



IBM Systems
Virtualization Engine
Enterprise Workload Manager

Version 2 Release 1





IBM Systems

Virtualization Engine
Enterprise Workload Manager

Version 2 Release 1

Note

Before using this information and the product it supports, read the information in “Notices,” on page 341.

Seventh Edition (June 2006)

This edition applies to version 2, release 1 of IBM Virtualization Engine and to all subsequent releases until otherwise indicated in new editions. This version does not run on all reduced instruction set computer (RISC) models nor does it run on CISC models.

© Copyright International Business Machines Corporation 2004, 2005, 2006. All rights reserved.

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

Enterprise Workload Manager	1
What's new	2
Printable PDF	6
EWLM roadmap	7
Getting started	9
EWLM overview	9
Management environment	10
EWLM management domain	11
EWLM domain manager	12
EWLM managed server	12
EWLM Control Center	13
Firewall broker	15
Partition workload group	16
Monitoring work	17
Application instrumentation	20
Data collection and aggregation	23
EWLM load balancing	23
Load balancing communications	25
Partition management	26
Coexisting workload management products	27
EWLM concepts	31
Classification	31
Classification rule	33
Filter	35
EWLM-supplied filter types	35
Filter values for the predefined platforms	37
Platform-specific filter types	38
Application-specific filter types	40
Correlator	42
Domain policy	43
Example: Structure of a domain policy	44
Service policies	44
Workloads	47
Applications	47
Transaction classes	48
Platforms	48
Process classes	49
Partition classes	50
Real-time reporting	51
Performance objectives	52
Scenarios	53
Scenario: Classifying and monitoring	
transactions processed by the WebSphere	
Application Server	53
Create a new domain policy, workload and	
service classes	54
Create a domain policy, applications and	
transaction classes	55
Deploy the domain policy and activate the	
service policy	55
Determine if performance goals are met	56
Scenario: Classifying and monitoring	
operating system processes	56
Create a new domain policy, workload and	
service classes	57

Create a domain policy, platforms and	
process classes	58
Deploy the domain policy and activate the	
service policy	59
Determine if performance goals are met	59
Scenario: Classifying and monitoring i5/OS	
operating system processes separately from	
WebSphere transactions	60
Create a new domain policy, workload and	
service classes	61
Create a domain policy, applications and	
transaction classes	62
Create a domain policy, platforms and	
process classes	63
Deploy the domain policy and activate the	
service policy	64
Determine if performance goals are met	64
Scenario: Using the sample banking domain	
policy	64
Task 1: Define performance objectives in	
service classes	66
Task 2: Define work to service classes using	
transaction and process classes	68
Task 3: Review and verify the service	
policy and deploy the domain policy	72
Specified operating environments for EWLM	74
Planning for EWLM	79
Installing EWLM domain manager	89
Command usage considerations	91
Installing EWLM on z/OS	94
Installing the EWLM managed servers	95
Using IBM Director for managed server	
installation	95
Using Remote Package Deployer for managed	
server installation	96
Manually distributing and installing the	
managed server	97
Manually install managed server using	
interactive mode	98
Manually install managed server using	
silent mode	98
Manually install managed server using	
console mode	99
i5/OS: Installing i5/OS managed server	
on local system	100
Windows: Managed server installation	
considerations	100
AIX: Loading the netWLM fileset	101
Prerequisite	101
Install the netWLM fileset	101
HP-UX: Installing LSOF utility for load	
balancing support	102
Solaris: Installing LSOF utility for load	
balancing support	102
Fix package installation considerations for	
EWLM	102

Configuring EWLM	105	Run the EWLM managed server configuration wizard	143
Configuration wizard usage considerations	106	Run createMS	143
Configuring EWLM on AIX	108	Creating a firewall broker	145
Configuring a domain manager	108	Run the EWLM managed server configuration wizard	145
Run the EWLM domain manager configuration wizard	108	Run createFB	146
Run createDM	109	Authorizing Linux users to ARM services	148
Configuring a managed server	111	Enabling ARM services on Linux	148
Run the EWLM managed server configuration wizard	112	Stopping ARM services on Linux	148
Run createMS	112	Installing required RPMs to allow EWLM to manage a Linux partition	149
Creating a firewall broker	114	Assigning users to EWLM Control Center roles	149
Run the EWLM managed server configuration wizard	114	Configuring EWLM on Solaris	151
Run createFB	115	Configuring a managed server	151
Assigning users to EWLM Control Center roles	117	Run the EWLM managed server configuration wizard	151
Authorizing AIX non-root users to start EWLM on an AIX managed server	118	Run createMS	151
Authorizing AIX users to ARM services	120	Creating a firewall broker	153
Enabling ARM services on AIX	120	Run the EWLM managed server configuration wizard	153
Configuring EWLM on HP-UX	121	Run createFB	154
Configuring a managed server	121	Authorizing Solaris users to ARM services	156
Run the EWLM managed server configuration wizard	121	Enabling ARM services on Solaris	156
Run createMS	121	Verifying the status of ARM services on Solaris	156
Creating a firewall broker	123	Disabling ARM services on Solaris	157
Run the EWLM managed server configuration wizard	123	Configuring EWLM on Windows	157
Run createFB	124	Configuring a domain manager	157
Authorizing HP-UX users to ARM services	126	Run the EWLM domain manager configuration wizard	157
Enabling ARM services on HP-UX	126	Run createDM	158
Disabling ARM services on HP-UX	126	Configuring a managed server	160
Verifying the status of ARM services on HP-UX	127	Run the EWLM managed server configuration wizard	160
Configuring EWLM on i5/OS	127	Run createMS	160
Add EWLM library to your system library list	127	Creating a firewall broker	162
Configuring a domain manager	128	Run the EWLM managed server configuration wizard	162
Run the EWLM domain manager configuration wizard	129	Run createFB	163
Run createDM	129	Assigning users to EWLM Control Center roles	165
Configuring a managed server	132	Authorizing Windows users to ARM services	166
Run the EWLM managed server configuration wizard	132	Enabling ARM services on Windows	167
Run createMS	132	Configuring EWLM on z/OS	167
Creating a firewall broker	134	Updating configuration files on z/OS	167
Run the EWLM managed server configuration wizard	134	Updating the BPXPRMxx member in PARMLIB	168
Run createFB	135	Configuring a domain manager on z/OS	168
Assigning users to EWLM Control Center roles	137	Configuring z/OS for Virtualization Engine WebSphere for z/OS	176
Authorizing i5/OS users to ARM services	138	Configuring a managed server on z/OS	179
Enabling ARM services on i5/OS	139	Run createMS	179
Configuring EWLM on Linux	139	Configuring a firewall broker on z/OS	181
Configuring a domain manager	139	Run createFB	181
Run the EWLM domain manager configuration wizard	139	Assigning users to EWLM Control Center roles	183
Run createDM	140	Authorizing z/OS users to start EWLM	185
Configuring a managed server	142	Authorizing z/OS users to ARM services	188
		Enabling ARM services on z/OS	189

	Disabling ARM services on z/OS	189		Enabling ARM on DB2 Universal Database for use by WebSphere Application Server on Windows	212
	Verifying the status of ARM services on z/OS	189		Enabling ARM on DB2 Universal Database for use by WebSphere Application Server on z/OS	213
	Starting EWLM.	189		Enabling ARM on WebSphere Application Server 5.1.1 or 6.0	215
	Starting EWLM on the systems in your domain	189		Enabling ARM on WebSphere Application Server 5.1.1	215
	Starting EWLM on the domain manager	190		Enabling ARM on WebSphere Application Server 6.0	223
	Starting EWLM on the managed server.	190		Enabling ARM on the EWLM plug-in for IIS 5.0 or 6.0	234
	Starting EWLM on the firewall broker	191		Enabling ARM on EWLM plug-in for IIS 5.0 or 6.0 on Windows	234
	Starting the EWLM Control Center	192		Enabling ARM on HTTP Server	236
	Start EWLM (STRWLM) i5/OS CL command	193		Enabling ARM on the HTTP Server with a WebSphere Application Server HTTP plug-in	236
	Stopping EWLM	194		Enabling ARM on the HTTP Server with an independent HTTP plug-in.	243
	Stopping EWLM on the domain manager	194		Enabling ARM on the HTTP Server for iSeries with an iSeries HTTP plug-in	246
	Stopping EWLM on the managed server	194		Enabling EWLM load balancing	248
	Stopping the firewall broker	195		Enabling partition management	249
	End EWLM (ENDWLM) i5/OS CL command	195		Configuring the partition profile to specify a partition workload group	249
	Stopping EWLM on z/OS	196		Installing and configuring the EWLM managed server on the partition	251
	Stopping EWLM on the domain manager on z/OS	196		Enabling EWLM partition management	252
	Stopping EWLM on the managed server on z/OS	196		Changing a partition's partition workload group	252
	Stopping the EWLM firewall broker on z/OS	196		Customizing EWLM configuration properties	253
	Stopping the EWLM Control Center.	196		Changing domain manager configuration properties	253
	Stop the EWLM Control Center using the stopWAS command	197		Changing managed server configuration properties	257
	Stop the EWLM Control Center Windows service.	197		Changing firewall broker properties	260
	Managing EWLM	198		Changing the WebSphere Application Server administrator	262
	Securing EWLM	198		Changing the EWLM Control Center configuration	263
	Accessing the domain manager through a firewall	198		Changing the EWLM Control Center configuration on z/OS	264
	Example: Accessing the domain manager through an HTTP proxy server	198		Displaying EWLM configuration properties	265
	Example: Accessing the domain manager through a SOCKS server.	199		Displaying domain manager properties.	265
	Specifying a secure SSL certificate for Web browser and EWLM Control Center communications	202		Displaying managed server properties	265
	Setting up secure communications	203		Displaying firewall broker properties	266
	Using server SSL between the domain manager and managed servers	203		Displaying EWLM Control Center properties	266
	Using client/server SSL between the domain manager and managed servers.	204		Managing an EWLM domain with the EWLM Control Center	267
	Using client/server SSL between the domain manager and firewall brokers	204		Managing a domain policy	267
	Enabling ARM on IBM-instrumented applications	205		Creating a domain policy	268
	Enabling ARM on DB2 Universal Database for use by WebSphere Application Server	206		Managing service policies	272
	Enabling ARM on DB2 Universal Database for use by WebSphere Application Server on AIX	206		Managing managed servers	273
	Enabling ARM on DB2 Universal Database for use by WebSphere Application Server on i5/OS	208		Managing applications in EWLM.	274
	Enabling ARM on DB2 Universal Database for use by WebSphere Application Server on Linux	209		Managing an application filter policy	276
	Enabling ARM on DB2 Universal Database for use by WebSphere Application Server on Solaris	210		Managing platforms in EWLM	278
				Managing partitions in EWLM	279
				Enabling EWLM Control Center tracing	280

Removing EWLM Control Center users or groups	281	Managed servers and firewall brokers	321
Monitoring performance data with EWLM	283	Uninstalling EWLM on i5/OS	322
Examining performance data	286	Uninstalling the EWLM managed server from a remote system	322
Monitoring application topology	288	Uninstalling EWLM managed server on local system	323
Monitoring server topology	289	Deleting domain manager configuration data	323
Examining topology data	290	Deleting a Virtualization Engine WebSphere Application Server profile on z/OS	324
Formulas used to calculate topology statistics	294	Deleting managed server configuration data	324
Monitoring partitions in EWLM	295	Deleting firewall broker configuration data	324
Monitoring EWLM environments that use a load balancer	298	Fix package uninstallation considerations for EWLM	325
Performance statistics logging	300	Troubleshooting EWLM	326
Performance statistics logging plug-in descriptor	300	Troubleshooting: Applications	326
Performance statistics logging output formats	304	lsarm	327
Determining where performance problems reside	309	ewlmWinADTool.exe	328
Example: Performance index is 0	311	Verifying the status of ARM services on z/OS	329
Example: Performance index is > 1	312	Troubleshooting: EWLM Control Center	329
Example: Service classes not meeting their goal	317	Troubleshooting: Managed server, domain manager, and firewall broker	333
Example: Allocation of managed server processor utilization	318	Related information for EWLM	336
Example: Classification of work requests to service classes	318	Accessibility features for EWLM	338
Configuring serviceability options	319	Appendix. Notices 341	
Working with service logs	320	Programming Interface Information	342
Uninstalling EWLM	320	Trademarks	343
Domain manager and EWLM Control Center	321	Terms and conditions.	343

Enterprise Workload Manager

Enterprise Workload Manager (EWLM) is a robust performance management tool that allows you to monitor and manage work that runs within your environment. EWLM is robust in that it allows you to define specific performance goals. In addition, EWLM allows you to monitor application-level transactions separate from operating system processes. Furthermore, EWLM allows you to assign performance goals to *specific* work. EWLM allows you to not only view the central processing unit (CPU) usage for systems within your domain but also determine which work contributes the most to the overall system CPU usage.

EWLM provides you with workload management capabilities that offer many benefits to not only large complex environments but also small less complex environments. For more information on environments, see “EWLM management domain” on page 11.

The problem

How can you ensure that work requests perform as expected in your IT environment? One of the most important elements of successful performance management is understanding transaction response times and topologies. So, no matter how complex your business environment, you are likely interested in answering the following:

- Are work requests completing successfully? If not, where are they failing?
- Are application-level transactions completing according to your performance goals?
- Are operating system processes completing according to your performance goals?
- Is the work for an entire partition completing according to your performance goals?
- Are successful work requests completing within the expected response time? If not, where are the bottlenecks?
- How many work requests complete during specific time intervals compared to previous time intervals? Is the workload growing?
- Do the system-level resources ensure optimal performance? If not, can processing power be shifted to alleviate bottlenecks?
- Is the workload balanced to ensure optimal performance? If not, can work be redirected to other systems to alleviate bottlenecks?
- Are Service Level Agreements (SLAs) that define specific performance results being met? If not, what can be done to meet the goals?

To accurately answer these questions, you must have the ability to do the following:

- Identify work requests based on business priority.
- Track the performance of work requests across server and subsystem boundaries.
- Manage the underlying physical and network resources to achieve your specified performance goals.

The solution

IBM® Enterprise Workload Manager (EWLM) allows you to define business-oriented performance goals for an entire domain of servers, and then provides an end-to-end view of actual performance relative to those goals. EWLM is a key component of the IBM Virtualization Engine™. The Virtualization Engine solution simplifies management and utilization of IT resources. As a part of this solution, EWLM does the following:

- Provides an understanding of the actual flow of work, based on dynamic discovery.

- Automatically detects server and application topologies so that it can rapidly and efficiently identify the likely origin of performance problems.
- Allows users to create policies that correspond to Service Level Agreements (SLA).
- Dynamically adjusts processing units among partitions to ensure that performance goals are met.
- Communicates with load balancers to help ensure that performance goals are met.
- Encourages middleware and third-party software vendors to instrument their applications with the Open Group Application Response Measurement 4.0 (ARM) APIs. This allows EWLM to obtain granular performance data for multi-tiered application-level transactions.
- Provides formal, public programming interfaces on which systems management products can build. These products can use EWLM to handle interactions with ARM-instrumented middleware and to provide server and application topology and performance statistics.

More specifically, EWLM allows you to:

- Monitor *operating system processes* separate from *application-level transactions*.
- Compare the actual performance of work to performance goals that you define.
- Identify where performance bottlenecks exist.
- Identify how much different workloads contribute to the overall system CPU utilization.

A typical EWLM management environment consists of a domain manager, a combination of managed servers, and the EWLM Control Center, which is the user interface for managing EWLM. In addition, an EWLM environment may also include firewall brokers, if a firewall exists between the domain manager and managed server. Also, an EWLM environment can include an external load balancer that distributes traffic based on goal-based recommendations from the domain manager. Lastly, an EWLM environment can contain partition workload groups, if EWLM is to adjust procession power among partitions to ensure that performance goals are met.

For more information on which operating systems support the various EWLM functions, see “Specified operating environments for EWLM” on page 74.

What’s new

View this topic to learn more about the enhancements to EWLM for version 2 release 1.

EWLM adds the following new functions and enhancements:

Added function for EWLM logical partition management

EWLM has the ability to adjust the processing power among partitions in a partition workload group. This allows EWLM to automatically adjust the processing power of a partition to ensure that the performance goals are met.

To learn more about how this works, see “Partition management” on page 26.

To enable partition management, see “Enabling partition management” on page 249.

Enhanced monitoring capability to include all work for a partition

EWLM has the ability to monitor an entire partition by using a partition class. A partition class identifies which partition EWLM is to monitor. A partition class is very useful if you want EWLM to monitor a partition that contains applications that do not instrument the Application Response Measurement (ARM) 4.0 standard APIs.

For more information on a partition class, see “Partition classes” on page 50.

Simplified the EWLM configuration

To configure a domain manager, managed server, or firewall broker, you assign each object a configuration ID. Then, if you want to start or stop EWLM on that object or change the object's configuration, you can access the object by using the configuration ID.

Ensure that you log the configuration IDs that you create so that you can refer to them at a later time. Use the planning worksheets to store this data.

Enhanced the EWLM information

EWLM roadmap

This new topic outlines the high-level steps that you must complete to quickly and efficiently install, configure, and use EWLM.

Planning for EWLM

This new topic provides multiple worksheets for you to complete before you install and configure EWLM. You will use the values that you record in these worksheets when you install and configure EWLM.

Scenarios

This new topic collection provides more scenarios that illustrate how you can use EWLM to monitor application-level transactions, operating system processes, or both. In addition, they illustrate how you can identify how much of the processing power, also referred to as the central processing unit (CPU) usage, that these different types of work consume. Therefore, you can identify if the operating system processes, application-level transactions, or specific workload that is consuming the resources.

Enhanced the EWLM Control Center

Increased the usability of the Control Center Web interface when creating classification rules

The EWLM Control Center provides a graphical user interface to create classification rules. This makes it easier to create rules when you define a domain policy. To learn more about how to create classification rules while working in the EWLM Control Center, click Help. Then, select **How do I . . . → Work with . . . → Domain policy → Work with classification rules.**

For more detailed help on how to use the interactive tool to create classification rules, click  on the Add/Edit Rule page of the EWLM Control Center.

Introduced a new hop details report

New report that contains the performance data for application-level transactions. This report allows you to quickly and easily identify which hop may be the cause of a performance problem. A hop is an application or server that processes transactions. As transactions flow from one application or server to another, a hop occurs.

Added transaction data that includes the number of transactions that are in progress

Within the hop details report or topology data report, you can view not only the number of transactions that are stopped, completed or failed, but also the number of transactions that are in progress. The performance data for these transactions is not included in the performance statistics. This data can be particularly useful when determining if transactions are not completing. In addition, you can compare the number of in progress transactions to the performance index to determine the level of validity of the performance index. This data is included in the detail reports for service classes, transaction classes, process classes, or partition classes, if applicable.

Enhanced the performance index calculations to not include failed transactions

Failed transactions can significantly impact the validity of the performance index if too many transactions fail rather quickly. Therefore, the performance index calculation does not include performance data for transactions that end in a failed state.

Enhanced the Control Center to display the past 24 hours worth of performance data


EWLM increased the amount of performance data that is stored in memory to contain up to 24 hours worth of data. EWLM stores this data on the EWLM domain manager file system. This allows you to navigate backward in time to view performance data for the previous 24 hour time period. This data is automatically deleted after 24 hours. If you want to save this data, you must export the data before it is overwritten by the most recent 24 hours worth of data.

Added support to export and save all performance data

The EWLM Control Center reports and monitors were enhanced so that more reports and monitors support the export function. This allows you to save performance data.

Enhancements to the EWLM Control Center online help

Added field level help for new pages of the Control Center

Click  from any page within the EWLM Control Center to view help for a specific field. In addition, use the navigation system in the online help system to view task and concept help.

Updated task help

Updated and enhanced the tasks that are described in the **How do I set up . . .** and **How do I work with . . .** sections of the help. These tasks have been enhanced to provide more step-by-step instructions for various tasks that you can complete.

Updated concepts

Increased the number of concepts that are explained in the **Tell me about . . .** section of the help.

What's new as of 30 June 2006

To take advantage of the new function described in this update to the information center, ensure that you install fix package 2.1.030 or later.

“Performance statistics logging” on page 300

Added a new function that allows EWLM to store the performance statistics that it collects in long term storage. If enabled, EWLM stores this data on the domain manager's local file system.

“Changing a partition's partition workload group” on page 252

Enhanced the EWLM partition management function to dynamically accept changes when a partition moves from one partition workload group to another. EWLM is able to recognize the partition's workload group name change without you having to stop and start EWLM. For EWLM to dynamically accept logical partition workload group name changes, ensure that you adhere to the process outlined in this new topic.

