WebSphere Application Server V8.5.5 Performance Tuning

WA815 (Classroom)
ZA815 (Self-paced)

Highlights

- Intermediate training for administrators, business partners, independent software vendors, and consultants who work with WebSphere Application Server

- Covers WebSphere Application Server Network Deployment version 8.5.5

Course description

This course teaches you how to monitor and tune WebSphere Application Servers for improved performance. You learn about performance tuning methods and monitoring tools that apply to both the application server environment and Java EE applications. These methods and tools for load testing, monitoring, and tuning are applied to various WebSphere components, such as the application server's Java virtual machine (JVM), JDBC connection pools, JMS resources, and the general runtime environment.

Instructor-led lectures cover topics such as performance tuning concepts and tasks, monitoring server performance, and tools and techniques for load testing. The course provides suggestions for tuning the JVM, JDBC connection pools, and application profiling.

Hands-on lab exercises throughout the course give you practical experience by using tools to generate user loads on an application and monitoring key performance metrics such as response time and throughput. You also learn how to gather information from WebSphere built-in performance advisors and how to use this information to tune, test, and monitor Java EE applications.

The lab environment for this course uses the RedHat Linux platform.

For information about other related courses, visit the IBM Training website:

ibm.com/training

General information

Delivery method

Classroom or self-paced virtual classroom (SPVC)

Course level

ERC 2.2

Product and version

WebSphere Application Server Network Deployment version 8.5.5

Audience

This course is designed for anyone who works on WebSphere related applications and projects, including administrators, IBM Business Partners, independent software vendors, and consultants. The course is applicable to users running WebSphere Application Server on various platforms including IBM i systems.

Learning objectives

After completing this course, you should be able to:
- Apply performance tuning methods to WebSphere Application Servers
- Perform load testing by using Apache JMeter
- Monitor application server performance by using WebSphere and the IBM Support Assistant
- Monitor and tune the JVM for optimum throughput and response time
- Monitor and tune connection pools for optimum performance
- Implement best practices for general WebSphere runtime performance
- Use the IBM Health Center tool to profile and tune Java EE applications
- Load test and monitor an application server cluster

Prerequisites

Before taking this course, you should have:
- WebSphere administration skills, which you can learn by successfully completing an IBM WebSphere Application Server V8.5 or V8.5.5 administration course (WA585/ZA585 or WA855/ZA855), or through practical experience with administering a WebSphere Application Server environment
- Basic operating skills for the Linux operating system

Duration

4 days

Skill level

Intermediate

Classroom (ILT) setup requirements

Processor	Intel Core 2 Duo P8400 (2.26 GHz, two cores) or faster
GB RAM	4.0 GB
GB free disk space	120 GB
Network requirements	LAN / Internet
Other requirements	Three workstations are required for each student in the class. These machines must be able to communicate with each other over the LAN and have Internet access.

Notes

The following unit and exercise durations are estimates, and might not reflect every class experience. If the course is customized or abbreviated, the duration of unchanged units will probably increase.

This course is an update of the following previous courses:
- WA715 or VA715, *IBM WebSphere Application Server V7 Performance Tuning*

Course agenda

Course introduction
Duration: 30 minutes

Unit 1. WebSphere Application Server systems and components
Duration: 1 hour

Overview	This unit provides an overview of the topology of a system. It identifies and describes key components and troubleshooting points.
Learning objectives	After completing this unit, you should be able to: • Describe stand-alone server architecture • Describe Network Deployment (ND) cell architecture • List and describe the function of IBM products that are involved in implementing stand-alone and distributed architectures • Identify the components of the application server and describe the services that they provide • Identify the components of an ND cell and describe the function of each • Describe features of the Liberty profile • Describe functions of Intelligent Management • Describe the flow of an application request • Describe the flow of administration requests • Identify common bottlenecks in the end-to-end flow of client requests

Unit 2. Overview of performance concepts and tuning tasks
Duration: 2 hours

Overview	This unit describes the need for performance monitoring and tuning. It defines and gives examples of common performance concepts and explains several views of performance such as user, system, and application.
Learning objectives	After completing this unit, you should be able to: • Describe performance planning and design tasks that are performed during development • Explain the value of application profiling, static code analysis, and runtime analysis that is performed during development • Explain the goals and value of automated performance testing • Describe the goal of performance monitoring

Exercise 1. POD configuration
Duration: 30 minutes

Overview	This exercise covers the setup of your exercise POD. The term "POD" describes the group of three machines that work together for the lab exercises. You use a set of scripts that configure the three machines to work together.
Learning objectives	After completing this exercise, you should be able to: • Use the POD of three machines for the lab exercises

Unit 3. Apache JMeter and load testing
Duration: 1 hour

Overview	This unit describes how to use Apache JMeter as a load testing tool.
Learning objectives	After completing this unit, you should be able to: • Describe the features and functions of Apache JMeter • Create, run, and analyze load tests by using Apache JMeter • Run a ramp-up test • Describe the features of the DayTrader benchmark application

