

# IBM - Basic z/OS Tuning Using the Workload Manager

---

<b>Code:</b>	ES54G
<b>Length:</b>	4.5 days
<b>URL:</b>	<a href="#">View Online</a>

---

Do you need to know how to establish a practical performance management program for your z/OS system- This course is designed for new performance analysts to learn to work with the Workload Manager (WLM) in goal mode. Learn concepts of WLM and performance management in the z/OS system using the WLM.

Learn how to analyze Resource Monitoring Facility (RMF) reports and implement service definitions using the WLM Interactive System Productivity Facility (ISPF) application. The course uses z/OS hands-on lab exercises to reinforce the concepts and techniques discussed in lecture.

## Hands-on labs

This course includes hands-on lab exercises. Each student team, working on their own z/OS system, will configure RMF, analyze RMF reports to find bottlenecks, and utilize the WLM dialogs to create goals and classification rules in a service definition that manages a supplied z/OS workload.

## Skills Gained

- Describe a performance and tuning methodology
- Develop a systematic z/OS performance and tuning plan
- Describe the factors which could affect the performance of an z/OS system
- Use the WLM ISPF application
- Describe the components of a service definition
- Define workloads and service levels and classification rules
- State which z/OS commands affect WLM operation
- Identify the major WLM services for z/OS, including enclaves and application environments, and how they are used by DB2, WebSphere, and CICS
- Analyze CPU performance when running in a shared LPAR environment
- Utilize and monitor zIIP and zAAP specialty engines
- Measure and tune z/OS DASD, processor storage, and coupling facility configurations
- Explain the functions and facilities of RMF and SMF
- Analyze performance bottlenecks using RMF
- Use Workload License Charges (WLC), defined capacity and soft capping to manage software costs
- Describe advanced z/OS environments that utilize Intelligent Resource Director (IRD), HiperDispatch, z/OSMF

Workload Management, and I/O Priority Manager

- Use the z/OSMF Workload Management (WLM) task
  - Use Performance Monitoring with z/OSMF
  - Modify a WLM service definition to meet the requirements for monitoring a specific system workload
  - Create and customize Monitoring Desktops
  - Review any issues by using the Monitoring Desktops options displays
  - Assess the performance of the workloads running on the z/OS

## Who Can Benefit

This is an intermediate course for z/OS system programmers, z/OS performance analysts, and z/OS performance administrators new to performance management for their z/OS system.

**Note:** ES54 is intended for individuals new to WLM and the z/OS performance area

## Prerequisites

You should:

- Understand basic MVS **and** z/OS operation, such as job flow through JES, job scheduling paging, swapping, dispatching controls, and I/O scheduling
- Have a basic knowledge of the purpose of the Workload Manager's function in managing system workloads
- Be familiar with using TSO **and** ISPF to manage data sets **and** run batch jobs

## Course Details

### Day 1

- Welcome
- Unit 1 - Tuning methodology
- Unit 2 - Using SMF and RMF to monitor performance
- Lab 1 - Introduction to your system
- Lab 2 - Using RMF Monitor I and Monitor II

### Day 2

- Unit 3 - Performance impacts when running in a shared LPAR environment
- Unit 4 - Basic system workload management (part 1)
- Lab 3 - Implementing a WLM environment on z/OS (part 1)

### Day 3

- Unit 4 - Basic system workload management (part 2)
- Lab 3 - Implementing a WLM environment on z/OS (part 2)

## Day 4

- Unit 5 - WLM commands, internals, and service
- Lab 4 - Using RMF Monitor III to solve performance problems

## Day 5

- Unit 6 - z/OS DASD performance topics
- Unit 7 - Tuning processor storage
- Unit 8 - Miscellaneous performance topics
- Lab 5 - z/OSMF and performance management

---

Refer a friend or colleague and get up to \$100 Amazon gift card\* — when they book training!

[Learn More](#)