“Partition management” on page 26

This topic is updated to include an illustration that depicts how the EWLM partition management function works.

“Fix package installation considerations for EWLM” on page 102

The fix package installation considerations is updated to provide you with more information on how to install EWLM fix packages. In addition, this topic clarifies what installation methods you can use to install a domain manager fix versus a managed server fix.

“Monitoring EWLM environments that use a load balancer” on page 298

This is a new topic that describes how to view the performance data for a load balancer, if one exists in your environment.

“Monitoring partitions in EWLM” on page 295

This topic is updated to include an example of a partition workload group details report.

“Start EWLM (STRWLM) i5/OS CL command” on page 193

The library name where the STRWLM i5/OS[®] CL command resides is updated.

“Creating a domain policy” on page 268

This is a new topic that you can use as an aid when you define a domain policy. Use the worksheets to help identify the workloads in your domain that you want EWLM to monitor. Then, use this data to create a domain policy that is specific to your environment.

“Application-specific filter types” on page 40

The application-specific ARM filter descriptions for the WebSphere® Application Server are replaced with links to the corresponding documentation in the WebSphere Application Server information center.

“Installing EWLM domain manager” on page 89

Updated this section to streamline the install documentation into one topic collection that applies to all supported operating systems.

“Configuring EWLM” on page 105

Created a new topic collection that contains only the configuration documentation. From this topic, you can select the operation system on which you installed EWLM to retrieve a list of configuration tasks that are specific to that operating system.

“Starting EWLM” on page 189

Created a new topic collection that contains only the tasks that describe how to start EWLM. After EWLM is installed and configured, you must start EWLM on each system that exists in your EWLM domain. From this topic, you can retrieve a list of tasks that describe how to start EWLM, depending on which EWLM components exist in your EWLM domain.

“Specified operating environments for EWLM” on page 74

Operating environment requirements were updated based on the latest versions and release of operating systems that EWLM is supported on.

“Enabling ARM on DB2 Universal Database for use by WebSphere Application Server” on page 206

Topics in this section were updated for clarity and correctness.

“Enabling ARM on WebSphere Application Server 5.1.1 or 6.0” on page 215

Topics in this section were updated for clarity and correctness. When applicable, some sections of information were removed and replaced with links to the WebSphere Application Server Web site.

“Enabling ARM on HTTP Server” on page 236

Removed documentation for all platforms except AIX® and Windows® and instead link to the Apache Software Foundation Web site for information on how to enable ARM.

“Configuring a domain manager on z/OS” on page 168

Updated to include documentation for the updateWAS command.

What’s new as of 3 February 2006

To take advantage of the new function described in this update to the information center, ensure that you install fix package 2.1.020 or later.

EWLM managed server support expanded to include SUSE Linux® Enterprise Server

EWLM expanded its support to include SUSE Linux Enterprise Server (SLES) 9. For more information, see “Specified operating environments for EWLM” on page 74.

“Coexisting workload management products” on page 27

This is a new topic that describes different workload management tools, their capabilities, and special considerations when used in conjunction with EWLM.

“Working with service logs” on page 320

This is a new topic that describes how to locate the system log files if you encounter a problem and want to view the diagnostic information.

“Managing applications in EWLM” on page 274

This is an updated topic that clarifies which applications are predefined in the EWLM Control Center. In addition, this topic is updated to provide more details on how to manage an application and its transactions.

“Managing an application filter policy” on page 276

This is a new topic that describes how to edit or create an application filter policy file. Each application that EWLM is to monitor with a transaction class must have an application filter policy file. You may need to edit an existing application filter policy if the application’s ARM support changes. In addition, you may need to create an application filter policy if the ARM-instrumented application that EWLM is to monitor is not predefined in the EWLM Control Center.

“Examining performance data” on page 286

This is a new topic that describes the performance data fields, **Goal**, **Performance**, **Performance Index (PI)**, that are used throughout the **Monitor** section of the EWLM Control Center.

“Troubleshooting: EWLM Control Center” on page 329

This topic is updated to include additional errors and their recovery steps.

“Managing platforms in EWLM” on page 278

This topic is updated to provide more details on how to manage a platform and its operating system processes with EWLM.

“Managing partitions in EWLM” on page 279

This topic is updated to provide more details on how to manage a partition with EWLM.

“Monitoring partitions in EWLM” on page 295

This topic is updated to provide more techniques on how to monitor the performance of partitions in your domain.

“Configuring a domain manager on z/OS” on page 168



This topic is updated to identify additional user ID authorization requirements that might be necessary when issuing the createDM command.

“Configuring a domain manager” on page 108

This topic now describes the target user ID that EWLM creates when you run this command. You are not to delete this user ID nor edit it. EWLM needs this user ID and password when you access EWLM in the Health Center of the Virtualization Engine Console.

How to see what is new or changed

To help you see where technical changes have been made, this information uses:

- The  image to mark where new or changed information begins.
- The  image to mark where new or changed information ends.

Printable PDF

Use this topic to view and print a PDF of the EWLM topic collection.

| To view or download the PDF version of this document, select Enterprise Workload Manager (about 5900 KB). This PDF includes all of the EWLM information.


| Saving PDF files

| To save a PDF on your workstation for viewing or printing:

- | 1. Right-click the PDF in your browser (right-click the link above).
- | 2. Click the option that saves the PDF locally.
- | 3. Navigate to the directory in which you want to save the PDF.

- | 4. Click Save.

Downloading Adobe Reader

- | You need Adobe Reader installed on your system to view or print these PDFs. You can download a free copy from the Adobe Web site (www.adobe.com/products/acrobat/readstep.html) .

| EWLM roadmap

- | View this topic to obtain a high-level list of the tasks to complete to get the basic function of EWLM started.

- | Use this roadmap if you want to install and configure EWLM in a manner that uses the basic configuration. To learn about the full capabilities of EWLM, complete the process as outlined in the EWLM topic collection, which includes requirements, planning, installing and configuring EWLM, and then managing EWLM, as opposed to this roadmap. Use this roadmap if you only want a high-level view of tasks to complete.



Prerequisites and planning

The first task is planning for EWLM. Planning is very important, and the time you invest in planning will make your implementation as easy as it can be. For more information, complete the following:

1. Ensure that you read and understand the EWLM terminology. For more information, see “EWLM overview” on page 9 and “EWLM concepts” on page 31.
2. Complete the EWLM planning worksheets. For more information, see “Planning for EWLM” on page 79.

Note: If EWLM is the only system service in your Virtualization Engine environment, you only need to complete the planning worksheets that are specific to EWLM.



Installation

After you plan your EWLM implementation strategy, you can begin installing and configuring EWLM. The information provided will help you successfully install and configure EWLM for your specific operating environment. For more information, complete the following:

1. Install the EWLM domain manager. For more information, see “Installing EWLM domain manager” on page 89.
2. Install the EWLM managed server. For more information, see “Installing the EWLM managed servers” on page 95.
Note: There are multiple methods that you can use to install the EWLM managed servers depending on what methods are supported in your EWLM environment. Ensure that you follow the instructions as described in the EWLM topic collection that are specific to the method that you choose to use.
3. If applicable, enable ARM on the applications that EWLM is to monitor. For more information, see “Enabling ARM on IBM-instrumented applications” on page 205.
Note: This step is required only if you want EWLM to obtain more granular performance data for the transactions that the application processes or if you want to identify specific work that an application processes.

Configuration

1. If applicable, configure the EWLM domain manager. For more information, see “Configuring a domain manager” on page 108.
Note: For those platforms that support the EWLM domain manager configuration wizard, you can start it as part of the installation process or you can manually start it after the domain manager install completes.
2. If applicable, configure the EWLM managed server. For more information, see “Configuring a managed server” on page 111. Complete this step for each managed server.
Note: For those platforms that support the EWLM managed server configuration wizard, you can start it as part of the installation process or you can manually start it after the managed server install completes.
3. If applicable, configure the firewall broker. For more information, see “Creating a firewall broker” on page 114.
Note: If communications between the EWLM domain manager and managed servers are secured by a SOCKS server firewall, you must complete this step. The EWLM firewall broker allows communication to securely flow through the firewall. Only one firewall broker is needed for each trusted zone.
4. Assign user IDs that reside in the domain manager’s local registry to EWLM Control Center roles. For more information, see “Assigning users to EWLM Control Center roles” on page 117.

Starting EWLM

1. Start EWLM on each component of your EWLM environment.
 - a. Start EWLM on the domain manager.
 - b. Start EWLM on the managed servers in your domain.
 - c. Start EWLM on the firewall brokers, if any exist.

For more information about how to start EWLM, see “Starting EWLM on the systems in your domain” on page 189.

2. Start the EWLM Control Center. For more information, see “Starting the EWLM Control Center” on page 192.



Working with EWLM

After EWLM is installed and configured, you can use it to monitor the workloads in your EWLM domain. In addition, you can configure and enable advanced functions of EWLM. For more information, see the following steps:

1. Create a domain policy that defines the workloads that EWLM is to monitor and the performance goals that the work is to be measured against. For more information, see “Managing a domain policy” on page 267.
2. Optional advanced function. Enable EWLM partition management. For more information, see “Enabling partition management” on page 249.
3. Optional advanced function. Enable load balancing. For more information, see “Enabling EWLM load balancing” on page 248.
4. Work with the EWLM Control Center. For more information, see “Managing an EWLM domain with the EWLM Control Center” on page 267.
5. Monitor the performance of your workloads. For more information, see “Monitoring performance data with EWLM” on page 283.

Getting started

Use this information to begin learning about EWLM. Furthermore, you can use the high-level EWLM roadmap to complete step-by-step instructions that allow you to implement the basic function of EWLM as quickly as possible.

EWLM overview

IBM Enterprise Workload Manager (EWLM) provides you with a complete workload management environment that manages, monitors, and reports the performance of workloads that run in your environment.

EWLM offers the following capabilities:

Monitor workloads

EWLM has the ability to monitor application-level transactions and operating system processes. You can also define specific performance goals for each workload that runs in your EWLM domain. EWLM collects and aggregates performance data for the work that it monitors.

Manage partition resources

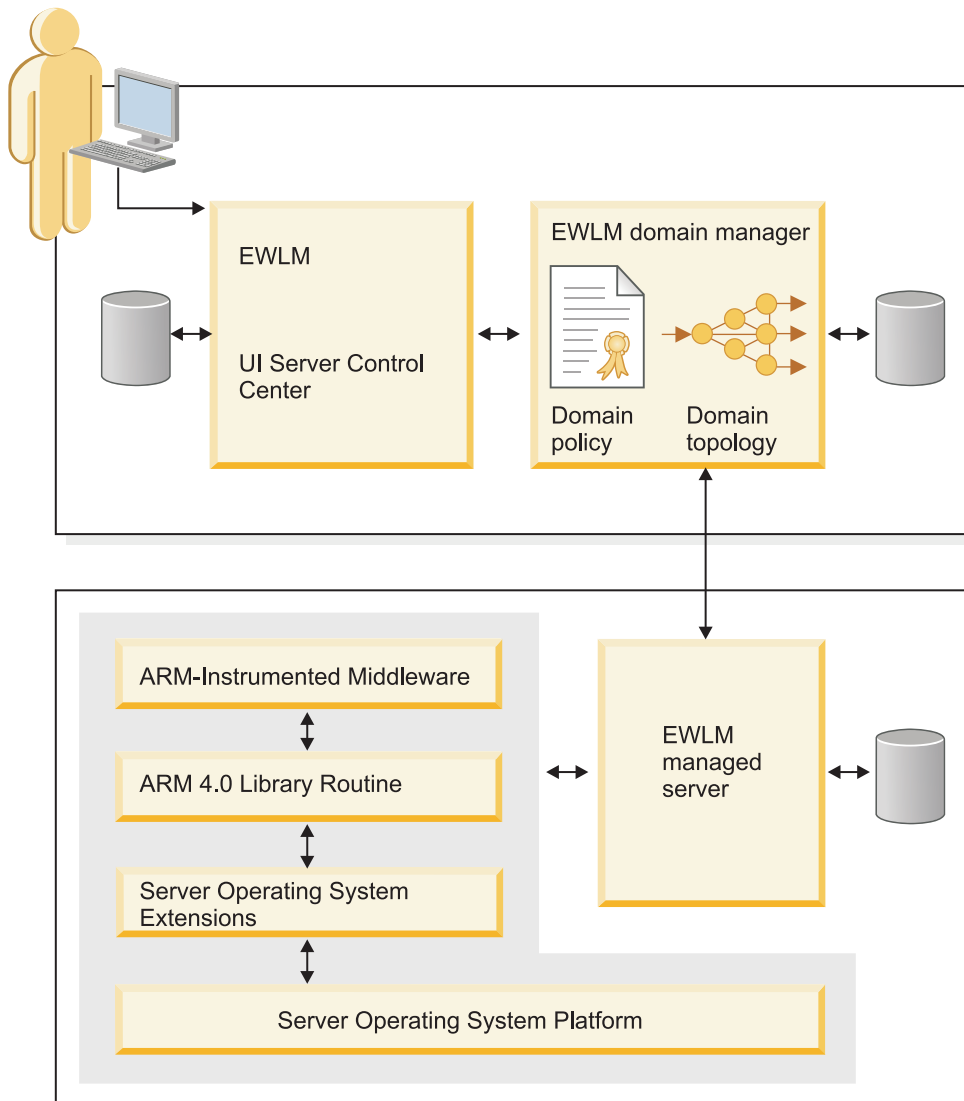
EWLM has the ability to adjust the processing power (Central Processing Units) among partitions in a partition workload group to ensure that performance goals are met.

Distribute workloads

EWLM has the ability to send weight values to load balancers that recommend how to distribute work to endpoint systems to ensure that performance goals are met.

EWLM provides these capabilities across system and subsystem boundaries, regardless of operating system, in an EWLM domain.

The following figure illustrates the relationship between each of the EWLM components; EWLM domain manager, EWLM Control Center, managed server, ARM-instrumented middleware, and operating system platform. View the subtopics for more details on each of these components.



EICAA509-2

View the following flash demo for an animated overview of the core functions that EWLM provides. The following Flash demo requires the Flash plug-in .

Management environment

The *management environment* is a high-level view of your business's network of servers and applications that you use to process work. The complete management environment consists of all applications and servers that are used to complete day-to-day work requests for your business.

With EWLM, you can monitor the applications and servers within your business environment. Depending on the servers and applications in your environment, you can monitor the entire business environment or a select set of servers and applications.

To monitor a server, you must install and configure the EWLM managed server function on the server. In addition, any server that contains an application that you want EWLM to monitor needs to include the managed server function.

To monitor an application, the application needs to use the Application Response Measurement (ARM) 4.0 standard APIs. In addition, you must add the application to the domain policy.

| To learn more about each piece of the EWLM environment, view the following subtopics.

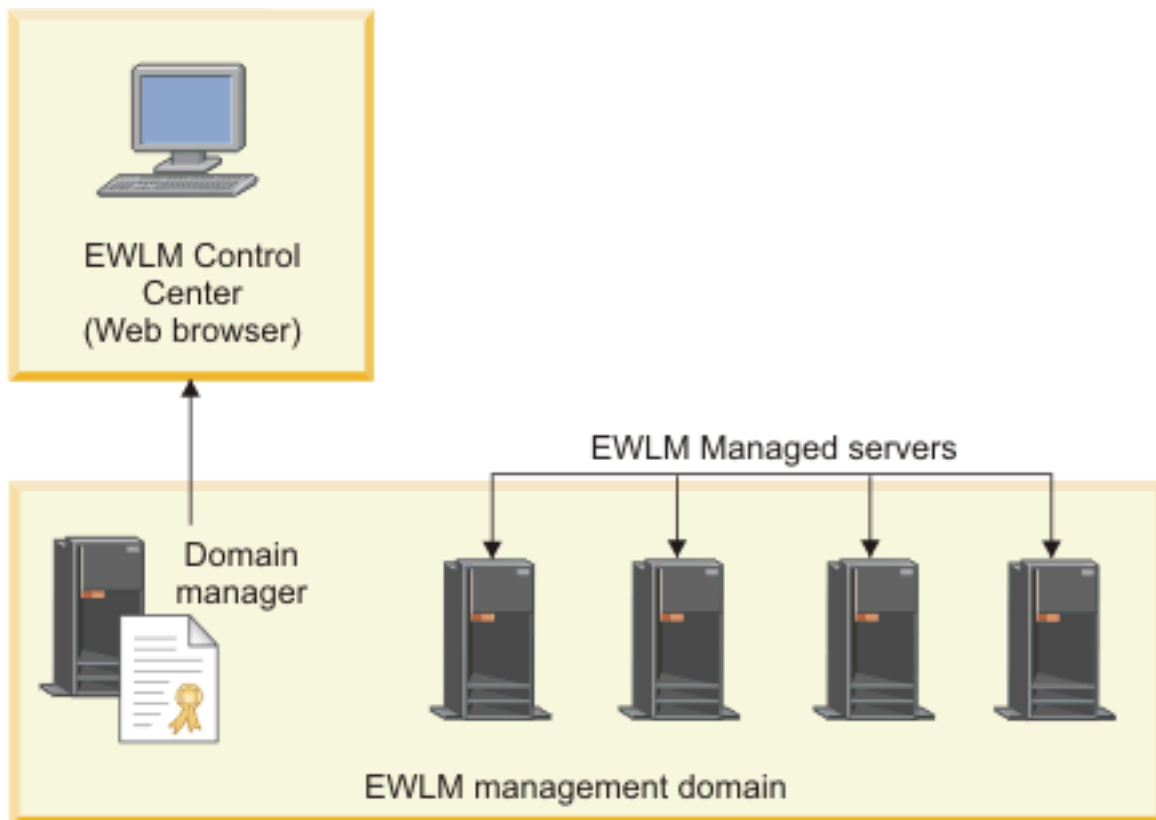
| **EWLM management domain:**

| An *EWLM management domain* consists of a collection of EWLM managed servers and a domain manager.

| The domain manager is the central point of control for a domain because it coordinates the activation of policies on the managed servers and the collection of performance data. A managed server is a server whose work requests are monitored by EWLM. The managed server sends performance data to the domain manager. The domain manager includes the EWLM user interface that is referred to as the EWLM Control Center.

| EWLM makes it easy to manage your servers in a domain because the managed servers can run on varying operating systems. EWLM can obtain performance and transaction data in a heterogeneous environment and report the data to you in a single report.

| The following figure is an example environment that includes the components of an EWLM management domain.



EICAA508-1

| This is one example of an EWLM management domain. Each EWLM management environment is specific to your business needs.

| An EWLM management domain can differ in size and complexity but still provide robust workload management capabilities.

| For example, a large complex environment such as an e-business environment usually consists of some combination of routers, edge servers, Web servers, Web application servers, Enterprise JavaBean (EJB) servers, legacy transaction servers, and database servers; all of which might run on different hardware and operating systems. In a mixed-platform, multi-tiered application environment, the flow of work may

| start at a portal, such as a Web server process, and may then involve one, two, or even more hops through additional server processes spread across two or more tiers of operating system images.

| A small, less complex environment may consist of only one or two database or application servers, which might run on the same type of operating system.

| In either case, EWLM allows you to monitor specific work that runs in your environment according to performance goals that you define. In addition, EWLM allows you to monitor the application-level transactions separately from the operating system processes.

| In either environment, EWLM allows you to:

- | • Define *specific* performance goals that meet your business needs.
- | • Monitor all or a specific set of *application-level transactions*.
- | • Monitor all or a specific set of *operating system processes*.
- | • Monitor not only the system CPU utilization but also how much the work contributes to the overall CPU utilization.
- | • Automatically adjust processing power among partitions to ensure that performance goals are met.
- | • View real-time performance monitor, which allows you to view how much the performance fluctuates over time.
- | • View real-time transaction monitor, which indicates how many transactions are completed over time.

| **EWLM domain manager:**

| The *domain manager* is the central point of control for an EWLM management domain.

| After you install the EWLM domain manager, you must configure it and start EWLM on the domain manager. Then, you can start the EWLM Control Center. From the EWLM Control Center you can manage your EWLM domain. From the EWLM Control Center, the domain manager coordinates the activation of policies on the managed servers that are defined in the EWLM domain. In addition, the domain manager coordinates and manages the performance data sent to it by each managed server in the domain. The domain manager provides this data to you through the EWLM Control Center monitors and reports.

| The domain manager is the system that communicates with load balancers, if they are implemented in your environment. In addition, the domain manager makes resource adjustments to partition workload groups, if they are implemented in your environment.

| **Related concepts**

| “Specified operating environments for EWLM” on page 74

| View this topic to ensure that your system meets the hardware and software requirements for each component of EWLM.

| **Related tasks**

| “Starting EWLM on the systems in your domain” on page 189

| View this topic to learn how to start EWLM on the components of EWLM that you have installed and configured in your EWLM domain. The components of EWLM that you can install and configure include the domain manager, managed servers, and firewall brokers.

| “Configuring a domain manager” on page 108

| View this topic to learn how to configure a domain manager.

| **EWLM managed server:**

| A *managed server* is a server or server instance that EWLM monitors.

| After you install a managed server, you must configure the EWLM managed server. In doing so, you
| specify to what EWLM domain it belongs. After the managed server function is installed, configured, and
| started, the managed server collects and aggregates performance data. Then, it periodically sends this
| performance data to the domain manager.

| The name of the EWLM managed server is the same name as the server name. In the EWLM Control
| Center, the managed server host name, also referred to as managed server name, specifies a unique name
| by which a server is known on a network. The fully qualified host name consists of the host name and
| the TCP domain name. The TCP domain name specifies a group of systems that share a common suffix
| (domain name). For example, in the fully qualified host name *server1.us.ibm.com*, *server1* is the host name
| and *us.ibm.com* is the TCP domain name that contains the *server1* host name. The fully qualified host
| name uses a character-based format such as *server1.us.ibm.com*. However, some servers might use a
| numerical-based name. The numerical-based host name is not the same as the system's IP address.

| **Related concepts**

| "Specified operating environments for EWLM" on page 74

| View this topic to ensure that your system meets the hardware and software requirements for each
| component of EWLM.

| **Related tasks**

| "Starting EWLM on the systems in your domain" on page 189

| View this topic to learn how to start EWLM on the components of EWLM that you have installed and
| configured in your EWLM domain. The components of EWLM that you can install and configure
| include the domain manager, managed servers, and firewall brokers.

| "Configuring a managed server" on page 111

| View this topic to learn how to configure a managed server.

| **EWLM Control Center:**

| The EWLM Control Center is a Web-based user interface that allows you to manage and monitor the
| performance of servers in an EWLM management domain. It is installed with and hosted by the domain
| manager.



