFrames.jsp	nconsole#adminconsole.var
<input type="checkbox"/>	le.perf.pmi/pmiNavTree.jsp	nconsole#adminconsole.var
<input type="checkbox"/>	ef.pmi/refreshScript.jsp	nconsole#adminconsole.var
<input type="checkbox"/>	visedConfigBoxLayout.jsp	nconsole#adminconsole.var
<input type="checkbox"/>	/revisedContentLayout.jsp	nconsole#adminconsole.var
<input type="checkbox"/>	dmin/confirmEnableRPA.jsp	nconsole#adminconsole.var
<input type="checkbox"/>	ontainedContentLayout.jsp	nconsole#adminconsole.var
<input type="checkbox"/>	obdetermination/blank.jsp	nconsole#adminconsole.var

Monitoring Data
Show Me

Logging Data
Show Me

Performance Tools © 2005 IBM Corporation

This screen capture shows where Tivoli Performance Viewer can be accessed from the Administrative Console. It also shows the new look-and-feel of TPV.

Demonstrations on how to configure and use the Tivoli Performance Viewer can be accessed from this slide by clicking the Show Me icons.

Usage

- Summary reports
 - ▶ Monitor server's real-time performance and health
 - ▶ See WebSphere Application Server Information Center for usage tips
- Performance modules
 - ▶ Drill down on specific areas of interest
- Logs
 - ▶ Save performance data for later analysis or problem determination



The Tivoli Performance Viewer can be used in several ways, a few of which are discussed in the slide. The summary reports should be used as your first source of information about the Application Server's current performance. They provide a window into how the server and its environment are doing. Summary reports are provided for servlets, EJB's and their methods, thread pools and JDBC connection pools. The WebSphere Application Server Information Center provides good hints and tips on how to effectively use these views to analyze your environment.

The graphs and tables that show the individual performance metrics should be used as your next step to analyzing your Application Server's performance. They allow you to drill down on specific PMI counters as well as correlate PMI counters against each other.

Lastly, you can use the logging functionality of TPV to record performance data for use in problem determination or analysis of your application's performance. It is also important to note that TPV should not be used as a long term performance logging solution. If you desire a long running solution you should look to IBM Tivoli's portfolio of applications to provide this functionality.

Section

Performance Advisors



The next section will discuss the Performance Advisors that are provided with WebSphere Application Server.



Performance Advisors

- Provide configuration advice based on collected PMI data
- Advice based on basic rules of thumb for tuning WebSphere Application Server
 - ▶ Rules are IBM defined and non-configurable
- **Does not** automatically tune Application Server based on advice
 - ▶ Administrator must manually apply recommendations
 - ▶ Suggested settings need to be checked against baseline performance to verify improvement



The Performance Advisors provide the user with performance advice based on PMI data collected from their running Application Server and their server's current configuration. The advice that is generated by the advisors is based on a predetermined rule set provided by IBM. The rules contained within the rule set were defined by the WebSphere Performance Team and are best practices for tuning your WebSphere Application Server. It is also important to note that the Performance Advisors do not automatically make changes to your Application Server's configuration, they just suggest potential changes that could be made to the configuration to possibly increase performance. The Application Server's administrator is still responsible for making the suggested changes within the Administrative Console. Caution should also be taken when implementing changes suggested by the Performance Advisors. All suggestions should be fully tested one-by-one to verify that performance is positively impacted.

The key thing to note here is that the TPV Performance Advisor is now included in the Administrative Console as well. This follows because TPV has moved within the Administrative Console.

Advisors		Runs when you ask for it • Calculation intensive operations	Runs in the background • The first step toward auto tuning
	Performance Advisor in Tivoli Performance Viewer	Runtime Performance Advisor	
Location of execution	Application Server	Application server	
Invocation of tool	Tivoli Performance Viewer	Administrative console	
Output	Tivoli Performance Viewer graphical user interface (GUI)	SystemOut.log file and the Administrative Console	
Frequency of operation	When you select refresh in Tivoli Performance Viewer	Configurable	
Types of advice	<ul style="list-style-type: none"> ▪ ORB service thread pools ▪ Web container thread pools ▪ Connection pool size ▪ Persisted session size and time ▪ Prepared statement cache size ▪ Session cache size ▪ Dynamic cache size ▪ JVM heap size ▪ DB2 Performance Configuration Wizard 	<ul style="list-style-type: none"> ▪ ORB service thread pools ▪ Web container thread pools ▪ Connection pool size ▪ Persisted session size and time ▪ Prepared statement cache size ▪ Session cache size 	
			