Exercise 2. Apache JMeter basics
Duration: 1 hour

Overview	This exercise provides a basic introduction to using Apache JMeter to create test plans that simulate multiple users who access a web-based application. You learn how to use JMeter listeners to monitor performance metrics such as response time and throughput, and how to use the JMeter Test Script Recorder to generate test scripts by navigating a website.
Learning objectives	After completing this exercise, you should be able to: • Start Apache JMeter • Create a simple test plan • Run the test plan and monitor application performance • Use JMeter to record a test script by navigating a remote website

Exercise 3. DayTrader Benchmark installation
Duration: 1 hour

Overview	In this exercise, you install the DayTrader Benchmark example application.
Learning objectives	After completing this exercise, you should be able to: • Set up the WebSphere Application Server ND environment • Install and configure the DayTrader application • Test and explore the DayTrader application

Exercise 4. Using Apache JMeter to load test DayTrader
Duration: 1 hour

Overview	In this exercise, you learn how to use Apache JMeter to run a ramp-up test on the DayTrader application. At the end of the exercise, you have a test plan that you can use as a baseline for later performance tuning exercises.
Learning objectives	After completing this exercise, you should be able to: • Tune and back up the DayTrader database • Run a ramp-up test on DayTrader to find the saturation point • Monitor CPU usage on the application server host and the database server host • Establish a baseline of performance for the DayTrader application server

Unit 4. WebSphere performance data and tools
Duration: 1 hour and 30 minutes

Overview	This unit describes performance data and performance data analysis tools.
Learning objectives	After completing this unit, you should be able to: • Describe important performance data for monitoring WebSphere based applications • Identify useful performance tools for monitoring performance data that describes the behavior of WebSphere based applications

Exercise 5. Performance monitoring tools
Duration: 1 hour

Overview	This exercise familiarizes you with the performance monitoring tools and performance advisor facilities that are available in WebSphere Application Server and IBM Support Assistant.
Learning objectives	After completing this exercise, you should be able to: • Monitor performance by using Tivoli Performance Viewer • Use the performance advisors, including runtime Performance Advisor and Performance Advisor in Performance Viewer • Use the Health Center tool in the IBM Support Assistant to monitor a running JVM

Unit 5. WebSphere performance tuning methods
Duration: 1 hour

Overview	This unit describes various techniques and methods for monitoring and tuning performance.
Learning objectives	After completing this unit, you should be able to: • Describe performance tuning goals, strategies, and considerations • Evaluate a list of the top 10 monitoring considerations • Gather performance data for WebSphere Application Server components • Detect performance bottlenecks • Determine optimum queue sizes • Configure WebSphere Application Server system queues • Identify and evaluate WebSphere tuning parameters

Unit 6. Introduction to the JVM
Duration: 1 hour

Overview	This unit provides an overview of the JVM and garbage collection policies.
Learning objectives	After completing this unit, you should be able to: • Describe components of the JVM and overall architecture • Describe JIT and AOT compilers • Describe the garbage collection process • Describe garbage collection policies such as *generational concurrent* and *optimize for throughput* • Describe JVM heap memory usage

Exercise 6. Exploring GC policies and monitoring JVM performance
Duration: 1 hour

Overview	In this exercise, you explore garbage collection policies and monitor their performance during load testing.
Learning objectives	After completing this exercise, you should be able to: • Enable verbose GC for an application server • Manually view verbose GC logs • Configure the optthruput GC policy • Configure the optavgpause GC policy • Use Apache JMeter to load test an application server • Use IBM Garbage Collection and Memory Visualizer (GCMV) to analyze and compare GC data

Unit 7. Tuning the JVM
Duration: 1 hour and 30 minutes

Overview	This unit describes how to tune the JVM and how to troubleshoot common JVM problems.
Learning objectives	After completing this unit, you should be able to: • Size the JVM heap • Tune various GC policies • Troubleshoot common JVM problems

Exercise 7. Tuning the JVM
Duration: 2 hours

Overview	This exercise covers tuning the application server JVM. You configure and load test various heap sizes for TradeServer1. You use monitoring and analysis tools such as Health Center and GCMV to gather tuning recommendations for the JVM.
Learning objectives	After completing this exercise, you should be able to: • Configure the initial and maximum JVM heap size • Use Apache JMeter to load test various heap sizes • Use Heath Center to monitor garbage collection and heap usage • Use GCMV to gather tuning recommendations for the JVM

Exercise 8. Troubleshooting JVM problems
Duration: 1 hour

Overview	In this exercise, you learn how to detect and troubleshoot common JVM problems. Problems that cause the server to hang or crash can cause poor performance.
Learning objectives	After completing this exercise, you should be able to: • Use the IBM Support Assistant Health Center tool to monitor Java heap size and usage • Detect and troubleshoot a server that is experiencing out-of-memory exceptions • Use the IBM Support Assistant Memory Analyzer tool to analyze heap dumps • Detect and troubleshoot a server with hung threads • Use the IBM Support Assistant Thread Monitor Dump Analyzer tool to analyze thread dumps