The three categories of tasks, which align with EWLM user authority levels, that are supported by the EWLM Control Center are the following:

Administration tasks

Administration of an EWLM management domain involves creating, deploying, and maintaining the EWLM domain policy. The domain policy describes the set of service classes, transaction classes, process classes, and partition classes for EWLM to apply to the EWLM domain.

With this authority level, you can work with the Set up, Manage, and Monitor sections in the Control Center.

Operation tasks

You can display the operational state of EWLM and of each server in a management domain, as well as trigger activities that alter operational states. As an operational administrator, your job is to manage the service policy in effect for the domain. Other operation tasks include:

- Querying the identity of the service policy that is currently in effect for the management domain.
- Initiating activation of a specific service policy, by policy name.

The general operational capabilities of the EWLM Control Center allow you to view the servers currently in the management domain and display the overall state of each server instance. If

necessary, you can stop the EWLM managed server function on the system from the EWLM Control Center. However, to restart EWLM on the managed server, you must manually start EWLM.

With this authority, you can work with the Manage and Monitor sections in the Control Center.

Monitoring tasks

EWLM captures relevant performance statistics on each operating system instance within a management domain and then reports an aggregated view of the real-time state of the domain to the EWLM domain manager. The EWLM Control Center provides a user-friendly way of viewing this information. Through the EWLM Control Center, you can do the following:

- Display simple views of the recent ability of the management domain to satisfy the performance goals declared in the active service policy. The goal-versus-actual view for each service class allows you to rapidly identify potential performance problems in the context of the customer business objectives.
- Display real-time graphs of selected performance metrics, such as goal-versus-actual plots and transaction throughput statistics. Graphical views allow you to actively monitor specific detailed information of interest.
- Graphically display the topology of application and operating system instances supporting a specific service class or transaction class. Topology views allow you to rapidly understand relationships between various components of the environment providing service to a customer business process.

With this authority, you can work with the Monitor section in the Control Center.

As a performance administrator, you can also use the EWLM Control Center to display detailed levels of EWLM performance and topology statistics to identify the source of performance bottlenecks. Through the user interface, you start with an overall service class summary view, which displays the application environment that provides service to a specific service class. You can then look further into the operating system instances and application processes that support the service class. This support allows you to understand where potential problems exist, and to identify the additional support personnel that may need to be directly involved to resolve those problems.

You can import and export policy information with the EWLM Control Center to enable migration from the EWLM Control Center to a user interface provided by a systems management product or to apply a particular domain policy to other domains. Policy information is provided in XML format, which allows you to migrate the policy information to the new systems management interface.

Related tasks

“Assigning users to EWLM Control Center roles” on page 117

View this topic to learn how to assign user IDs that exist on the domain manager’s local user registry to access the EWLM Control Center.

“Starting the EWLM Control Center” on page 192

View this topic to learn how to start the EWLM Control Center.

Firewall broker:

A *firewall broker* provides the ability to allow communication through a firewall.

If a firewall exists between two systems in a network, only authorized communication is allowed through the firewall. In EWLM, the domain manager and managed servers must be able to send and receive communications. If a firewall exists between a domain manager and managed server, you must create a firewall broker. This allows EWLM communication to flow between the systems.

In EWLM, you must configure a firewall broker if a firewall that uses a SOCKS server exists between a domain manager and managed server. The firewall broker allows the domain manager and managed server to communicate through the SOCKS server.

| EWLM firewall broker support is included with the EWLM managed servers. Therefore, you can
| configure a firewall broker from any system on which the managed server support is installed. However,
| you only need to create one firewall broker per trusted zone. When you configure the firewall broker, you
| define which ports to use for EWLM communications between the managed server and firewall as well
| as which ports to use for EWLM communications between the domain manager and firewall.

| **Related concepts**

| “Specified operating environments for EWLM” on page 74

| View this topic to ensure that your system meets the hardware and software requirements for each
| component of EWLM.

| “Securing EWLM” on page 198

| EWLM provides a default level of security. However, you can take additional steps to enhance the
| security level for your business environment.

| **Related tasks**

| “Starting EWLM on the systems in your domain” on page 189

| View this topic to learn how to start EWLM on the components of EWLM that you have installed and
| configured in your EWLM domain. The components of EWLM that you can install and configure
| include the domain manager, managed servers, and firewall brokers.

| “Creating a firewall broker” on page 114

| View this topic to learn how to create a firewall broker. You need to create a firewall broker to allow
| EWLM communication between the domain manager and managed server, if the managed server
| accesses the domain manager by using a SOCKS server.

| **Partition workload group:**

| A *partition workload group* identifies a set of partitions that reside on the same physical system. The
| partition profile specifies the name of the partition workload group that it belongs to, if applicable.

| A partition workload group is defined when you use the Hardware Management Console to configure a
| partition. The **partition workload group** field is intended for applications, such as EWLM, to use. EWLM
| uses this value to identify which partitions it is to manage. Note that the partition workload group is
| only available on systems that support the required hardware. For more information on the environments
| in which partition management is supported, see “Specified operating environments for EWLM” on page
| 74.

| EWLM monitors each partition in a partition workload group. Each partition in a partition workload
| group is a separate EWLM managed server instance. Note that EWLM does not require each partition to
| run the same operating system but the partitions must reside on the same physical system. However,
| EWLM can manage multiple partition workload groups within one EWLM domain.

| If you define a partition workload group in your EWLM domain, EWLM adjusts the processing power of
| the partitions in the partition workload group. EWLM bases its adjustments on calculations that compare
| the actual performance of work processed by the partition workload group to the goals defined in the
| domain policy.

| **Related concepts**

| “Specified operating environments for EWLM” on page 74

| View this topic to ensure that your system meets the hardware and software requirements for each
| component of EWLM.

| “Domain policy” on page 43

| A *domain policy* is a high-level container that specifies performance goals for work processed in the
| EWLM domain.

| “Partition management” on page 26

| *Partition management* refers to the ability of EWLM to automatically adjust the processing power
| among partitions within a partition workload group. The partitions in a partition workload group
| must reside on the same physical machine.

Related tasks

“Enabling partition management” on page 249

Use this topic to learn how to enable logical partition management in EWLM.

Monitoring work

EWLM monitors application-level transactions and operating system processes or both.

The granularity of the performance data depends on the type of work that it monitors. EWLM provides the most granular performance data when it monitors application-level transactions because the applications use the ARM 4.0 standard APIs. These APIs allow EWLM to monitor a transaction as it *hops* from one application to another to continue processing. This allows you to determine at which *hop* a performance problem may exist.

When monitoring any type of work processed in the EWLM domain, EWLM is able to provide you with an end-to-end view of the application and servers that process work. In addition, you can view how the work flows from one application or server to another.

The following list identifies how EWLM categorizes the work that it monitors.

Application-level transactions

This type of work includes the transactions that an application that uses the ARM 4.0 standards processes. EWLM monitors these transactions as they flow from one application to another. Thus, it provides hop-level detail that allows you to pinpoint which hop may be the cause of a performance bottleneck.

Operating system processes

This type of work includes the processes that run directly on the operating system. Processes include work such as service processes, daemons, and long running batch work. EWLM monitors the overall processing time for operating system processes from start to end.

Partitions

This type of work includes both application-level transactions and operating system processes for a partition. When EWLM monitors work for a partition, the applications do not have to use the ARM 4.0 standard APIs. This allows you to monitor the transactions that an application processes even if the application does not use the ARM 4.0 APIs. However, when monitoring work for a partition, EWLM does not collect hop-level transaction data for these transactions because the application might not use the ARM 4.0 APIs. When monitoring a partition, EWLM monitors the partition as a whole. This does not provide details such as the amount of time that transactions and processes incur, but it does provide details such as how much processor time the partition uses and how much processor delay the entire partition experiences.

The following process describes how EWLM monitors work that the EWLM domain processes.

1. The work request starts.

A work request is submitted to an application or server. The attributes of the work identify what type of work it is. When a work request is submitted, it contains attributes that specify information specific to the work request. The attributes indicate if it is an application-level transaction and, if so, from what application it originated or if it is an operating system process. In addition, the work has other attributes such as the host name from which it originated or the job name.

2. The work request enters the EWLM domain.

An application or server within the EWLM domain receives the work request. The work request might originate in the EWLM domain or start from an application or server that is outside of the domain. EWLM does not monitor the work until it enters the applications or servers within the EWLM domain.

3. EWLM classifies the work request.

EWLM classifies the work to a transaction or process class, depending on which type of work it

is. In addition, EWLM classifies the work to a partition class, if the work is processed by a partition that EWLM is monitoring via a partition class. Note that partition and process classes are optional.

If the work is an application-level transaction, EWLM classifies the transaction to a transaction class. This is done when the first ARM instrumented application in the domain processes the transaction. This application is referred to as the *entry application*.

If the work is an operating system process, EWLM classifies the process to a process class defined for that particular operating system, if one exists. Process classes are optional.

In either type of classification, EWLM compares the work to the classification rules. If it matches the classification rules, EWLM classifies the work to that class.

In addition, if the transaction or process is processed by a partition that EWLM is monitoring, EWLM also classifies the work to the partition class.

For more information about classification, see “Classification” on page 31.

4. Application-level transactions only: ARM constructs a correlator for the work request based on the classification information.


If the type of work is an application-level transaction, the ARM instrumentation constructs a correlator with the classification information. This informs secondary applications and other managed servers that process the transaction that EWLM already classified it to a transaction class. Therefore, EWLM does not have to classify the transaction again. EWLM uses the data from the correlator to determine the following:

- The amount of time that each application or server used to process the transaction.
- The name of the application or server that processed the transaction.
- The end-to-end transaction flow as it moves from one application or server to the next.

The movement of the work request from one application or server to another is considered a *hop*. When a hop occurs, the correlator, which is referred to as the *parent correlator*, for the predecessor hop is updated. Then, a new correlator, which is referred to as the *current correlator*, is constructed for the successor hop. This process continues for each hop that occurs.

5. The work request completes.

When the work request completes, the domain manager consolidates all of the performance data that it received from the managed servers. The domain manager includes this performance data in the detail reports, monitors, and topologies. Use the Monitor view in the EWLM Control Center to obtain this detailed performance information.

The following Flash demo explains how EWLM can monitor an application-level transactions as it is processed by servers and applications in an EWLM domain. The following Flash demo requires the Flash plug-in .

(“Transcript”).

Transcript:

Enterprise Workload Manager

IBM Enterprise Workload Manager (EWLM) provides you with a complete workload management environment that manages, monitors, and reports performance statistics based on work requests submitted to applications and servers in an EWLM management domain. Click continue to learn how EWLM monitors a work request.

In this scenario, there are three servers located in the EWLM domain, one of which is the domain manager. The domain manager coordinates and manages the performance data sent to it by each managed server in the domain.

| The EWLM Control Center is a Web-based user interface and is hosted on the server identified as the domain manager.

| To learn how EWLM monitors work, this scenario will examine a transaction that flows through the EWLM domain.

| In this example, a bank teller, John, processes a request for a customer. From a Web application, he submits a request to transfer money from the customer's checking account to savings account.

| The first application in the EWLM to receive the bank teller's request is the Apache application. The Apache application runs on an EWLM managed server and instruments the Application Response Measurement (ARM) 4.0 standard APIs.

| The Apache server is being monitored by EWLM. Because Apache is the first application in the EWLM domain to receive the work request, it classifies it. A work request is classified once and it is classified by the entry application.

| EWLM classifies work to a transaction or process class when it first enters the EWLM domain. In addition, EWLM may also classify the work to a partition class. EWLM monitors this particular work as an application-level transactions because it originated from an application that uses ARM.

| To classify the work, EWLM compares the application-level transaction to the classification rules defined in the transaction classes defined for the Apache application.

| Transaction class A classification rules:
| EWLM: OS Platform = Windows 5.0.0.0 OR
| EWLM: System Name = Bank A OR
| EWLM: user name = BankTeller(*).

| Bank teller John's user name is BankTellerJohn and all other tellers use a similar user name format. Therefore, EWLM classifies all transactions initiated by tellers to this transaction class.

| Transaction class A
| Service class: Web transactions

| EWLM monitors the performance of all transactions assigned to Transaction class A according to the service class defined for Transaction class A. These transactions use the Web transactions performance goal that indicates for work to complete with an average response time of one second.

| To inform other applications and servers in the EWLM domain that EWLM has classified this transaction to Transaction class A, EWLM uses a correlator. The correlator is attached to the transaction. This is referred to as the parent correlator.

| Classification is complete and the work request moves to the IBM WebSphere Application Server to continue processing. This application reads the parent correlator and determines that EWLM has already classified the transaction. The transaction is updated with another correlator (referred to as the current correlator). The current correlator provides performance data that is specific to the current hop, WebSphere Application Server.

| Then, the transaction moves to the IBM DB2® server to continue processing. DB2 reads the parent correlator and adds another correlator to the work request. This correlator contains performance data for the DB2 hop.

| The work request completes. The domain manager consolidates all of the performance data from the correlators associated with the transaction as it flowed from one application to the next, which are viewed as hops. This data is available in the Monitor section of the EWLM Control Center.

“Monitoring performance data with EWLM” on page 283

View this topic collection to learn about the reports and monitors that you can use to monitor the performance of work processed by the EWLM domain in which EWLM monitors. You can use any combination of the reports and monitors to obtain the performance data that suites your needs.

Application instrumentation

Instrumenting an application with the appropriate APIs allows EWLM to properly monitor the application-level transactions.

To ensure that work requests are performing as expected in a multi-tiered heterogeneous server environment, you must be able to identify work requests based on business importance. In addition, you must be able to track the performance of those requests across server and subsystem boundaries, and manage the underlying physical and network resources used to achieve specified performance goals. You can collect this data by using versions of middleware that have been instrumented with the Open Group’s Application Response Measurement (ARM) 4.0 standard. ARM 4.0 provides a set of interfaces that are called by an application and then used by EWLM to calculate the response time and status of work processed by the application. EWLM is responsible for aggregating and reporting the data collected from each managed server in a particular management domain. If the operating environment uses a standard, such as the ARM 4.0 interfaces, the operating system can leverage the same set of base measurements regardless of platform.

C applications that are instrumented to participate in an EWLM environment must be coded to the ARM 4.0 specification level and then bound to a specific runtime ARM library. Instrumented applications dynamically bind to the ARM library routine that is distributed with a particular operating system platform.

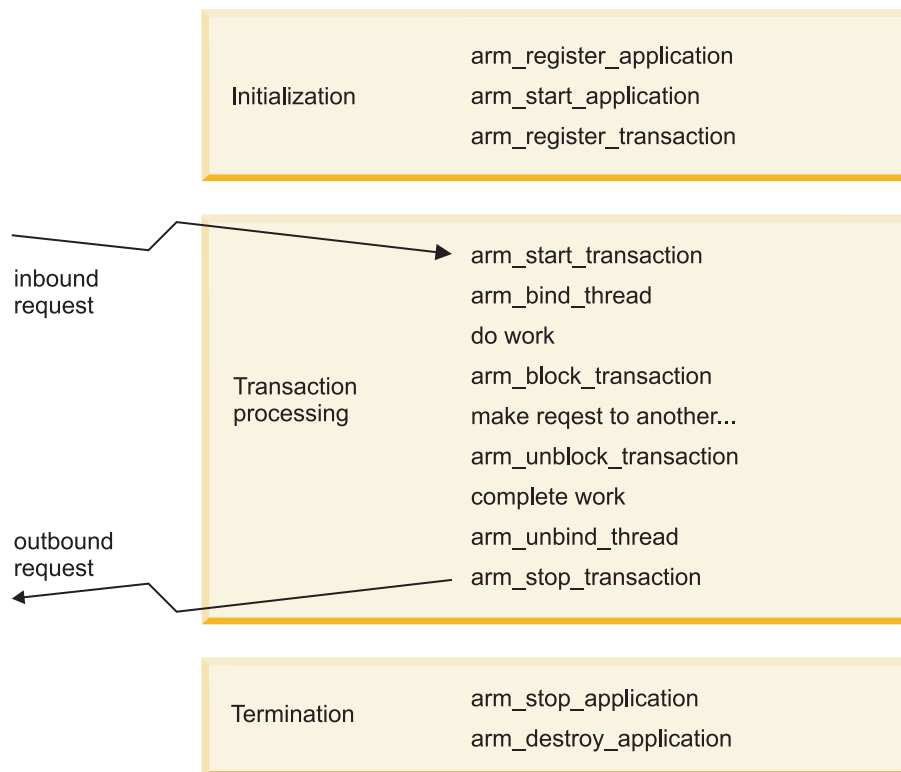
Note: While it is possible to bind the operating system to the i5/OS ARM library at runtime, the recommended technique is to bind to the ARM library at application-build time rather than during runtime for performance reasons.

Java™ applications use the Java native interface (JNI) to transform Java method calls into equivalent C function calls. You must configure the library search path to include the directory that contains the ARM C shared libraries. If the Java code fails to load the shared library because it was not configured properly, the method calls will be accepted; however, the underlying C function calls will not be made. As a result, applications such as EWLM will not collect data for the application.

The library routine exports the full set of ARM 4.0 interfaces, as defined by the Open Group standard. For a complete definition of the ARM 4.0 standard, refer to Open Group Application Response

Measurement - ARM  (www.opengroup.org/tech/management/arm/) Web site .

The following diagram depicts the core ARM 4.0 function calls that EWLM supports for capturing transaction data. Each of the function calls is described in greater detail in the API topic collection.



EICAA512-1

To accommodate the application instrumentation needs of systems management products, the ARM 4.0 library routine delivered by each operating system platform also provides an ARM adapter. The ARM adapter layer is unique to the EWLM implementation of ARM 4.0, and is not an element of the Open Group standard. The systems management product uses the adapter function to extend the basic capabilities of EWLM provided by the platform, such as accumulation of more detailed reporting data, sending alerts when unusual conditions are detected, or logging of performance statistics in a performance management database.