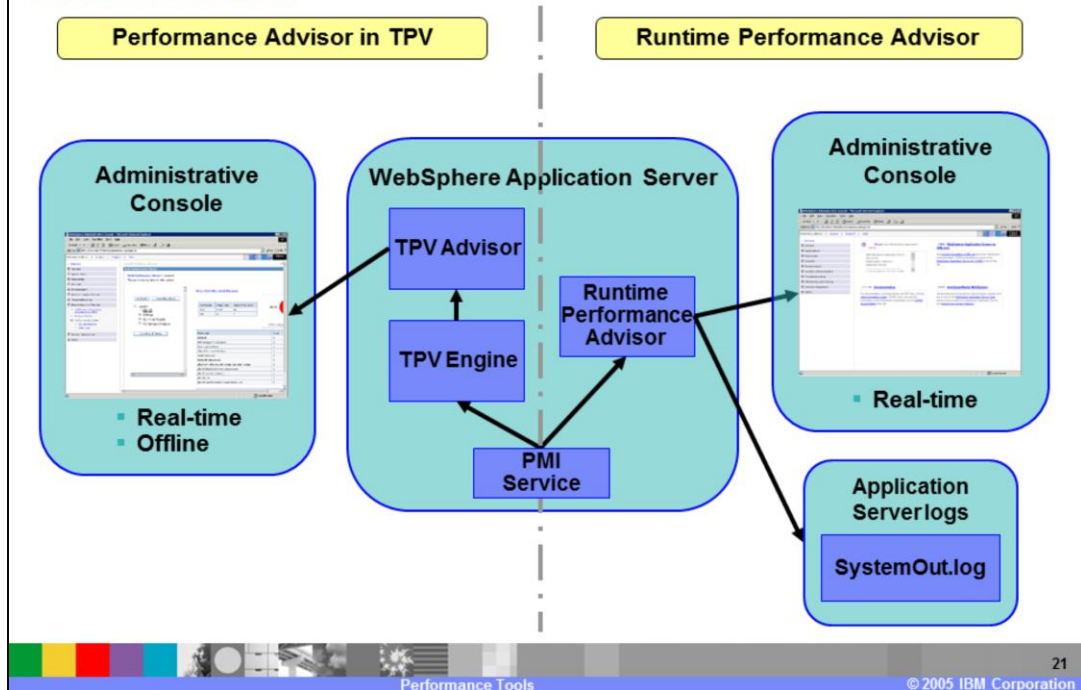
The Performance Advisors share a lot of similarities but differ in couple of key areas. The table in the slide points out these similarities and differences.

The Performance Advisors are the same in that they both reside within the Application Server's JVM, are invoked and configured from within the Administrative Console, and generate advice from the same rule set.

The Performance Advisors differ from each other in the following ways. The TPV Performance Advisor will only generate advice when you ask it to, while the Runtime Performance Advisor generates advice on a configured interval. This interval is configured from within the Administrative Console. Additionally the TPV Performance Advisor outputs its advice to a Performance Advisor section of the Tivoli Performance Advisor, while the Runtime Performance Advisor outputs advice to both your Application Server's SystemOut.log file and the Administrative Console. Lastly the Runtime Performance Advisor generates performance advice for only a subset of the advice categories supported by the TPV Performance Advisor. For example the Runtime Performance Advisor can not generate advice for the Application Server's dynamic cache settings.

Demonstrations on how to configure and use each of the Performance Advisors can be accessed from this slide by clicking the Show Me icons.

Architecture



The Performance Advisor's architecture has changed slightly because of the Tivoli Performance Viewer moving within the Administrative Console. This also means that the TPV Performance Advisor has also moved within the Administrative Console, and therefore runs within the Application Server's JVM. The architecture for the Runtime Performance Advisor stays the same from WebSphere Application Server V5.