Unit 8. Tuning the connection pool
Duration: 45 minutes

Overview	This unit describes connection pooling and how to tune it to maximize the chances that an application is able to get a connection to a data source when it needs one.
Learning objectives	After completing this unit, you should be able to: • Configure connection pool parameters in the administrative console • Use Tivoli Performance Viewer to monitor the connection pool • Use the Tivoli Performance Viewer Performance Advisor to generate tuning advice • Test, monitor, and tune a connection pool • Monitor and tune the prepared statement cache

Exercise 9. Tuning JDBC connection pools and enabling servlet caching
Duration: 1 hour

Overview	In this exercise, you examine the connection pool configuration for the Trade data sources. You use the Tivoli Performance Viewer and Apache JMeter to monitor, tune, and load test connection pools.
Learning objectives	After completing this exercise, you should be able to: • Examine and configure connection pool performance parameters by using the administrative console • Configure the Tivoli Performance Viewer to monitor connection pool performance statistics • Use Apache JMeter to load test an application • Monitor an application by using Tivoli Performance Viewer and Apache JMeter • Tune the connection pool performance parameters after analyzing results from the Tivoli Performance Viewer • Enable servlet caching for an application server • Use the Dynamic Cache Monitor to view caching activity and cache contents • Monitor the performance of the DayTrader application when servlet caching is enabled

Unit 9. WebSphere runtime performance tuning
Duration: 1 hour and 30 minutes

Overview	This unit presents topics about tuning that cover the end-to-end WebSphere Application Server runtime environment.
Learning objectives	After completing this unit, you should be able to: • Identify and tune parameters that are associated with database connectivity, EJBs, and dynamic caching • Identify and tune parameters that are associated with WebSphere messaging • Implement performance best practices for WebSphere clusters • Explain performance considerations associated with using 64-bit WebSphere

Unit 10. Application profiling and tuning
Duration: 1 hour

Overview	This unit describes tools that can be used to help tune application performance.
Learning objectives	After completing this unit, you should be able to: • Describe areas of performance bottlenecks for applications • Use the IBM Java Health Center to investigate memory issues in applications • Use IBM Java Health Center to profile application execution • Use IBM Java Health Center to examine application thread information • Use IBM Java Health Center to examine application lock information

Exercise 10. Application profiling with Java Health Center
Duration: 1 hour

Overview	The Java Health Center has many features for examining what is happening in a running JVM. This exercise covers the use of the Java Health Center for analyzing application performance. You use the Java Health Center to examine JVM characteristics that are related to application execution time, memory usage, and concurrency.
Learning objectives	After completing this exercise, you should be able to: • Use the Java Health Center to find application methods with the longest execution time • Use the Java Health Center to examine garbage collection behavior • Use the Java Health Center to find the location of large object allocations • Use the Java Health Center to analyze calls to System.gc() • Use Java tracing to find the location of calls to Sytem.gc() • Use the Java Health Center to examine locking behavior

Unit 11. WebSphere clusters and scalability
Duration: 1 hour

Overview	This unit describes WebSphere Application Server clusters and scalability and Intelligent Management.
Learning objectives	After completing this unit, you should be able to: • Describe the static clusters and dynamic clusters • Compare the benefits of vertical and horizontal scaling • Explain the use of multiple processors for scalability • Describe queuing considerations for vertical and horizontal scaling • Describe the main features of Intelligent Management

Exercise 11. Load testing an application server cluster
Duration: 1 hour and 30 minutes

Overview	In this exercise, you create a horizontal cluster for the DayTrader application. You then use the IBM HTTP plug-in to route requests to DayTrader. Finally, you run some load tests to measure the scalability of a horizontal cluster.
Learning objectives	After completing this exercise, you should be able to: • Install the DayTrader Application to a cluster • Configure the IBM HTTP plug-in to route requests to the cluster • Use Apache JMeter to load test the cluster • Use the Apache server-status tool to monitor web server usage • Monitor CPU usage on cluster member hosts • Reconfigure cluster member weights to make better use of system resources

Unit 12. Course summary
Duration: 20 minutes

Overview	This unit summarizes the course and provides information for future study.
Learning objectives	After completing this unit, you should be able to: • Explain how the course met its learning objectives • Access the IBM Training website • Identify other IBM Training courses that are related to this topic • Locate appropriate resources for further study

For more information

To learn more about this course and other related offerings, and to schedule training, visit **ibm.com/**training.

To learn more about validating your technical skills with IBM certification, visit **ibm.com**/certify.

To stay informed about IBM training, visit the following sites:

IBM Training News: http://bit.ly/IBMTrainEN

YouTube: youtube.com/IBMTraining

Facebook: facebook.com/ibmtraining

Twitter: twitter.com/websphere_edu