ARM-instrumented applications

As a general rule, applications for which instrumentation is necessary are those acting as application servers, that is, applications that service requests from multiple clients, typically in parallel. Middleware applications, such as DB2 Universal Database™ and WebSphere Application Server, are normally deployed in multi-tier customer solutions and can experience the types of problems that EWLM addresses. IBM currently provides ARM-instrumentation support on the following middleware applications:

- DB2 Universal Database, Version 8.2
- WebSphere Application Server, Version 5.1.1.1 or later
- IBM Web serving plug-ins

The WebSphere Application Server ships the following Web application servers, which are instrumented by default because they use the ARM-instrumentation framework provided by WebSphere Application Server. They include, but are not limited to, the following applications:

- Apache HTTP Server
- IBM HTTP Server
- Microsoft® Internet Information Services (IIS)

In addition, the following non-WebSphere Application Server Web serving plug-ins which use the same filter types as the WebSphere Application Server Web serving plug-ins, listed previously instrument the ARM 4.0 standard:

- | – Independent plug-in for Apache HTTP Server 2.0.47, or later
 - | – Independent plug-in for IBM HTTP Server 2.0.47, or later
 - | – IBM HTTP Server for iSeries™
 - | – EWLM-supplied plug-in for Internet Information Services (IIS) 5.0 and 6.0
- | Use this plug-in for static Web page serving only. It is not recommended for use in production environments. Deployment of this IIS plug-in for other than static Web page serving can result in inaccurate ARM-transaction-event reporting, and subsequently inaccurate EWLM interpretation of transaction data.

| **Application environment**

| Each application process registers with the ARM implementation on a specific platform supported by EWLM. This allows EWLM to dynamically construct a topology view of services deployed throughout the configuration. Certain, specific information passed by the ARM API calls is included in the filtering criteria that EWLM uses for transaction classification.

| An application is identified to EWLM by a set of properties, passed as strings, as follows:

- | • Registered application name that every instance of the application uses. This parameter can be a product name or a software element name. For example, the application name passed through ARM by WebSphere Application Server v5 is *WebSphere* and for v6 is *WebSphere:APPLICATION_SERVER*. The application name is a required ARM property that EWLM uses.
- | • Application group name that a collection of instances of the application uses. This parameter can be the name of a group of instances that process a shared workload. For example, a group name passed to ARM by the WebSphere HTTP plug-in can be something like *IBM HTTP SERVER/2.0.47*. The group name is an optional ARM property that EWLM uses, if available.

| The registered application name combined with the application group name comprise an *application environment*. The application environment provides the foundation for transaction data collection and reporting across the entire application topology. An application's filters are organized by application environment. The application specifies the application environment and its instrumented filters in an XML document to the domain manager so that the EWLM Control Center is aware of which filters apply to a specific application. The application filters are required when you create a domain policy so that you can create rules that use the filters to identify work requests.

| **Related concepts**

| “EWLM-supplied filter types” on page 35

| View this topic to learn about the filter types that are available to both applications and platforms. These are the filters that you can use when you create classification rules in a transaction, partition, or process class.

| “Application-specific filter types” on page 40

| View this topic to learn about filter types that are available to each predefined application. These filter types are in addition to the EWLM-supplied filter types that you can use in classification rules. These filters are defined in the application's filter policy XML file.

| “Enabling ARM on IBM-instrumented applications” on page 205

| Use this topic to learn how to enable ARM on supported middleware applications.

| **Related information**

| EWLM APIs

| View this topic collection to learn more about the EWLM APIs.

| WebSphere Application Server v5.1 Information Center

| View this Web site to learn more about WebSphere Application Server v5.1.

| WebSphere Application Server v6 Information Center

| View this Web site to learn more about WebSphere Application Server v6.

Data collection and aggregation

A valuable capability of EWLM is its ability to collect and aggregate performance data. This performance data collection capability allows EWLM to report to you if performance goals are met. In addition, it allows EWLM to make automatic resource adjustments if goals are not met.

The service policy in effect for an EWLM management domain provides the common reference point for collecting and aggregating performance statistics, which enable the following activities:

- Real-time reporting
- Real-time identification of potential performance problems
- Real-time navigation through data to identify the likely source of a problem
- Adjustment of partition workload group resources, if enabled
- Ability for EWLM to send load-balancer weight recommendations to a load balancer, if enabled

EWLM collects and aggregates performance statistics so that you can readily compare a high-level goal to actual performance at all times. You can also view detailed statistics that explain why the actual results were achieved as opposed to the desired results.

Each operating system instance within an EWLM management domain accumulates and summarizes performance and topology information locally so that it can be transmitted to the EWLM domain manager. The domain manager maintains a 24-hour history of aggregated information, at any given time.

The global performance and topology information is available to other systems management products through a set of programming interfaces that are publicly available on the EWLM domain manager. This allows any systems management product to extract, log, report, and analyze EWLM global statistics. The APIs accommodate simultaneous usage by multiple add-on products, independent of which (if any) product is acting as the primary systems management infrastructure for the customer environment. The various products might extract information at varying time intervals, without regard to the time periods that the EWLM domain manager uses to incorporate the results it receives from the participating operating system instances.

You can also use these publicly available programming interfaces to extract more detailed information from each operating system instance. As in the case of global EWLM statistics, multiple systems management products can extract this information during a time period that is appropriate for their specific usage. The statistics include more detailed resource usage and delay information, as well as correlation to specific operating system processes that support each middleware application.

Related concepts

“Real-time reporting” on page 51

Real-time reporting is the performance data that EWLM periodically updates in the EWLM Control Center.

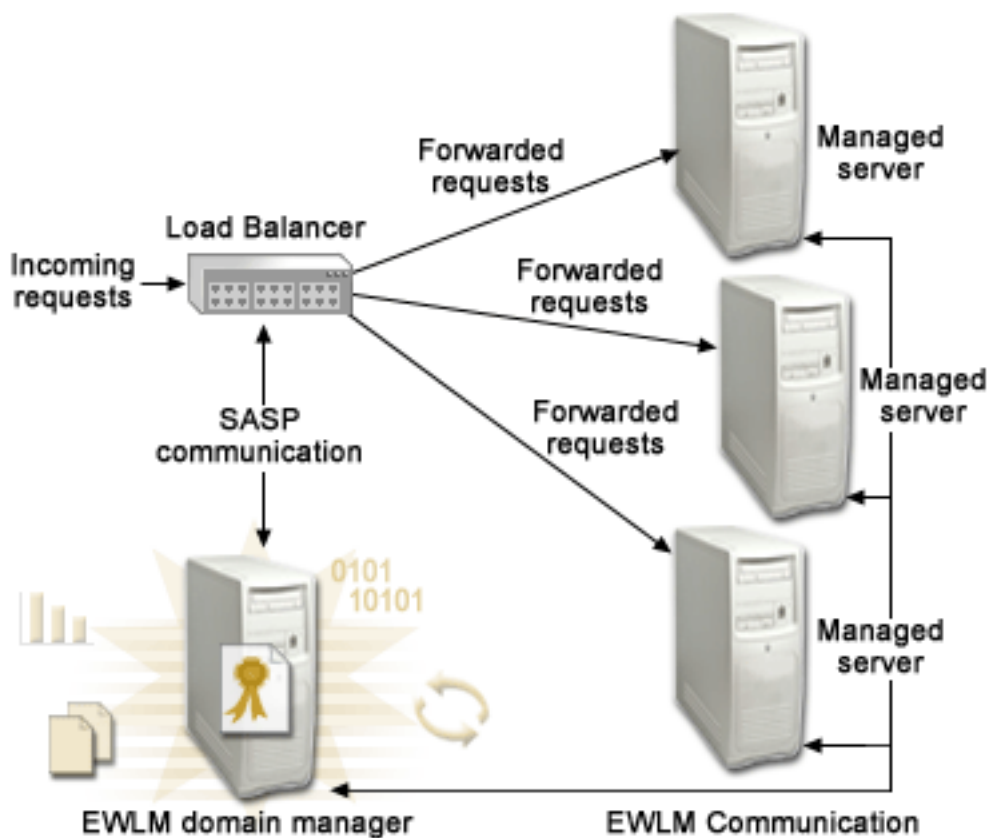
EWLM load balancing

Load balancing is an optional function of EWLM that is valuable if you want EWLM to send load balancing recommendations to a load balancer. This function allows the load balancer to evenly distribute incoming work to ensure that performance goals are met.

When applications in an EWLM domain are ARM-instrumented, EWLM can dynamically track the topology of work requests across multiple heterogeneous operating system platforms and application environments. The instrumentation also allows EWLM to compute intricate, application-specific metrics that are used to determine the following information:

- The history of work requests
- The likelihood of future work being completed
- The ability of the work to achieve goals on any particular system in the domain

- | If an application in the EWLM domain is not ARM-instrumented, the managed server can still provide the domain manager with a variety of advanced system-level statistics to develop such metrics.
- | System and workload management frameworks have proven to be valuable utilities that system administrators use to monitor and manage enterprise applications and the systems on which they run. While these systems have a great deal of knowledge about these applications, they generally have little ability to control the rate at which these applications receive work. In contrast, load balancers can control this rate, but typically have little information about the application's ability to handle the request. You can use EWLM's load balancing feature to advise Server/Application State Protocol (SASP) supported load balancers to balance workloads based on administrator-defined goals and the ability of the applications to satisfactorily handle the workloads. This feature provides for a merging of workload management and load balancing in a way that fills the void present in both.
- | The following diagram illustrates what a typical EWLM load-balancing scenario might look like:



- | When you use EWLM to provide load-balancing recommendations, an external load balancer is responsible for actually distributing the incoming requests to servers waiting to perform the work. The load balancer contacts the domain manager to obtain the weights which provide the EWLM recommendations for how to distribute incoming requests. The domain manager to load balancer communication uses the Server/Application State Protocol (SASP) that you might secure with SSL.
- | EWLM load-balancing weights are computed based on the following factors in the order specified:
 - | 1. **Machine or application capacity**
This metric is based on the remaining capacity of the servers.
 - | 2. **Success rate**

The percent of the most recent transactions that finished successfully. This avoids requests being sent to a server, failing, and then sending an error code back to the client. To a load balancer, the request has finished quickly, but in actuality the work has not successfully completed. EWLM recommends that load balancers divert work away from servers or applications where work is failing.

3. Topology requests

Load balancers and schedulers only have knowledge of the servers to which they are sending traffic. If the request is part of a distributed application, the request goes on to do subsequent work in application servers, database servers, and possibly transaction servers after it leaves the first server. EWLM knows the topology of work coming into each server and, to some degree, knows which servers the work will go to next. So, EWLM can influence load balancers to divert work from first-tier servers that have bottlenecks downstream.

4. Likelihood of the server or application to reach or exceed performance goals

EWLM recommends that load balancers and schedulers send work to servers that have a recent history of meeting their goals.

5. Business importance of the work currently running

EWLM recommends that load balancers and schedulers send work to a server that is running less important work, if load and system health are equal. This prevents new work from displacing work at the server that is running more important work.

The weights that EWLM sends reflect the business importance, topology, and ability of the server or application instance to meet its specified business goals.

EWLM load balancing is supported on the following load balancer:

- CISCO Content Switch Module for the CISCO Catalyst family of switches

Related tasks

“Enabling EWLM load balancing” on page 248

Use this topic to learn how to enable EWLM to communicate with a load balancer.

Load balancing communications:

To distribute workloads evenly among systems within your domain and ensure performance goals are met, EWLM must communicate with a load balancer.

EWLM communicates with load balancers through a binary protocol called Server / Application State Protocol (SASP). The load balancer starts all of the SASP connections. When EWLM connects to a load balancer, the load balancer can send requests to accomplish the following tasks:

- Register and deregister group members with the EWLM domain manager
- Receive weights for registered group members
- Quiesce group members
- Configure settings for the connection

The EWLM domain manager replies to all request messages that the load balancer issues in a reply message. If so configured, the EWLM domain manager might also send special weight messages to the load balancer without the load balancer issuing a get-weights request. These special weight messages have no reply. If either end of the connection is not synchronized, either the domain manager or load balancer can end the connection.

A load-balancing group member can be a particular application or an entire system. If the load balancer registers application-level members, EWLM determines if these applications are ARM-instrumented. Groups of ARM-instrumented applications provide EWLM with the most data to calculate weights and might be more effective given the application-specific nature of this data. System-level and non-ARM-instrumented applications will cause EWLM to use only system data when calculating load balancing weights.

| EWLM recommendations are passed to the load balancer in the form of integer weights. These weights
| are relative to each member of the group. It is expected that the load balancer use these weights as a
| distribution by which to determine the amount of work to send to each group member.

| “Enabling EWLM load balancing” on page 248

| Use this topic to learn how to enable EWLM to communicate with a load balancer.

| Server/Application State Protocol

| View this Web site for more information about the communication between workload managers and
| load balancers.

| Partition management

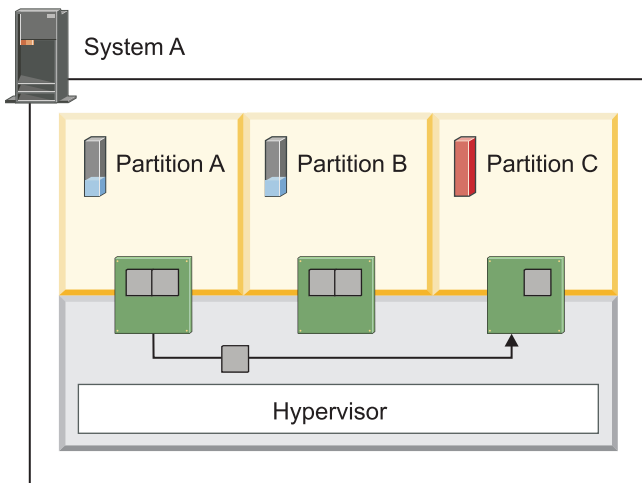
| *Partition management* refers to the ability of EWLM to automatically adjust the processing power among
| partitions within a partition workload group. The partitions in a partition workload group must reside on
| the same physical machine.

| If you implement this function of EWLM, EWLM automatically adjusts the processing units among the
| partitions without you having to do anything. If EWLM determines that the processing units for a
| partition need to be adjusted, EWLM adheres to the following when making adjustments:

- | • EWLM distributes only processing units from the shared processing pool to partitions in the partition
| workload group.
- | • EWLM will not reduce a partition’s processing units below the partition’s minimum processing units
| setting or raise a partition’s processing units above the maximum processing units setting. This is true
| for both capped and uncapped partitions.
- | • EWLM uses the partition weight if there is contention for the available processing power.

| The EWLM partition management function identifies the partitions that need more processing power and
| determines how much of the processing power in the shared pool to give to the partition. In addition, if
| more than one partition is in need of more processing power, EWLM determines what fraction of the
| shared processing power to give to each partition.

|
|



EICAA534-0

| . The following graphic includes one physical system that contains three partitions that EWLM is managing. This
| graphic illustrates how EWLM can move processing power from the shared processing pool to a partition that is in
| need.

| The *partition weight* specifies a number that indicates how to distribute shared processing power among
| uncapped partitions. Any available unused capacity is distributed to contending partitions in proportion
| to the established value of the partition weight, where 255 is the highest weight value. EWLM adjusts the
| partition weight based on the workload’s performance.

| EWLM adjusts the processing power for both capped and uncapped partitions, but not dedicated partitions.

| Use the EWLM Control Center Partition Workload Group details report to view the number of processing units that a partition is currently entitled to use. EWLM automatically adjusts the **Processing units** field in the managed server details report to reflect the resource adjustment changes that it makes. In addition, you can use the Hardware Management Console to display the changes that EWLM makes to the processing units.

| If you do not implement the EWLM partition management function, you can use the views available in the Monitor section of the EWLM Control Center Monitor to determine where performance problems exist. Then, use your systems management interfaces to adjust resources as you determine necessary.

| **Related concepts**

| “Partition workload group” on page 16

| A *partition workload group* identifies a set of partitions that reside on the same physical system. The partition profile specifies the name of the partition workload group that it belongs to, if applicable.

| **Related tasks**

| “Enabling partition management” on page 249

| Use this topic to learn how to enable logical partition management in EWLM.

| **Related information**

| Shared processors in iSeries Information Center

| View this topic in the iSeries Information Center to learn more about shared processors.

| **Coexisting workload management products**

| Each environment has a different set of workload management tools available to use, in addition to EWLM.

| You need to examine each of these tools to ensure that the tool or tools that you use meets your workload management needs. Each workload management tool offers different capabilities. You must identify which tool is appropriate for your needs and determine which can function with EWLM, if installed on the same system.

| View the following table to learn more about workload management products, their capabilities, and if they can reside on the same system as EWLM.

Table 1. List of workload management tools.

Workload management tool	Description	Supported environments	Can the tool adjust system resources (CPU or memory) ?	Can the tool balance the distribution of workloads?	Can the tool exist on the same system as EWLM?	EWLM considerations
Enterprise Workload Manager	Performance management tool that automatically adjusts system resources and makes recommendations to load balancers on how to distribute incoming work to ensure that user-defined performance goals are met.	iSeries, pSeries®, zSeries®, xSeries®, Hewlett-Packard, SUN, Windows	Yes, automatically	Yes, with an external load balancer		
Dynamic logical partitioning (LPAR) and Micro-Partitioning™ 1	Dynamic logical partitioning enables you to manually move partition resources without requiring a restart of the system or affected partitions. The Micro-Partitioning function enables logical partitions to use unused processing resources in the shared processor pool. 1	iSeries, pSeries	Yes, manually through dynamic LPAR or automatically through Micro-Partitioning	No	Yes	Dynamic LPAR makes adjustments when CPU thresholds are not met. In contrast, EWLM adjusts CPU processing power on demand to ensure performance goals for <i>specific</i> workloads are achieved.

Table 1. List of workload management tools. (continued)

Workload management tool	Description	Supported environments	Can the tool adjust system resources (CPU or memory) ?	Can the tool balance the distribution of workloads?	Can the tool exist on the same system as EWLM?	EWLM considerations
Dynamic logical partitioning (LPAR)	Dynamic logical partitioning enables you to manually or automatically move partition resources without requiring a restart of the system or affected partitions.	zSeries	Yes, manually or automatically	No	Yes	Dynamic LPAR makes adjustments when CPU thresholds are not met. In contrast, EWLM adjusts CPU processing power on demand to ensure performance goals for specific workloads are achieved.

Table 1. List of workload management tools. (continued)

Workload management tool	Description	Supported environments	Can the tool adjust system resources (CPU or memory) ?	Can the tool balance the distribution of workloads?	Can the tool exist on the same system as EWLM?	EWLM considerations
Partition Load Manager (PLM)	Partition Load Manager (PLM) allocates resources to partitions on demand within the constraints of a user-defined policy.	pSeries	Yes, automatically.	No	Yes, as long as they do not manage the same LPAR.	PLM manages partition CPU usage on demand. In contrast, EWLM adjusts CPU processing power on demand to ensure performance goals for <i>specific</i> workloads are achieved. In short, PLM is resource orientated and EWLM is goal orientated. IBM does not recommend that PLM and EWLM manage the same partition's entitled capacity, virtual CPU usage, or workload weight values at the same time.

Table 1. List of workload management tools. (continued)

Workload management tool	Description	Supported environments	Can the tool adjust system resources (CPU or memory) ?	Can the tool balance the distribution of workloads?	Can the tool exist on the same system as EWLM?	EWLM considerations
z/OS® Workload Manager (WLM)	z/OS Workload Manager (WLM) is the product from which EWLM originated. WLM monitors workloads on the z/OS system according to the user-defined performance goals in a service definition.	zSeries	Yes, automatically	Yes, with an external load balancer	Yes	WLM only monitors work on the z/OS operating system. EWLM monitors work that spans system boundaries and operating systems.
<p>Note:</p> <ol style="list-style-type: none"> The Micro-Partitioning function is available only as part of the Advanced POWER™ Virtualization hardware feature. 						

Related information

<http://publib-b.boulder.ibm.com/abstracts/sg246785.html?Open>

EWLM concepts

Before taking advantage of the full capability of EWLM, you need to understand the concepts of EWLM.

Classification

Classification refers to the process that EWLM uses to assign work to a transaction, partition, or process class.

To properly classify work, EWLM first identifies what type of work it is monitoring, such as an application-level transaction or an operating system process. Based on the type of work, EWLM classifies the work to a corresponding transaction class or process class. In addition, if EWLM assigns a transaction to a transaction class or assigns an operating system process to a process class, the work is also assigned to a partition class, if the work is processed by a partition that EWLM is to monitor.

In EWLM, there are three types of classifications: transaction classification (application-level transactions), process classification (operating system processes), and partition classification (partition). These classification types correspond directly to the categories of work that EWLM monitors as described in Monitoring work.

Transaction classification

Transaction classification is the process in which EWLM assigns application-level transactions to transaction classes. To do this, EWLM compares the transaction’s attributes to the classification rules defined in the application’s transaction classes. If a transaction’s attributes meet the conditions of the classification rules, EWLM assigns the transaction to that transaction class. Then, EWLM monitors the performance of that transaction against the performance goal defined for that transaction class.

Transaction classification occurs when the first ARM-instrumented application in the EWLM domain receives the transaction. This is referred to as the *entry application*. Therefore, it is essential to define transaction classes in the domain policy for any application that will serve as an Entry application.