Roadmap to Self-Optimizing

- The goal

- ▶ Self-optimizing Application Server

- ▶ Application Server is here

"Assisting users"

Level	1: Basic	2: Managed	3: Predictive	4: Adaptive	5: Autonomic
Description	Rely on reports, product and manual actions to manage IT components	Management software in place to provide facilitation and automation of IT tasks	Individual components and systems management tools able to analyze and recommend actions	IT components collectively able to monitor, analyze and take action with minimal human intervention	IT components collectively and automatically managed by business rules and policies
Benefits		Greater system awareness Improved productivity	Reduced dependency on deep skills Faster/better decision making	Balanced human/system interaction IT agility and resiliency	Business policy drives IT management Business agility and resiliency

Performance Tools 22
© 2005 IBM Corporation

The Performance Advisors are a key piece of the WebSphere Application Server’s goal to become self-optimizing. Currently WebSphere Application Server is at stage three of this goal. Currently, WebSphere Application Server is capable of analyzing its environment and providing recommendations to the administrator. Based on the advice received, the administrator can efficiently analyze and configure his or her environment to increase its performance. The end point that WebSphere Application Server wants to reach is a business application environment that is capable of tuning itself based on defined business rules and policies.

Usage

- Should only be used during stable production load tests
 - ▶ Application needs to remain stable during production tests
 - Any exceptions and deadlock issues need to be resolved prior to running
- Should be enabled once production load tests have reached peak load levels
 - ▶ Exclude ramp up and ramp down times from monitoring
 - Advice may become contradictory
 - ▶ Certain types of advice **only** generated when processor is being stressed
- **Important:** Tune your application before you tune your Application Server



How and when should the Performance Advisors be used? It is recommended that the advisors be used in production load tests against an application that has had all of its major defects and deadlocks eliminated. It is important to keep in mind that unexpected application errors may cause generated advice to be skewed or incorrect.

It should also be noted that the Performance Advisors should only be enabled when your production load tests are at their peak load levels. Enabling the advisors during the ramp up or ramp down times of a production load test could cause generated advice to be skewed or incorrect. For example, advice generated when the system is at 40% capacity versus a system that is at 90% capacity will be different. Additionally certain types of advice require that the processor is stressed before advice is generated.

The important point to take away from this, is that you should tune your application before you tune your Application Server.

Section

Summary and Reference



The next section will summarize the points discussed in this lecture.

Summary

- WebSphere Application Server provides many tools to assist in tracking down performance problems
- Performance Monitoring Infrastructure (PMI) externalizes performance data
- Tivoli Performance Viewer (TPV) allows you to visualize and analyze collected PMI data
- Performance Advisors provide configuration advice based on collected PMI data



In summary, this presentation has focused on the different performance tools provided by WebSphere Application Server. The Performance Monitoring Infrastructure service is used to externalize the Application Server's performance data. The Tivoli Performance Viewer allows the administrator to visualize and analyze the collected PMI data. The Performance Advisors provide configuration advice based on the collected PMI data and the Application Server's current configuration.

Trademarks, Copyrights, and Disclaimers

The following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both:

IBM	CICS	IMS	MQSeries	Tivoli
IBM (logo)	Cloudscape	Informix	OS/390	WebSphere
e!(logo)/business	DB2	Series	OS/400	xSeries
AIx	DB2 UniversalDatabase	Lotus	pSeries	zSeries

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are registered trademarks of Microsoft Corporation in the United States, other countries, or both.

Intel, ActionMedia, LANDesk, MMX, Pentium and ProShare are trademarks of Intel Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Linux is a registered trademark of Linus Torvalds.

Other company, product and service names may be trademarks or service marks of others.

Product data has been reviewed for accuracy as of the date of initial publication. Product data is subject to change without notice. This document could include technical inaccuracies or typographical errors. IBM may make improvements and/or changes in the product(s) and/or program(s) described herein at any time without notice. Any statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business. Any reference to an IBM Program Product in this document is not intended to state or imply that only that program product may be used. Any functionally equivalent program, that does not infringe IBM's intellectual property rights, may be used instead.

Information is provided "AS IS" without warranty of any kind. THE INFORMATION PROVIDED IN THIS DOCUMENT IS DISTRIBUTED "AS IS" WITHOUT ANY WARRANTY, EITHER EXPRESS OR IMPLIED. IBM EXPRESSLY DISCLAIMS ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NONINFRINGEMENT. IBM shall have no responsibility to update this information. IBM products are warranted, if at all, according to the terms and conditions of the agreements (e.g., IBM Customer Agreement, Statement of Limited Warranty, International Program License Agreement, etc.) under which they are provided. Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products in connection with this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. IBM makes no representations or warranties, express or implied, regarding non-IBM products and services.

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents or copyrights. Inquiries regarding patent or copyright licenses should be made, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. The actual throughput or performance that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput or performance improvements equivalent to the ratios stated here.

© Copyright International Business Machines Corporation 2004. All rights reserved.

Note to U.S. Government Users - Documentation related to restricted rights-Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract and IBM Corp.