Process classification

Process classification is the process in which EWLM assigns operating system processes to process classes. To do this, EWLM compares the process's attributes to the classification rules defined in the platform's process classes. If an operating system process meets the conditions of the classification rules, EWLM assigns the process to that process class. If it does not meet the conditions of any process class, EWLM does not assign it to a process class because process classes are optional. If EWLM assigns it to a process class, EWLM monitors the performance of that process against the performance goal defined for that process class.

Process classification occurs when the process is initiated by a managed server that is part of the EWLM domain. EWLM immediately classifies the process according to the process classes defined for that managed server's operating system.

Partition classification (includes both transactions and processes)

Partition classification is the process in which EWLM assigns all work that a partition processes to a partition class. A partition class allows EWLM to collect overall resource usage data about a partition. Only partitions that are eligible for partition workload group management are eligible to be assigned to a partition class.

Transactions or processes that run in a partition that is assigned to a partition class can also be classified to transaction class or process classes. For example, if a domain policy contains a transaction class for WebSphere and the domain policy contains a partition class server on which WebSphere runs, EWLM will classify the work that WebSphere processes to both the transaction class and the partition class. This allows you to monitor work processed by the WebSphere application as part of an overall partition workload and as separate workload that only includes WebSphere transactions.

The classification rules in the transaction class, process class, or partition class determine which work EWLM is to assign to the class.

Related task

"Monitoring work" on page 17

EWLM monitors application-level transactions and operating system processes or both.

Related concept

"Classification rule" on page 33

A *classification rule* is a way to identify work that the applications and servers in your domain process. You create classification rules in transaction, process, and partition classes. A classification rule comprises a filter type, operation, and value. You can create multiple classification rules to create a conditional set of rules.

"Transaction classes" on page 48

A *transaction class* contains classification rules that identify application-level transactions. Each transaction class also specifies a service class, which defines the performance goal, that the transactions performance is to meet. If a transaction's attributes meet the conditions of the classification rules, then EWLM assigns the transaction to the class.

"Process classes" on page 49

A *process class* contains classification rules that identify a group of processes that run on a specific operating system. Each process class also specifies a service class performance goal that the operating system processes are to adhere to. If an operating system process meets the conditions of the classification rules, then EWLM assigns the process to the class.

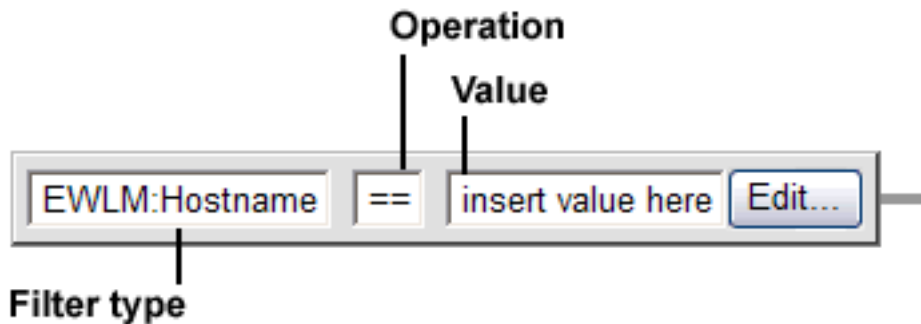
"Partition classes" on page 50

A *partition class* identifies partitions that are to be managed with a single goal and business importance. The partition class is associated with a service class that contains the goal and business

importance. You use partition classes as part of EWLM's partition workload group management. EWLM manages the CPU resources based on the overall performance of partitions in a partition class.

Classification rule

A *classification rule* is a way to identify work that the applications and servers in your domain process. You create classification rules in transaction, process, and partition classes. A classification rule comprises a filter type, operation, and value. You can create multiple classification rules to create a conditional set of rules.



The following list describes each component of a classification rule:

- **Filter type:** Indicates which attribute is used to identify the work. For more information about possible filter types that you can use to identify work, see “Filter” on page 35.
- **Operation:** Indicates if the attributes of the work being monitored is to equal or not equal the value.
- **Value:** Specifies a value for the rule. The value can include wildcard and mask values. You may see these special values used in other locations of the Control Center; however, this is the only user-defined field in which the special values are allowed.

- (*) Wildcard

Use (*) at the end of a filter value to represent any character and an unlimited number of characters. You can only use the wildcard value at the end of a value because it represents an unlimited number of characters. For example, *department(*)* or *database(*)*.

- (\?) Mask

Use (\?) to represent any single character in a filter value. In addition, you can specify the mask value multiple times in the value. For example, *bank(\?)teller* or *Admin(\?)(\?)User*.

Furthermore, you can combine wildcard and mask values in a filter value. For example, *\sec\file\(\?)(*)*, *department2(\?)9(*)*, or *user(\?)(\?)23(*)*.

When you create classification rules, the specific rules that you define depend on the job attributes for the work that the applications and servers in your EWLM domain process. This topic provides example classification rules that include the filter type, operation, and value.

If you do not want to identify specific work on your domain, you can use the default transaction class that specifies a wildcard value. Then, EWLM monitors all work, regardless of the classification rules. This makes it easy for EWLM to monitor the work in your EWLM domain without you having to identify specific work that the domain processes. In a similar manner, you can create a partition class or process class that uses the wildcard value. You must create these classes with a wildcard value. EWLM does not provide a default class for these because they are optional ways to monitor work.

If you can or want to identify specific work that your domain processes, you must define classification rules that identify that work. Then, EWLM can monitor the work relative to the service class goal that you define.

Classification rule details

When you create classification rules, the specific rules that you define depend on the job attributes for the work that the applications and servers in your EWLM domain process. This information provides example classification rules that include the filter type, operation, and value.

If you want to identify specific work on your domain, you can use the default transaction class that specifies a wildcard value. Then, EWLM monitors all work, regardless of the classification rules. This makes it easy for EWLM to monitor the work in your EWLM domain without you having to identify specific work that the domain processes. In a similar manner, you can create a partition class or process class that uses the wildcard value. You must create these classes with a wildcard value. EWLM does not provide a default class for these because they are optional ways to monitor work.

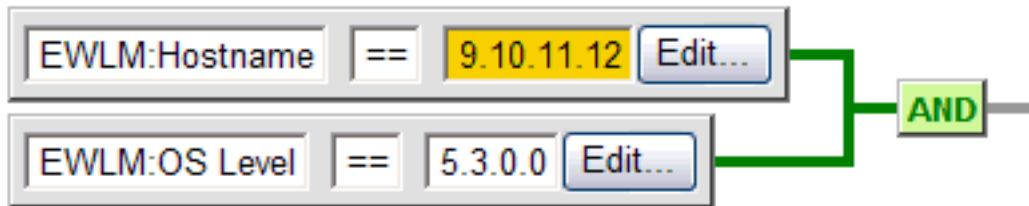
If you can or want to identify specific work that your domain processes, you must define classification rules that identify that work. Then, EWLM can monitor the work relative to the service class goal that you define.

Use the following example classification rules to learn about various ways to join rule statements. In addition, note that each classification rule, which may contain more than one rule statement, can specify separate service classes. This allows you to assign different performance goals to different groups of work.

Example classification rules

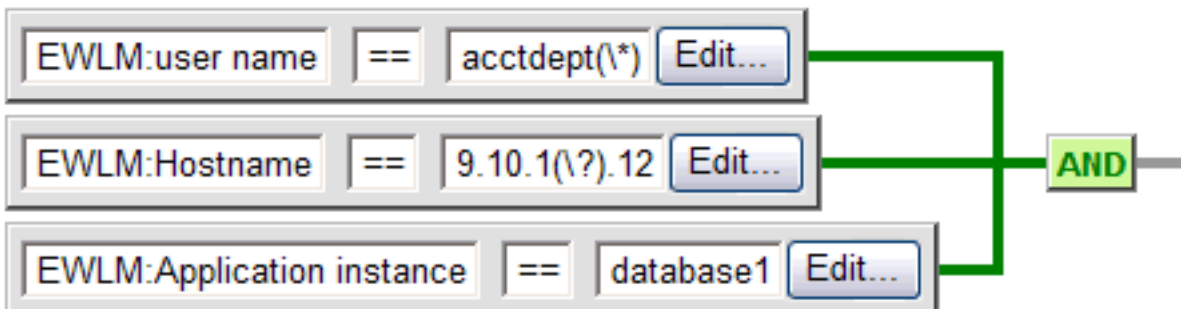
Example 1

The following is a classification rule that contains two rule statements joined by an AND logical operator. Work must match both rule statements for EWLM to assign it to this class.



Example 2

The following is a classification rule that contains three rule statements joined by an AND logical operator. Work must match all three rule statements for EWLM to assign it to this class. In addition, these rules demonstrate how you can use the wildcard (*) and mask (\?) values.



Note:

1. (*) represents a wildcard value. Use it to represent any character and an unlimited number of characters. This string is only allowed at the end of a value.
2. (\?) represents a mask value. Use it to represent any single character. Use this string anywhere in a filter value: the beginning, middle, or end.
3. If a work request matches the conditions of the rule statement, EWLM assigns it to the class.

Filter

A *filter* is used by EWLM to identify work that applications and platforms in your EWLM domain process. The filters that you can use in a classification rule depend on the filters that you define when you set up the application or platform.

A filter type is a specific filter that you use to identify a work request. Each platform and application that EWLM monitors provides a different set of filter types that you can use. In addition, EWLM supplies a list of filter types that you can use with any application or platform. The filter types that you can use for each application or platform are defined in the application's or platform's definition file, which might also be referred to as a filter policy.

EWLM provides a set of applications and platforms that are set up in the EWLM Control Center. This means that the application and platform filter policies define the filters that you can use to monitor the work that the application or platform processes.

For more information about what filter types are available and how to use a filter in a transaction class or process class rule, see the following subtopics

“Managing an application filter policy” on page 276

The application filter policy defines the application-specific filter types that you can use in classification rules to identify transactions for EWLM to monitor. Each application that you want EWLM to monitor must have an application filter policy. The EWLM Control Center contains a predefined list of applications and their filter policies. However, you may need to edit these application filter policies or create a new application filter policy for an application that is not predefined in the Control Center.

“Classification rule” on page 33

A *classification rule* is a way to identify work that the applications and servers in your domain process. You create classification rules in transaction, process, and partition classes. A classification rule comprises a filter type, operation, and value. You can create multiple classification rules to create a conditional set of rules.

EWLM-supplied filter types:

View this topic to learn about the filter types that are available to both applications and platforms. These are the filters that you can use when you create classification rules in a transaction, partition, or process class.

These filters are predefined in EWLM. Therefore, when you define an application you can select which of the EWLM-supplied filter types to include in the application's filter policy definition. The list of EWLM-supplied filter types varies slightly between application filter definitions and platform filter definitions. Note that this list is not an all inclusive list and use of these filter types depends on the application or platform for which the filter is used. The EWLM-supplied filters are derived from the standard ARM 4.0 properties.

If you encounter any problems using these filters, refer to your application or operating system documentation to verify which filters are supported.

The following table lists the EWLM-supplied filters that you can use in a classification rule in a transaction, process, or partition class, except where noted.

Filter types for applications or platforms ¹				
Filter type ³	Description	Available platforms	Maximum length	Transaction, partition, or process ²
EWLM: OS Platform	The name of the operating system, such as AIX or i5/OS.	Any	255	Transaction and process
EWLM: OS Level	The release level of the operating system. For example, AIX 5.2.0.0 or z/OS 1.6.0.0.	Any	255	Transaction, partition, and process
EWLM: Hostname	A unique name by which a server is known on a network. The fully qualified host name consists of the host name and the TCP domain name. The TCP domain name specifies a group of systems that share a common suffix (domain name). For example, if the fully qualified host name is <i>server1.us.ibm.com</i> , <i>server1</i> is the host name and <i>us.ibm.com</i> is the TCP domain name that contains the <i>server1</i> host name. The fully qualified host name uses a character-based format such as <i>server1.us.ibm.com</i> . However, some servers might use a numerical-based name. The numerical-based host name is not the same as the system's IP address.	Any	255	Transaction, partition, and process
EWLM: Cluster Name	The name of the operating system cluster if the operating system uses clusters. For z/OS, the cluster name is the sysplex name.	Any	255	Transaction, partition, and process
EWLM: System Name	The system's name for the operating system instance. For example, in AIX and i5/OS, this is the name of the node.	Any	255	Transaction, partition, and process
EWLM: Management Domain Name	The name of the EWLM domain.	Any	255	Transaction, partition, and process
EWLM: Application instance	The application instance name from an <i>arm_start_transaction</i> call from the product that first processes the work request. The application instance might be a server application that either routes the work request to another server or a server application that completes the work request.	Any	127	Transaction
EWLM: Transaction name	The registered transaction name from an <i>arm_register_transaction</i> call from the product that first processes the work request. The name is generated by the middleware application. It contains alphanumeric characters.	Any	127	Transaction
EWLM: user name	The user name, as understood by the instrumented application, that initiates the work request. This is the user name specified in the user sub-buffer of an <i>arm_start_transaction</i> call.	Any	127	Transaction

Filter types for applications or platforms ¹				
Filter type ³	Description	Available platforms	Maximum length	Transaction, partition, or process ²
EWLM: URI	<p>The Uniform Resource Identifier (URI) is how a point of content is identified on the Internet. The URI syntax depends on the scheme. Most URIs follow this general format:</p> <p><scheme>://<authority><path>?<query></p> <p>where the parameters are defined as follows:</p> <ul style="list-style-type: none"> <i>scheme</i> (required) specifies the type of scheme used, such as http, ftp, mailto, or urn. <authority><path>?<query> represents the scheme-specific parts whose format is determined by the specifications of the <i>scheme</i>. <p>The EWLM: URI value does not need to contain the entire URI value. You can identify a URI value by specifying only a portion of the value.</p> <p>The URI string value is specified in the transaction identity sub-buffer of an arm_register_transaction or arm_start_transaction call.</p>	Any	4095	Transaction
<p>Note:</p> <ol style="list-style-type: none"> These EWLM-supplied filter types are derived from EWLM or the ARM 4.0 standards. You must refer to your application or operating system documentation for a master list of filters that the application or operating system supports. These filters are based on ARM 4.0 standards but does not indicate that all applications and operating systems implement them. z/OS does not support process classes. Filter types that begin with EWLM: are reserved for EWLM use. Do not use this prefix in any applications instrumented with ARM 4.0. If EWLM: is used, the filter type might be overwritten by EWLM. 				

Related information

Uniform Resource Identifier
http://en.wikipedia.org/wiki/Uniform_Resource_Identifier

Filter values for the predefined platforms:

View this topic to view the values that each platform returns for each of the EWLM-supplied filter types.

Use these values when you define a classification rule to identify specific work for EWLM to monitor.

Predefined platforms and their associated filter values for each filter type identified					
EWLM: OS Platform	EWLM: OS Level	EWLM: Hostname	EWLM: Cluster Name	EWLM: System Name	EWLM: Management Domain Name
AIX	5.2.0.0 , 5.3.0.0	Defined by the user. For example, server1.ibm.com	Not applicable	Defined by the user.	Defined by the user when domain manager is configured.
HP-UX	B.10.10	Defined by the user. For example, 9.11.21.6 or server1.ibm.com	Not applicable.	Defined by the user.	Defined by the user when domain manager is configured.

Predefined platforms and their associated filter values for each filter type identified					
EWLM: OS Platform	EWLM: OS Level	EWLM: Hostname	EWLM: Cluster Name	EWLM: System Name	EWLM: Management Domain Name
Linux	2.6.5-7.191	Defined by the user. For example, myserver.mycompany.com	Not applicable	Defined by the user.	Defined by the user when domain manager is configured.
OS/400® (i5/OS) ¹	5.3.0.0 , 5.4.0.0	Defined by the user. For example, www.ibm.com	Defined by the user.	Defined by the user.	Defined by the user when domain manager is configured.
Solaris	5.9.0.0	Defined by the user. For example 9.11.26.1	Defined by the user.	Defined by the user.	Defined by the user when domain manager is configured.
Windows	5.0.x.y (Windows 2000), 5.2.x.y (Windows 2003) ²	Defined by the user. For example, 9.111.26.1	Defined by the user.	Defined by the user.	Defined by the user when domain manager is configured.
z/OS	1.6.0.0 , 1.7.0.0	Defined by the user. For example, server1.ibm.com	The sysplex name. Defined by the user.	The MVS™ system name. Defined by the user.	Defined by the user when domain manager is configured.
Note: 1. Note that the operating system returns OS/400 as the value for V5R3 i5/OS. 2. The letter <i>x</i> represents the major version of the Windows Service Pack and <i>y</i> represents the minor version of the Windows Service Pack.					

Platform-specific filter types:

View this topic to learn about filter types that are available to each supported operating system. These filter types are in addition to the EWLM-supplied filter types that you can use in classification rules. These filters are defined in the platform's XML file.

Use these filters when you define classification rules in a process class to identify the processes for EWLM to monitor.

The following table outlines the filter types that are available to specific operating systems. In addition, you can determine what the filter value's maximum length is and if a wildcard value is supported.

Table 2. Platform-specific filter types

Platform-specific filter types				
Filter type	Description	Available platforms	Maximum length	Supports wildcard (*)
CommandLine Arguments	The parameter of the command.	HP-UX	64 characters minus the length of the executable path	Yes
		Solaris	80 characters minus the length of the executable path	Yes
		Windows	256	Yes
ExecutablePath	<p>The process as it was launched that specifies the entire path.</p> <p>Windows example:</p> <p>driveletter:directoryName\subdirectoryname\filename.suffix</p> <p>AIX, HP-UX, and Solaris example:</p> <p>/directoryName/subdirectoryname/filename.suffix</p> <p>Note: For AIX, HP-UX, and Solaris, the paths do not specify a drive letter. They use forward slashes (/) instead of backward slashes (\), and begin with a forward slash (/).</p>	AIX	256	No
		HP-UX	64	Yes
		Linux	256	No
		Solaris	80	No
		Windows	256	Yes
GroupName	The group name that is processing the program.	AIX	8	No
		HP-UX	8	Yes
		Linux	8	No
		Solaris	8	No
UserName	The user name that is processing the program.	AIX	8	No
		HP-UX	8	Yes
		Linux	8	No
		Solaris	8	No
		Windows	20	Yes
WLMTag	The tag passed from an application to the AIX workload manager.	AIX	30	Yes
Effective Group Profile	The name of the group profile that is currently used by the job or *NONE. The value is a character string.	i5/OS	8	Yes
		Linux	10	Yes

Table 2. Platform-specific filter types (continued)

Platform-specific filter types				
Filter type	Description	Available platforms	Maximum length	Supports wildcard (*)
Effective User Profile	The name of the user profile which is currently used by the job or *NONE. The value is a character string.	i5/OS	8	Yes
		Linux	10	Yes
Command Base Name	The base name of the command.	Linux	16	Yes
		HP-UX	16	Yes
Job Accounting Code	The name of the job accounting code. The value is a character string.	i5/OS	10	Yes
Job Description Name	The job description name. The value is a character string.	i5/OS	10	Yes
Job Name	The simple job name. The value is a character string.	i5/OS	10	Yes
Job User Name	The user name part of a qualified job name. The value is a character string.	i5/OS	10	Yes
Subsystem Name	The name of the subsystem. The value is a character string.	i5/OS	10	Yes

EWLM makes internal use of filter name strings that begin with the substring *EWLM:*. Therefore, IBM strongly recommends that applications instrumenting for ARM 4.0 not specify any ARM 4.0 application identity or context property names nor any transaction identity or context property names that begin with substring *EWLM:* to avoid filter name errors.

This is not a master list of filter types that you can use when defining classification rules for a specific operating system. The filter types that you can use depend on the filter types available in the ARM instrumentation and which filter types the operating system supports. Refer to the operating system product documentation for more information on the filter types that are available. In addition, refer to the operating system product documentation for more information on valid filter values when you use the filter types. For example, the syntax of the filter value might vary by operating system.

- IBM eServer pSeries and AIX Information Center
(http://publib16.boulder.ibm.com/pseries/en_US/infocenter/base/)
- IBM iSeries Information Center
(<http://www.ibm.com/eserver/series/infocenter>)
- IBM z/OS V1R6.0 LibraryCenter
(http://publibz.boulder.ibm.com/bookmgr_OS390/libraryserver/zosv1r6/)
- SUN Solaris Documentation
(<http://www.sun.com/documentation/>)

Application-specific filter types:

View this topic to learn about filter types that are available to each predefined application. These filter types are in addition to the EWLM-supplied filter types that you can use in classification rules. These filters are defined in the application’s filter policy XML file.

Use these filters when you define classification rules in a transaction class to identify the transactions for EWLM to monitor. Work with the filters defined in an application’s filter policy in the EWLM Control Center by selecting **Set up** → **Applications**.

The following table outlines select filter types that are available to specific applications.

Table 3. Application-specific filter types

Application-specific filter types ¹		
Filter type	Description	Application
EJBname	For a description, see the related information section of this topic.	IBM WebSphere Application Server
HostInfo	The authority segment of the URI, not including the port information.	IBM Web serving plug-in
PluginType	The type of plug-in, for example, WebSphere or IHS/Apache or IIS.	IBM Web serving plug-in
Port	The TCP/IP port on which the request arrived, specified as a string representation of the decimal value.	IBM Web serving plug-in, IBM WebSphere Application Server
Protocol	The protocol the client used to connect to the plug-in, for example, HTTP 1.1 or HTTP/1.1. Protocol is specified as the scheme plus the level.	IBM Web serving plug-in
QueryString	The query segment of the URI string that is the portion of the dynamic URI that contains the search parameters of the request, not including the ? character. For example, http://www.ibm.com/support?p1=a&p2=b QueryString = p1=a&p2=b	IBM Web serving plug-in, IBM WebSphere Application Server
RemoteAddress	The request client's TCP/IP address specified as a string.	IBM Web serving plug-in
RemoteUser	The user name associated with the request specified as a string.	IBM Web serving plug-in
Scheme	The Web server scheme associated with the segment of the URI, for example, http or https.	IBM Web serving plug-in
ServerName	The name assigned to the server or virtual server to replace the actual host name of the machine. This applies only to the WebSphere plug-ins.	IBM Web serving plug-in
ServerVersion	The version of the Web server. This filter applies only to Internet Information Services (IIS).	IBM Web serving plug-in
Note:		
1. This is not an all-inclusive list of possible filter types that you can use when defining transaction classes for the applications specified. For example, filters for IBM DB2 Universal Database and WebSphere Application Server v6.0 are not listed here. Refer to your application product documentation to obtain a complete list of filters that the application's ARM support provides.		

Note:

1. The application filter types for IBM DB2 Universal Database are not supported by i5/OS. To monitor DB2 transactions on an i5/OS system, you can use the application group name as well

as the following EWLM-supplied filters: EWLM: OS Platform, EWLM: OS Level, EWLM: Hostname, EWLM: Cluster Name, EWLM: System Name and EWLM: Management Domain Name.

2. The application filter types for IBM DB2 Universal Database are not supported by z/OS. For more information, see “Enabling ARM on DB2 Universal Database for use by WebSphere Application Server on z/OS” on page 213.

This is not a master list of filter types that you can use when defining classification rules for specific applications. The filter types that you can use depend on the ARM APIs for the application you want EWLM to monitor.

EWLM applies an application’s ARM properties, the identity and context, as filters in EWLM classification. Refer to the application product documentation for more information on the ARM properties that are available. In addition, refer to the application product documentation for more information on valid filter values when you use the supported ARM properties as filters in EWLM.

Related concepts

“Application instrumentation” on page 20

Instrumenting an application with the appropriate APIs allows EWLM to properly monitor the application-level transactions.

Related references

WebSphere Application Server v5.1

(<http://publib.boulder.ibm.com/infocenter/ws51help/>)

WebSphere Application Server v6.0

(<http://publib.boulder.ibm.com/infocenter/ws60help/>)

WebSphere Application Server v6.0 ARM application properties and transaction context data

(http://publib.boulder.ibm.com/infocenter/wasinfo/v6r0/topic/com.ibm.websphere.nd.doc/info/ae/ae/rprf_armproperties.html)

IBM HTTP Server

(<http://www-306.ibm.com/software/webservers/httpservers/library>)

DB2 UDB preparing to work with IBM Enterprise Workload Manager (EWLM)

([http://www-1.ibm.com/support/docview.wss?rs=71&context=SSEPGG&dc](http://www-1.ibm.com/support/docview.wss?rs=71&context=SSEPGG&dc=DB520&dc=D600&dc=DB530&dc=D700&dc=DB500&dc=DB540&dc=DB510&dc=DB550&q1=EWLM&uid=swg21213956&loc=en_US&cs=utf-8&lang=en)

[=DB520&dc=D600&dc=DB530&dc=D700&dc=DB500&dc=DB540&dc=DB](http://www-1.ibm.com/support/docview.wss?rs=71&context=SSEPGG&dc=DB520&dc=D600&dc=DB530&dc=D700&dc=DB500&dc=DB540&dc=DB510&dc=DB550&q1=EWLM&uid=swg21213956&loc=en_US&cs=utf-8&lang=en)

[510&dc=DB550&q1=EWLM&uid=swg21213956&loc=en_US&cs=utf-8&lang=en](http://www-1.ibm.com/support/docview.wss?rs=71&context=SSEPGG&dc=DB520&dc=D600&dc=DB530&dc=D700&dc=DB500&dc=DB540&dc=DB510&dc=DB550&q1=EWLM&uid=swg21213956&loc=en_US&cs=utf-8&lang=en))

Module mod_arm4_ap20

View this topic to learn more about the i5/OS specific values used in the ARM directives.

Correlator

Tracking the flow of work from application to application requires that contextual information be associated with each transaction. This contextual information is called a *correlator*.

Each ARM 4.0 instrumented application provides interfaces to accept correlators with inbound transactions, and uses such interfaces in other applications to send correlators with outbound transactions.

What is perceived as one transaction can be implemented as many subtransactions across several different applications. To report data in a manner that matches the end-to-end view, there needs to be something that relates the sub-transactions with one another. In ARM-instrumented applications, this is accomplished by using correlators.

A correlator is a byte array that carries information from one subtransaction to another. Correlators are built by EWLM and applications are not expected to interpret their contents. They simply pass the correlators along with their transactions. Correlators vary in size from one provider’s ARM implementation to another. The flow of a request through a series of applications can be viewed as a call

tree of parent and child transactions. Each transaction in the tree has a unique correlator. Relationships are made by passing the correlator of the parent transaction to its child transactions. The application passes the parent correlator to EWLM on an `arm_start_transaction()` call, and receives a current correlator back. EWLM uses the correlator to keep track of both end-to-end and local information. An example of end-to-end information is the name of the transaction class.

EWLM propagates this information from the parent correlator to the child correlator so that the data for all transactions in the call tree is collected on behalf of the same transaction class and service class goal. The movement of the work request from a parent to a child correlator represents a *hop*. You can view performance statistics for each hop that processes work requests for a transaction.

An example of local information is the application environment relationship, which represents the current depth into the call tree. The correlator is what enables EWLM to collect data for a distributed application and present it in a meaningful way.

“Monitoring work” on page 17

EWLM monitors application-level transactions and operating system processes or both.

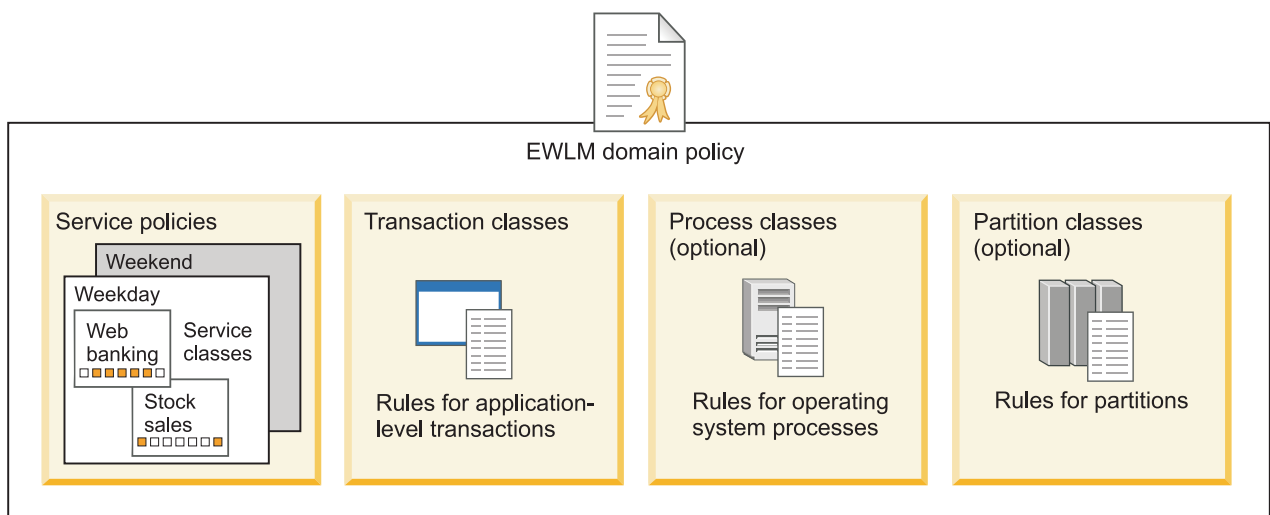
Domain policy

A *domain policy* is a high-level container that specifies performance goals for work processed in the EWLM domain.

The domain policy contains the following:

- Application definitions whose transactions EWLM is to monitor.
- Platform definitions whose operating system processes EWLM is to monitor, optional.
- Transaction classes that identify application-level transactions for EWLM to monitor.
- Process classes that identify operating system processes for EWLM to monitor, optional.
- Partition classes to identify partitions that EWLM is to monitor, optional.
- Performance goals defined in service classes that correspond to your business objectives or business partners’ service level agreements.
- Multiple service policies if you want to change a service class goal from one service policy to another.

The following graphic shows the relationship between the components of a domain policy. The domain policy contains service policies, service classes, transaction classes, and, optionally, process and partition classes.



EICAA501-1

Within your EWLM domain, you can have multiple domain policies. However, you can only deploy one domain policy per domain at a time. Deploying a domain policy refers to making the domain policy

available to the domain manager. In addition, only one of the service policies in the domain policy can be active at a time. An active service policy means that EWLM is monitoring the work requests that the EWLM domain processes according to the classification rules and goals in the active service policy.

Related concepts

“Partition workload group” on page 16

A *partition workload group* identifies a set of partitions that reside on the same physical system. The partition profile specifies the name of the partition workload group that it belongs to, if applicable.

Example: Structure of a domain policy:

To gain a better understanding of what a domain policy contains, view this example.

The following table shows the relationship between the components of a sample domain policy.

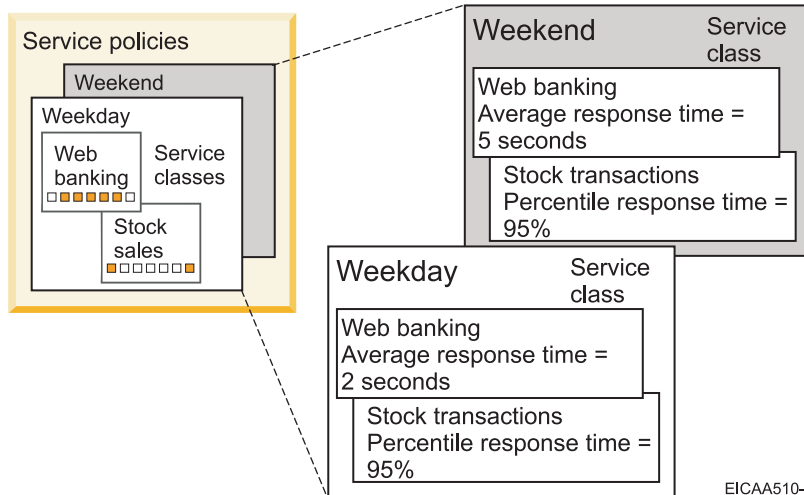
Domain policy				
Service classes				
	Service policy	Service policy	Service policy	
	Weekday	Weekend	Backup	
Web Medium Goal: Percentile response time	90% in 4 seconds Importance: High	Same as weekday service policy.	90% in 8 seconds Importance: Low	
Web High Goal: Percentile response time	90% in 2 seconds Importance: Highest	Same as weekday service policy.	90% in 10 seconds Importance: Low	
Database processes Goal: Percentile response time	90% in 3 seconds Importance: High	Same as weekday service policy.	90% in 9 seconds Importance: Low	
System fixes Goal: Velocity	Slow Importance: Low	Fastest Importance: High	Fastest Importance: High	
AIX partition: Velocity	Fastest Importance: High	Slow Importance: Low	Slow Importance: Low	
Transaction classes				
Application	Rules			Service class
	Filter type	Filter operation	Filter value	
Web serving plug-in (IBM HTTP server)	EWLM: URI	==	/SecFile/accountreps only/appentry/ (*)	Web High
DB2	EWLM: Application instance	==	database1	Database processes
Process classes				
Platform	Rules			Service class
	Filter type	Filter operation	Filter value	
AIX	EWLM: System Name	!=	banksys(*)	AIX partition
i5/OS	Effective Group Profile	==	Install	System fixes
Partition classes				
Platform	Rules			Service class
	Filter type	Filter operation	Filter value	
AIX	EWLM: System Name	==	aix1	AIX partition

Service policies:

| A *service policy* describes business performance objectives and the business importance of work running in an EWLM domain.

| The service policy contains all of the service classes, which contain the performance goals, defined in the domain policy. However, each service policy can prioritize the importance of each service class differently and specify different goals.

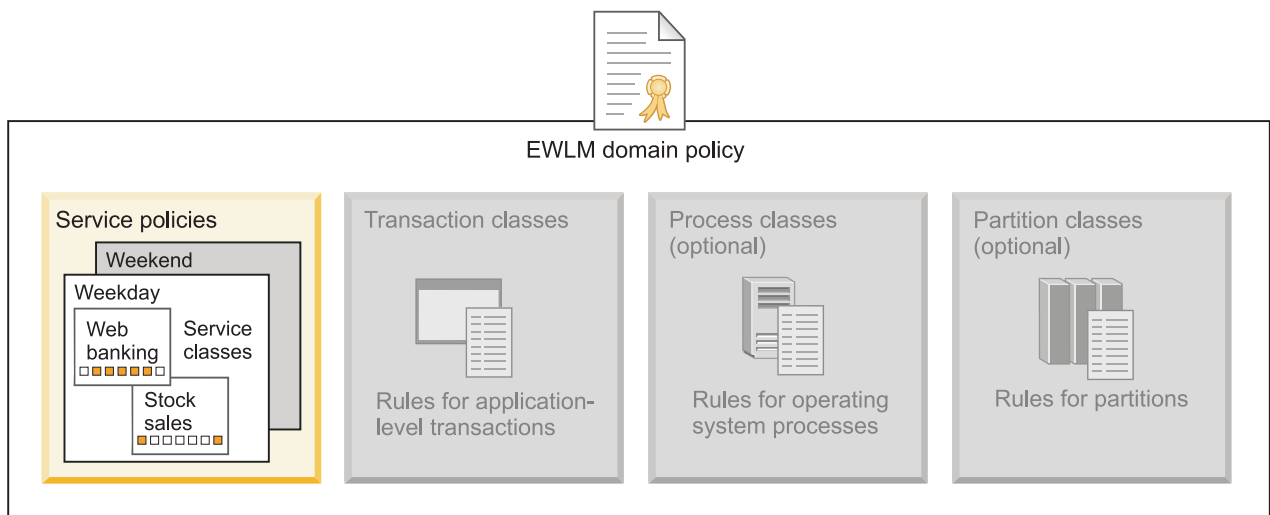
| The following graphic shows the relationship between the components of a service policy. A service policy contains services classes. The service classes specify performance goals.



| The service policy uses a top-down approach, describing the business view of workloads and tying it to the reporting and implicit management of the workloads. The service policy does not contain any explicit linkage to server topology or to the middleware. You can have several policies, that is, several sets of performance objectives intended for different purposes. For example, you might have a service policy for weekday processing and a policy for weekend processing.

| *Service classes:*

| A *service class* contains a performance goal. In addition, it indicates what type of work, transaction, process, or partition, can use the performance goal specified.



EICAA521-1

| Each transaction, partition, and process class that identifies specific work for EWLM to monitor, must also specify which service class to use. For example, the work that EWLM assigns to transaction class ABC is monitored against the performance goal defined in service class XYZ.

| Furthermore, you can create multiple transaction classes that specify the same service class. Likewise, you can create multiple process classes that use the same service class, or you can create multiple partition classes that use the same service class. However, you cannot use the same service class for different types of work. For example, transaction class ABC and process class ABC both cannot specify to use service class XYZ because the classes identify different types of work.

| When creating a service class for the transaction classes, ensure that the default transaction class specifies an appropriate service class goal. The default transaction class includes any application-level transaction that has not been classified. The transactions assigned to this default transaction class use the default service class goal. The same does not hold true for process classes or partition classes because these classes are optional.

| All of the service classes that you define in the domain policy are available to each service policy. Within each service policy, you can change the service class goal. You can define an unlimited number of service classes in a domain policy.

| *Performance goals:*

| A *performance goal* specifies how fast work should run on servers in your business environment.

| EWLM retrieves performance statistics for the work request as soon as it enters the EWLM domain. It continues to receive performance statistics for that work request until it completes. The performance goals are specified in the service classes, which are defined in the service policy. The following performance goals are available:

- | • **Percentile response time:** This goal specifies the percentage of work requests that are to complete within a specified amount of time. For example, 90% of work is to complete within 1 second.
- | • **Average response time:** This goal specifies the average amount of time that is to incur between the time a work request starts and ends. For example, work is to complete within 1 second.
- | • **Velocity:** This goal defines how fast work should run when it is ready, without delays due to processor constraints, storage problems, and I/O delays (for managed system resources). Use a velocity goal for work in which response time goals are not appropriate, such as service processes, daemons, and long-running batch work. You can define a velocity goal as one of the following speeds:
 - | – Fastest
 - | – Fast
 - | – Moderate
 - | – Slow
 - | – Slowest
- | • **Discretionary:** This goal specifies that work is to complete when resources are available. This goal does not require a specific time interval for work to complete nor an importance level. Use this goal for low priority work that does not require a particular performance goal.

| The performance goals that you can use in the service class depend on what type of work the service class is intended for.

| The following table describes what performance goals are available for the different types of work that EWLM can monitor.

Table 4. The performance goals available to each type of work that EWLM can monitor

Type of work	Component of domain policy that identifies the work for EWLM to monitor	Possible goal types
Transactions	Transaction class	<ul style="list-style-type: none"> • Average response time • Percentile response time • Velocity • Discretionary
Processes	Process class	<ul style="list-style-type: none"> • Average response time • Percentile response time • Velocity • Discretionary
Partitions (includes both transactions and processes)	Partition class	<ul style="list-style-type: none"> • Velocity • Discretionary

Workloads:

A *workload* is a group of service classes. This allows you to view the overall performance for a set of service classes.

For example, if you want to view the overall performance for all of your WebSphere Application Server transactions, you can group all of the service classes that these transactions use into one workload. Then, you can determine if these application-level transactions, as a whole, are meeting the performance objectives. Similarly, if you have a set of performance goals for a particular business partner, you can group them into a workload to determine if the work for that business partner is meeting the performance objectives.

Applications:

Applications, within EWLM, refer to the applications such as WebSphere Application Server, HTTP servers, and so on, that reside on the systems in your EWLM domain. In addition, these are the applications whose transactions EWLM is to monitor.

EWLM can monitor multiple applications. When monitoring an application, the level of performance data that EWLM collects for the application-level transactions depends on whether the application instruments the Application Response Measurement (ARM) 4.0 standard.

If the application uses the ARM 4.0 standard, then EWLM is able to obtain detailed transaction data for the application. EWLM can track the application-level transactions as they flow from one application to another, which is referred to as a hop. If the application is an *entry application*, which is the first ARM-instrumented application in the domain that receives the transaction, you must add the application to the domain policy and create a transaction class for the application.

If the application does not use the ARM 4.0 standard, EWLM can monitor the transaction time from start to end, but it does not obtain the granular performance data for each hop. Also, you must identify these transactions by defining a partition class rather than a transaction class.

Each application that EWLM is to monitor must be set up in the EWLM Control Center. To view a predefined list of applications set up, log in to the EWLM Control Center and select **Set up** → **Applications**. Then, the application needs to be added to a domain policy.

Related topics

“Application instrumentation” on page 20

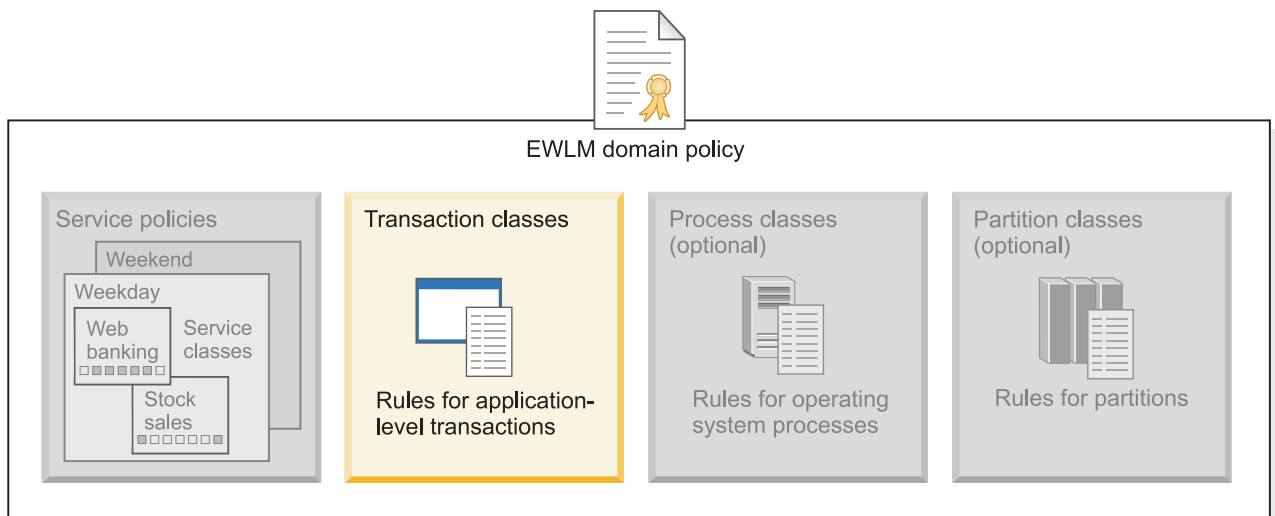
Instrumenting an application with the appropriate APIs allows EWLM to properly monitor the application-level transactions.

“Enabling ARM on IBM-instrumented applications” on page 205

Use this topic to learn how to enable ARM on supported middleware applications.

Transaction classes:

A *transaction class* contains classification rules that identify application-level transactions. Each transaction class also specifies a service class, which defines the performance goal, that the transactions performance is to meet. If a transaction’s attributes meet the conditions of the classification rules, then EWLM assigns the transaction to the class.



EICAA522-1

When creating transaction classes, consider the following usage notes:

- Create transaction classes for each potential entry application in your EWLM domain.
- EWLM only classifies a transaction to one transaction class.
- When assigning a transaction to a transaction class, EWLM uses the transaction classes in ascending order.
- Define specific transaction classes in low-numbered positions.
- Define general transaction classes in high-numbered positions.
- If a transaction does not meet the classification rules of any transaction class defined for the application, EWLM assigns it to the default transaction class in the highest numbered position that specifies a wildcard (*) value.

A transaction class is similar to a process class or partition class in it associates EWLM-monitored activity to service class objectives. However, a transaction class includes transactions that are reported through applications that instrument the ARM 4.0 standard, whereas, a process class includes processes as perceived by the operating system and a partition class includes all work that a partition processes.

To identify work in a transaction class, use a filter in a classification rule. For more information, see “Filter” on page 35.

Platforms:

Within EWLM, a *platform* is software that interfaces with the system's hardware. A more specific term is an operating system. EWLM uses the term platform to generically refer to the operating systems that EWLM can monitor. The EWLM Control Center uses the term platform in many field names.

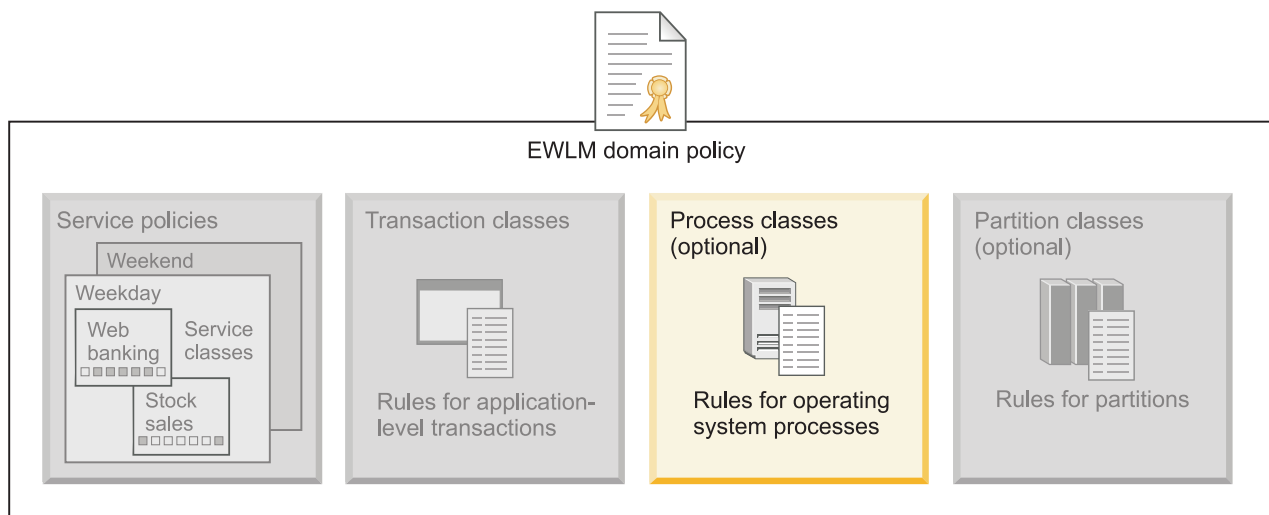
Each platform for which EWLM can monitor its operating system processes is predefined in the EWLM Control Center. These platforms are listed in the EWLM Control Center in **Set up** → **Platforms**. Only the platforms that support EWLM monitoring of their operating system processes are listed. This is not a list of operating systems that support the managed server function, but rather a list of platforms whose operating system processes can be monitored.

For EWLM to monitor a platform's operating system processes, EWLM uses process classes. These are defined in the domain policy. If a platform does not support process classes, it is not specified in the EWLM Control Center list of platforms.

Your EWLM domain can use any combination of the supported operating systems. For a list of supported operating systems, see Specified operating environments.

Process classes:

A *process class* contains classification rules that identify a group of processes that run on a specific operating system. Each process class also specifies a service class performance goal that the operating system processes are to adhere to. If an operating system process meets the conditions of the classification rules, then EWLM assigns the process to the class.



EICAA523-1

When creating process classes, consider the following usage notes:

- Create process classes for each platform whose operating system processes you want EWLM to monitor.
- EWLM only classifies a process to one process class.
- EWLM may also classify the process to a partition class, if applicable.
- When assigning a process to a process class, EWLM uses the process classes in ascending order.
- Define specific process classes in low-numbered positions.
- Define general process classes in high-numbered positions.
- Process classes are optional. Therefore, if a process does not meet the conditions of any of the classification rules, EWLM does not classify it to a process class.

A process class is similar to a transaction class or partition class in that it associates EWLM-monitored activity to service class objectives. However, a transaction class includes transactions that are reported through applications that instrument the ARM 4.0 standard, whereas, a process class includes processes as perceived by the operating system, and a partition class includes all work processed by a partition.

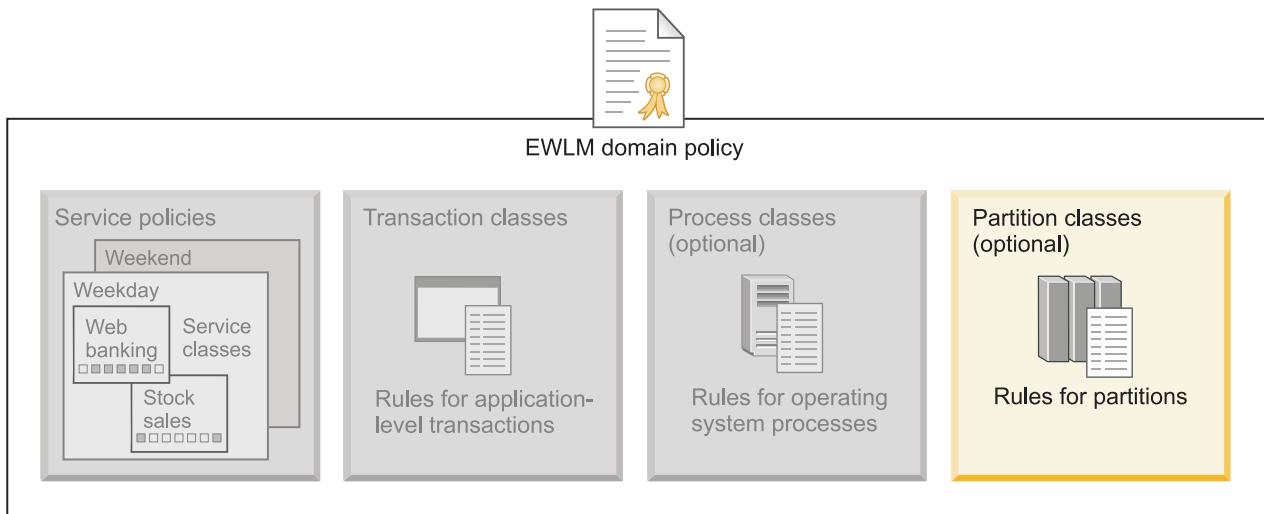
To identify work in a process class, use a filter in a classification rule. For more information, see “Filter” on page 35.

Partition classes:

A *partition class* identifies partitions that are to be managed with a single goal and business importance. The partition class is associated with a service class that contains the goal and business importance. You use partition classes as part of EWLM’s partition workload group management. EWLM manages the CPU resources based on the overall performance of partitions in a partition class.

A partition class includes all work that the partition processes regardless of the application instrumenting the ARM 4.0 standard APIs. However, note that EWLM does not collect performance data that is as detailed for application-level transactions that the partition process. To obtain more detailed performance data for application-level transactions, you must use a transaction class.

Note: If EWLM classifies work to a partition class, EWLM can also classify the work to either a transaction or process class.



EICAA529-0

A partition class allows you to monitor the work that a partition processes as a whole. In turn, you are not required to define specific classification rules to identify operating system processes nor application-level transactions.

When creating partition classes, consider the following usage notes:

- A partition class includes all work that a partition processes, which includes both application-level transactions and operating systems processes.
- A partition class includes application-level transactions for applications that do not use the ARM 4.0 standard APIs.
- EWLM only classifies partition work to one partition class.
- When assigning work to a partition class, EWLM uses the partition classes in ascending order.
- Define specific partition classes in low-numbered positions.
- Define general partition classes in high-numbered positions.

- Partition classes are optional. Therefore, if work does not meet the conditions of any of the classification rules, EWLM does not monitor the work in a partition class.
- The performance goal for a partition class must specify a velocity or discretionary goal because EWLM collects data for the partition as a whole.
- Work that EWLM assigns to a partition class can also be assigned to either a transaction or process class.
- EWLM monitors all work for a partition against one, single performance goal. If you want to monitor the work separately, you must also use transaction and process classes to define different performance goals.

A partition class is similar to a transaction class or process class in that it associates EWLM-monitored activity to service class objectives. However, a transaction class includes transactions that are reported through applications that instrument ARM standards, whereas, a process class includes processes as perceived by the operating system, and a partition class includes all work processed by a partition.

Real-time reporting

Real-time reporting is the performance data that EWLM periodically updates in the EWLM Control Center.

EWLM constantly updates this data because each managed server is periodically sending performance statistics to the domain manager. The domain manager consolidates the data from all of the managed servers in the EWLM domain and displays the data to you.

Use the Monitor section of the EWLM Control Center to view this real-time performance data.

When monitoring the performance of the work requests in your EWLM domain, you can view the following types of information:

Details report

A details report contains information that is specific to the item that you are viewing. For example, if you view a service class details report, you can view how many transactions are classified to the service class. In addition, you can view how many transactions failed versus completed successfully. You can also view a details report that is specific to a transaction class, process class, or managed server.

Monitor

A *monitor* is a graph that contains performance data that EWLM updates every 10 seconds. The specific data that the Monitor displays depends on which monitor you are viewing. The following is a list of monitors that you can view:

- **Processor utilization:** Allows you to view how much of the managed server's processor resources are used. The amount specified includes work requests monitored by EWLM and any other work requests that are being processed on the server.
- **Performance index:** Allows you to view when a service class exceeds its service class goal and when it does not meet the goal. The performance index (PI) is available only to service classes that specify average response time, percentile response time, or velocity goal. To determine the PI, EWLM divides the actual performance of the service class by the goal that is specified in the service class.
- **Goal achievement:** Allows you to view if the goal defined is met during specific time intervals. You can view the actual performance of work that EWLM monitors relative to the goal and how the performance fluctuates over time.
- **Transaction rate:** Allows you to view the number of transactions that complete per second. The transaction rate includes the number of transactions completed successfully or unsuccessfully.

Topology

A *topology* is a diagram that shows the end-to-end flow of applications or servers that process a work request. You can view an application topology to determine what applications are

processing work for a service class. In addition, you can view a server topology to determine what managed servers are processing work for a service class. Use the topologies to learn about the end-to-end flow of work requests processed in the EWLM domain.

For more information about how to use these monitor views, see “Monitoring performance data with EWLM” on page 283

Related concepts

“Data collection and aggregation” on page 23

A valuable capability of EWLM is its ability to collect and aggregate performance data. This performance data collection capability allows EWLM to report to you if performance goals are met. In addition, it allows EWLM to make automatic resource adjustments if goals are not met.

Performance objectives

Performance objectives are the high-level goals for the workloads that run in your IT environment. These performance objectives are translated into performance goals that you define in an EWLM domain policy service class.

In your business environment, you have specific goals regarding how quickly the work requests complete. These goals might be specific to how your own business’s work requests complete or might be goals that you have specified in Service Level Agreements (SLA) with business partners. The main objective for your business is to ensure your own business goals are met as well as goals for your business partners.

To ensure that these goals are met, you can use service class goals within your EWLM environment. You can create a service class goal that corresponds directly to an SLA or business goal.

For example, imagine that you have an SLA with a business partner that guarantees that when they access your Web application, the response time for their transactions to complete is less than two seconds. You can create a transaction class to identify the work requests that the business partner initiates. The transaction class contains rules that identify the business partner’s user names. Then, when a user submits a work request to the Web application, EWLM can identify the work requests initiated by user names associated with this business partner. You can use multiple rules to identify the work requests more specifically. After a work request matches a rule in the transaction class, EWLM classifies it to the associated service class. In this example, the transaction class for this business partner uses a service class with an associated service policy that specifies an average response time of two seconds. Then, you can view performance data for this service class to determine whether the business partner’s performance goal is met.

There are multiple ways that you can define a domain policy to meet the needs of your business environment. Within one service policy, you can set up a service class for each business partner. Also, you can set up multiple service classes if you have different performance goals for different types of work requests. Therefore, you can have a group of service classes that apply to one business partner. You can group these service classes into a workload so you can view performance data about the entire group of service classes. If you have different goals for different times of the day or days of the week, you can activate different service policies. Then, the weekday service policy contains service class goals for the weekday and the weekend service policy contains goals for the weekend.

To learn more about what goals you can specify in a transaction or process class, see “Performance goals” on page 46. To learn more about how to identify a work request in a transaction or process class rule, see “Filter” on page 35. The filters topic identifies the filter types that you can use when creating a rule in a process class or transaction class. Use the filter types to identify work requests.

Related task

Monitoring performance data with EWLM

Scenarios

Use can use these scenarios to learn about various ways to use EWLM.

Scenario: Classifying and monitoring transactions processed by the WebSphere Application Server

Use this scenario to learn how to create classification rules in a domain policy that identifies and monitors the work that the WebSphere Application Server processes. Similar techniques can be used to monitor work processed by other applications.

Situation

The company MyCo has signed Service Level Agreements with several partner companies that require certain Web-based transactions to be completed within a specified amount of time. You are the performance analyst for MyCo, and you must ensure that these transactions, in this case WebSphere Application Server transactions, meet their stated performance goals.

Objectives

As MyCo's performance analyst, you must ensure that your Web transactions meet the following performance goals:

1. To ensure that 90% of Web transactions initiated by business partner Company A have an average response time of 1 second as defined in the Service Level Agreement with Company A.
2. To ensure that the Web transactions initiated by business partner Company B have an average response time of 1 second as defined in the Service Level Agreement with Company B.
3. To ensure that all other Web transactions complete within an average response time of 2 seconds.

Details

To meet these objectives, you must define a domain policy that describes these performance requirements. The domain policy will have the following major elements:

Workload

This will define the WebSphere Application Server processes that EWLM will monitor in this domain.

Service classes

The MyCo administrator will create service classes to define the performance requirements for the transactions from each company.

Applications

EWLM includes filters for some specific applications. In this case, the WebSphere Application Server filters enable performance analysts to define the rules used to associate work with a service class by defining a transaction class.

Transaction class

This identifies the transactions coming in from company A and company B so that they can be monitored separately.

Prerequisites and assumptions

This scenario assumes the following:

- The EWLM domain manager is installed and configured.
- The EWLM managed server is installed and configured on the system that runs the WebSphere Application Server.
- The EWLM domain manager is started.
- The EWLM managed server is started.

- The EWL Control Center is started.

Configuration details

To ensure that the Service Level Agreements with MyCo's business partners are being met, the MyCo performance analyst will complete the following steps:

Create a new domain policy, workload and service classes:

First, the MyCo. performance analyst must create a new domain policy with the WebSphere Application Server workload, and create service classes that define the performance requirements for company A and B's transactions.

1. Log on to the EWL Control Center.
2. Click **Set up** → **Domain policies** → **New**.
3. Complete the Welcome page as indicated to do so.
4. Click **Next**
5. On the **Workload** page, create a new workload called **WebSphere Application Server**.
6. Select the new workload, and click **Next**.
7. On the **Service classes** page, click **New**.
8. On the **New service class** page, specify Company A for the **Name**.
9. Select **Transaction** for the **Type**.
10. Select **WebSphere Application Server** for the **Workload**.
11. Select **Percentile response time** for the **Goal type**. After you select this option, more fields display. Selecting a goal type of percentile response time will allow you to track whether some percentage of your transactions are meeting performance goals. If a few transactions have abnormally long response times, your goal will still be met, since the majority of the transactions completed on time. In contrast, a goal type of average response time would be affected by all transactions; a single very long running transaction out of a set of hundreds could significantly change the average response time.
12. Select **Highest** for the **Importance**. For goal based workload management its important to prioritize different types of work. In this case, indicating that this work has the highest important means that if resources become constrained, they may be taken from other sources so that this workload will continue to meet its goals.
13. Specify 90 for the **Percent**. This indicates the percentage of the total transactions that must complete in order to meet the performance goals.
14. Specify 00 Hours 00 Minutes 01 Seconds 000 Milliseconds for the **Time**.
15. Click **OK** to create the service class.
16. On the Service Classes page, click **New**.
17. On the **New service class** page, specify Company B for the **Name**.
18. Select **Transaction** for the **Type**.
19. Select **WebSphere Application Server** for the **Workload**.
20. Select **Average response time** for the **Goal type**. After you select this option, more fields display.
21. Select **Highest** for the **Importance**.
22. Specify 00 Hours 00 Minutes 01 Seconds 000 Milliseconds for the **Time**.
23. Click **OK** to create the service class.
24. On the Service Classes page, click **New**.
25. On the **New service class** page, specify Other WebSphere Application Server transactions for the **Name**.
26. Select **Transaction** for the **Type**.

- | 27. Select **WebSphere Application Server** for the **Workload**.
- | 28. Select **Average response time** for the **Goal type**. After you select this option, more fields display.
- | 29. Select **Highest** for the **Importance**.
- | 30. Specify 00 Hours 00 Minutes 02 Seconds 000 Milliseconds for the **Time**.
- | 31. Click **OK** to create the service class.
- | 32. Click **Next**

| **Create a domain policy, applications and transaction classes:**

| Add the WebSphere Application Server as the application to be monitored in the domain, and define transaction classes for both company A and B.

- | 1. On the **Applications** page, click **Add**.
- | 2. Select **WebSphere** for the **Name**.
- | 3. Click **OK**.
- | 4. Click **Next**.
- | 5. Select **WebSphere** in the **Select an application** field. EWLM will classify transactions based on the first application involved as it enters the systems. This is called the Entry point application.
- | 6. Click **New** to create a transaction class to identify the work that Company A initiates.
- | 7. On the **New Transaction Class** page, specify **CompanyAwork** for the **Name**.
- | 8. Click **New** to create a new rule.
- | 9. Select **EWLM:URI** in the **Select filter type** field. Choosing URI as a filter type causes all transactions associated with the specified URI to meet the rule's criteria and thus be classified with this transaction class.
- | 10. Select **==** for the operation.
- | 11. Specify **/SecFile/CompanyAreps/appentry/(*)** for the rule value.
- | 12. Click **OK** to create the rule.
- | 13. Click **OK** to create the transaction class. EWLM now assigns work that meets the conditions of this rule to this transaction class.
- | 14. Click **New** to create a transaction class to identify the work that Company B initiates.
- | 15. On the **New Transaction Class** page, specify **CompanyBwork** for the **Name**.
- | 16. Click **New** to create a new rule.
- | 17. Select **EWLM:URI** in the **Select filter type** field.
- | 18. Select **==** for the operation.
- | 19. Specify **/SecFile/CompanyBreps/appentry/(*)** for the rule value.
- | 20. Click **OK** to create the rule.
- | 21. Click **OK** to create the transaction class. EWLM now assigns work that meets the conditions of this rule to this transaction class.
- | 22. Click **Next**. You do not need to add any platforms because you are only monitoring the application-level work.
- | 23. Click **No** when an error is displayed indicating that you cannot create a process class without first adding a platform.
- | 24. Click **No** when an error is displayed indicating that you cannot create a partition class without first adding a platform.
- | 25. On the **Service policy** page, click **Finish**. This saves the domain policy

| **Deploy the domain policy and activate the service policy:**

| After defining the domain policy, you must deploy and activate it so that EWLM can start collecting data.

1. Go to **Set up** → **Domain policies**.
2. Select your domain policy so that it is highlighted.
3. Select **Deploy** from the **Select Action** field and click **Go**.
4. On the **Deploy domain policy** page, select **EWLM Service Policy** in the **Service policy to activate** screen. Or, select the appropriate service policy name depending on what you named it. The previous procedures indicated that you could use the default service policy, which is called **EWLM Service Policy**.
5. Click **Deploy**.

Determine if performance goals are met:

After you deploy the domain policy and activate the service policy, EWLM collects performance data. You can then view reports to see if the performance requirements are being met.

Use the Monitor section of the EWLM Control Center to determine if the transactions processed by the WebSphere Application Server are adhering to the performance goals that you defined. To do this, view the following reports:

- **Service classes:** View the performance index for the **Company A** service class. If the performance index is greater than one, it means that the goal being defined in the policy is being met. If the performance index is less than one, the goal is not being met. A performance index of none means that no transactions are currently associated with the service class.
- **Service classes:** View the performance index for the **Company B** service class.
- **Transaction classes:** View the **Company A work** transaction class to determine the number of transactions processed and other transaction-related data. You can go into the details screen to see more information about the transaction, including resource utilization.
- **Transaction classes:** View the **Company B work** transaction class to determine the number of transactions processed and other transaction-related data.
- **Transaction classes:** View the **Default - WebSphere** transaction class to view the performance of WebSphere Application Server transactions that are not associated with either the **Company A work** or **Company B work** transaction classes.
- **Managed Server:** View each class of work's contribution to the CPU load on the managed server.
- Use any other reports and monitors as necessary to troubleshoot performance problems, if they exist.

Scenario: Classifying and monitoring operating system processes

Use this scenario to learn how to create classification rules in a domain policy that identifies and monitors operating system processes. This specific scenario describes how to identify and monitor i5/OS and AIX operating system processes; however, similar techniques can be applied to monitor other operating system processes.

Situation

The performance analyst for MyCo must ensure that high priority workloads are getting the system resources to complete quickly, even if that performance comes at the expense of lower priority work.

MyCo has a partitioned system that is hosting an AIX system that provides interactive reports to their accounting department. The system also hosts an i5/OS partition that the development team uses to run nightly builds. Because the request for reports must be completed interactively, it should be given priority over the development work. The MyCo performance analyst would like to create a domain policy that measures performance data for these two workloads, and assigns the interactive work a higher priority.

| Objectives

| You have the following objectives:

- | 1. To classify AIX operating system processes that handle the accounting department's reporting software as a high velocity performance goal.
- | 2. To classify i5/OS operating system processes that handle the development team's test builds to a Low velocity performance goal.
- | 3. To monitor the AIX operating system processes to determine if the performance goal is met.
- | 4. To monitor the i5/OS operating system processes to determine if the performance goal is met.

| Details

| To meet these objectives, you must define a domain policy that describes these performance requirements. The domain policy will have the following major elements:

| Workload

| This will define the work for the accounting and development teams.

| Service classes

| The MyCo administrator will create service classes to define the performance requirements for the work handled by AIX and i5/OS system processes.

| Platforms

| The EWLM domain policy contains platform-specific filters that are used to define the rules for monitoring operating system processes. Once the AIX and i5/OS platforms have been added, the required filters are available.

| Process classes

| The process classes identify which types of operating system processes should be associated with a given service class.

| Prerequisites and assumptions

| This scenario assumes the following:

- | • The EWLM domain is manager installed and configured.
- | • The EWLM managed server is installed and configured on the system that runs the WebSphere Application Server.
- | • The EWLM domain manager is started.
- | • The EWLM managed server is started.
- | • The EWLM Control Center is started.

| Configuration details

| The MyCo performance analyst can define an EWLM domain policy to ensure that the accounting reports being processed on an AIX system are being given priority over the development builds being run on the i5/OS system. The MyCo performance analyst can complete the following steps:

| Create a new domain policy, workload and service classes:

| The MyCo. performance analyst must create a new domain policy with the accounting workload, and create service classes that define the different performance requirements for accounting reports and the development work on i5/OS.

- | 1. Log on to the EWLM Control Center.
- | 2. Click **Set up** → **Domain policies** → **New**.
- | 3. Complete the **Welcome** page.

- | 4. Click **Next**.
- | 5. On the **Workload** page, create a new workload called **Accounting workload**.
- | 6. Click **Next**.
- | 7. On the **Service classes** page, click **New**.
- | 8. On the **New service class** page, specify Accounting reports for the **Name**.
- | 9. Select **Process** for the **Type**.
- | 10. Select **Accounting** for the **Workload**.
- | 11. Select **Velocity** for the **Goal type**. After you select this option, more fields display. A goal type of velocity measures how quickly the work should run relative to other work. This is particularly useful for long-running workloads, where a response time would be difficult to specify. The following settings indicate that this work should complete faster than other work on the system.
- | 12. Select **Highest** for the **Importance**.
- | 13. Select **Fastest** for the **Level**.
- | 14. Click **OK** to create the service class.
- | 15. On the **Service classes** page, click **New**.
- | 16. On the **New service class** page, specify Development builds for the **Name**.
- | 17. Select **Process** for the **Type**.
- | 18. Select **Accounting** for the **Workload**.
- | 19. Select **Velocity** for the **Goal type**. After you select this option, more fields display.
- | 20. Select **Lowest** for the **Importance**.
- | 21. Select **Slowest** for the **Level**.
- | 22. Click **OK** to create the service class.
- | 23. Click **Next**.

| **Create a domain policy, platforms and process classes:**

| Add the platforms and the process classes that will identify which system processes should be monitored.

- | 1. Click **Next** twice to skip the **Applications** and **Transaction classes** page.
- | 2. On the **Platforms** page, click **Add**.
- | 3. In the **Name** box, select **OS/400**.
- | 4. Click **OK** to add the filters for i5/OS to the domain policy and return to the **Platforms** page.
- | 5. On the **Platforms** page, click **Add**.
- | 6. In the **Name** box, select **AIX**.
- | 7. Click **OK** to add the AIX filters to the domain policy and return to the **Platforms** page.
- | 8. Click **Next** to proceed to the **Process classes** page.
- | 9. On the **Process classes** page, select **AIX** in the **Select a server platform** page, and click **New**.
- | 10. In the **Name** field, enter **AIX reports**.
- | 11. In the **Service class** box, select **Accounting reports**.
- | 12. Click **New** in the rules area to define a new process rule. The following steps will define criteria that will be used to associate system processes with the service class defined in a previous procedure. In this example, the process rule will qualify all processes on the AIX system "mycoAIX01" started by the Accounting user group.
- | 13. Click the **Select filter type** box and select **EWLM: System Name**.
- | 14. Select **= =** for the operation.
- | 15. Specify **mycoAIX01** for the rule value.
- | 16. Click the **AND** button and drag it to the line extending from the rule you just defined. A new box is displayed.

- | 17. Click the **Select filter type** box and select **GroupName**.
- | 18. Select **= =** for the operation.
- | 19. Specify **Accounting** for the rule value.
- | 20. Click **OK** to create the rule.
- | 21. Click **OK** to create the process class. This process class now associates all work done by the Accounting group on the system named mycoAIX01 with the performance requirements specified in the Accounting reports service class.
- | 22. On the **Process classes** page, select **OS/400** in the **Select a server platform** page, and click **New**.
- | 23. In the **Name** field, enter **i5OS development work**.
- | 24. In the **Service class** box, select **Development builds**
- | 25. Click **New** in the rules area to define a new process rule. The following steps will define criteria that will be used to associate system processes with the service class defined in a previous procedure. In this example, the process rule will qualify all processes on the i5/OS system "mycoOS400x01" started by the Development user group.
- | 26. Click the **Select filter type** box and select **EWLM: System Name**.
- | 27. Select **= =** for the operation.
- | 28. Specify **mycoOS400x01** for the rule value.
- | 29. Click the button and drag it to the line extending from the rule you just defined. A new box is displayed.
- | 30. Click the **Select filter type** box and select **GroupName**.
- | 31. Select **= =** for the operation.
- | 32. Specify **Development** for the rule value.
- | 33. Click **OK** to create the rule.
- | 34. Click **OK** to create the process class. This process class now associates all work done by the Development group on the system named mycoOS400x01 with the performance requirements specified in the Batch jobs reports service class.
- | 35. Click **Next** twice to skip the **Partition classes** page and proceed to the **Service policy** page.
- | 36. On the **Service policy** page, click **Finish** to save the domain policy.

| **Deploy the domain policy and activate the service policy:**

| After defining the domain policy, you will need to deploy and activate it so that EWLM will start collecting data.

- | 1. Go to **Set up** → **Domain policies**.
- | 2. Select your domain policy so that it is highlighted.
- | 3. Select **Deploy** from the **Select Action** field and click **Go**.
- | 4. On the **Deploy domain policy** page, select **EWLM Service Policy** in the **Service policy to activate** screen. Or, select the appropriate service policy name depending on what you named it. The previous procedures indicated that you could use the default service policy, which is called **EWLM Service Policy**.
- | 5. Click **Deploy**.

| **Determine if performance goals are met:**

| After you deploy the domain policy and activate the service policy, EWLM collects performance data. You can then view reports to see if the performance requirements are being met.

| Use the Monitor section of the EWLM Control Center to determine if the operating system processes from AIX and OS/400 are adhering to the performance goals that you defined. To do this, view the following reports:

- | • **Service classes:** View the performance index for the **Accounting reports** service class.
- | If the performance index is greater than one, it means that the goal being defined in the policy is being met. If the performance index is less than one, the goal is not being met. A performance index of none means that no transactions are currently associated with the service class.
- | • **Service classes:** View the performance index for the **Development builds** service class.
- | • **Process classes:** View the **AIX reports** process class to view performance data for the related AIX processes.
- | • **Process classes:** View the **i5/OS development work** process class to view performance data for the related i5/OS jobs.
- | Looks into details to find out how much resource is being consumed by this type of work.
- | • **Managed Server:** View each class of work's contribution to the CPU load on the managed server.
- | • Use any other reports and monitors as necessary to troubleshoot performance problems, if they exist.

| **Scenario: Classifying and monitoring i5/OS operating system processes separately from WebSphere transactions**

| Use this scenario to learn how to create classification rules in a domain policy that identify and monitor all operating system processes for i5/OS, a specific i5/OS print process, and WebSphere Application Server transactions separately. This scenario illustrates how you can define a performance goal for operating system processes and a different performance goal for application-level work. This scenario uses a specific operating system and application; however, similar techniques can be applied to other operating system and application environments.

| **Situation**

- | The performance analyst for MyCo monitors the work that is processed on the i5/OS operating system. The IT environment that includes the following:
- | • One system that runs the i5/OS operating system which also includes the following:
 - | – WebSphere Application Server
 - | – Non-ARM instrumented applications
 - | – Operating system processes for i5/OS
 - | – Print processes in i5/OS

| **Objectives**

- | You have the following objectives:
- | 1. To classify i5/OS operating system processes to a Medium velocity performance goal.
 - | 2. To classify i5/OS print process as a Highest velocity performance goal.
 - | 3. To classify the transactions that the WebSphere Application Server processes to a percentile response time performance goal.
 - | 4. To monitor the i5/OS print process to determine if the performance goal is met.
 - | 5. To monitor the i5/OS operating system processes to determine if the performance goal is met.
 - | 6. To monitor the WebSphere Application Server transactions to determine if the performance goal is met.
 - | 7. To adjust the system resources if the performance goals are not met.

| **Details**

| To do this, you must define a domain policy that describes these performance requirements. The domain policy will have the following major elements:

| **Workload**

| This will define the work for the i5/OS server.

| **Service classes**

| The MyCo administrator will create service classes to define the performance requirements for the WebSphere Application Server transactions, and for the i5/OS processes.

| **Applications**

| EWLM includes filters for some specific applications. In this case, selecting WebSphere Application Server enables the performance analyst to define the rules used to associate work with a service class by defining a transaction class.

| **Transaction class**

| This identifies the type of WebSphere Application Server work that should be monitored and compared to the performance requirements defined in the associated service class.

| **Platforms**

| The EWLM domain policy contains platform-specific filters that are used to define the rules for monitoring operating system processes. Once the i5/OS platform is added, the required filters are available.

| **Process classes**

| The process classes identify which types of operating system processes should be associated with a given service class. In this scenario, two process classes are required, one to classify all i5/OS process classes, and one to classify the print process.

| **Prerequisites and assumptions**

| This scenario assumes the following:

- | • The EWLM domain manager is installed and configured.
- | • The EWLM managed server is installed and configured on the system that runs the WebSphere Application Server.
- | • The EWLM domain manager is started.
- | • The EWLM managed server is started.
- | • The EWLM Control Center is started.

| **Configuration details**

| The MyCo performance analyst can define an EWLM domain policy to see how three different types of workloads are using system resources. The following procedures will define a domain policy that will separate performance information for each workload.

| **Create a new domain policy, workload and service classes:**

| The MyCo performance analyst must create a new domain policy with a workload representing both the WebSphere Application Server transactions and the i5/OS processes. The MyCo performance analyst also needs to create a service class for the two different types of work.

- | 1. Log on to the EWLM Control Center.
- | 2. Click **Set up** → **Domain policies** → **New**.
- | 3. Complete the **Welcome** page as indicated to do so.
- | 4. Click **Next**.
- | 5. On the **Workload** page, create a new workload called **i5OS workload**.
- | 6. Click **Next**.
- | 7. On the **Service classes** page, click **New**.
- | 8. On the **New service class** page, specify WAS transactions for the **Name**.
- | 9. Select **Transaction** for the **Type**.
- | 10. Select **i5OS workload** for the **Workload**.

11. Select **Percentile response time** for the **Goal type**. After you select this option, more fields display. Selecting a goal type of percentile response time will allow you to track whether some percentage of your transactions are meeting performance goals. If a few transactions have abnormally long response times, your goal will still be met, since the majority of the transactions completed on time. In contrast, a goal type of average response time would be affected by all transactions; a single very long running transaction out of a set of hundreds could significantly change the average response time.
12. Select **Highest** for the **Importance**.
13. Specify 90 for the **Percent**.
14. Specify 00 Hours 00 Minutes 01 Seconds 000 Milliseconds for the **Time**.
15. Click **OK** to create the service class.
16. On the **Service classes** page, click **New**.
17. On the **New service class** page, specify i5OS processes for the **Name**.
18. Select **Process** for the **Type**.
19. Select **i5OS workload** for the **Workload**.
20. Select **Velocity** for the **Goal type**. After you select this option, more fields display. A goal type of velocity measures how quickly the work should run relative to other work. This is particularly useful for long-running workloads, where a response time would be difficult to specify. The following settings indicate that this work should complete faster than other work on the system.
21. Select **Medium** for the **Importance**.
22. Select **Moderate** for the **Level**.
23. Click **OK** to create the service class.
24. On the **Service classes** page, click **New**.
25. On the **New service class** page, specify i5OS print processes for the **Name**.
26. Select **Process** for the **Type**.
27. Select **i5/OS workload** for the **Workload**.
28. Select **Velocity** for the **Goal type**. After you select this option, more fields display. A goal type of velocity measures how quickly the work should run relative to other work. This is particularly useful for long-running workloads, where a response time would be difficult to specify. The following settings indicate that this work should complete faster than other work on the system.
29. Select **High** for the **Importance**.
30. Select **Fast** for the **Level**.
31. Click **OK** to create the service class.
32. Click **Next**.

Create a domain policy, applications and transaction classes:

- Add the WebSphere Application Server application to be monitored in the domain, and define a transaction class that will define which work to associate with the performance requirements.
1. On the **Applications** page, click **Add**. EWLM will classify transactions based on the first application involved as it enters the systems. This is called the Entry point application.
 2. Select **WebSphere** for the **Name**.
 3. Click **OK**.
 4. Click **Next**.
 5. Select **WebSphere** in the **Select an application** field.
 6. On the Transaction class window, select the default WebSphere Application Server transaction class.
 7. Select **Edit** from task list, and click **Go**.
 8. Select **WAS transactions** in the **Service class** box. The default transaction class for WebSphere Application Server contains a rule that filters all work associated with the application. Once the

- | MyCo performance analyst identifies the **WAS transactions** service class he or she can monitor this work to see whether it is meeting the performance requirements set in that service class.
- | 9. Click **OK** to save the changes to the transaction class.
- | 10. Click **Next** to proceed to the **Platforms** page.

| **Create a domain policy, platforms and process classes:**

| Next, add the platforms and the process classes that will identify which system processes should be monitored.

- | 1. On the **Platforms** page, click **Add**.
- | 2. In the **Name** box, select **OS/400**. EWLM includes several platform specific filters that you can use to categorize work. Adding OS/400 as a platform will make these filters available when you define a process class in the following steps.
- | 3. Click **OK** to add the filters for OS/400 to the domain policy and return to the **Platforms** page.
- | 4. Click **Next** to proceed to the **Process classes** page.
- | 5. On the **Process classes** page, select **OS/400** in the **Select a server platform** page, and click **New**.
- | 6. In the **Name** field, enter **OS400 print processes**.
- | 7. In the **Position** box, ensure that 1 is listed. If a process meets the criteria specified in more than one process class rules, it will only be classified by the process class with the lowest number position. In this example we will define process classes that will classify print processes and all i5/OS processes. Since the print process would meet the criteria specified by both process class rules (a print process is also an operating system process) we will need to ensure that the print process rule has lower position number (it will be listed first) than the process class for all i5/OS processes.
- | 8. In the **Service class** box, select **i5OS print processes**
- | 9. Click **New** in the rules area to define a new process rule. The following steps will define criteria that will be used to associate system processes with the service class defined in a previous procedure. In this example, the process rule will qualify the i5/OS print processes.
- | 10. Click the **Select filter type** box and select **Job Name**.
- | 11. Select **= =** for the operation.
- | 12. Specify **QPRTJOB** for the rule value.
- | 13. Click **OK** to create the rule.
- | 14. Click **OK** to create the process class. This process class will now qualify all print jobs, and associate them with the OS/400 print processes service class.
- | 15. On the **Process classes** page, select **OS/400** in the **Select a server platform** page, and click **New**.
- | 16. In the **Name** field, enter **OS400 processes**.
- | 17. In the **Position** field, enter 2. If a process meets the criteria specified in more than one process class rules, it will only be classified by the process class with the lowest number position. In this example we have defined process classes that will classify print processes and all i5/OS processes. Since the print process would meet the criteria specified by both process class rules (a print process is also an operating system process) we will need to ensure that the process rule for all i5/OS processes has a higher position number than (it will be listed after) the process class for the print processes.
- | 18. In the **Service class** box, select **i5OS processes**
- | 19. Click **New** in the rules area to define a new process rule. The following steps will define criteria that will be used to associate system processes with the service class defined in a previous procedure. In this example, the process rule will qualify all processes on the i5/OS system "mycoSys1."
- | 20. Click the **Select filter type** box and select **EWLM: System Name**.
- | 21. Select **= =** for the operation.
- | 22. Specify **mycoSys1** for the rule value.
- | 23. Click **OK** to create the rule.

24. Click **OK** to create the process class. This process class will now qualify all i5/OS processes, except for the print process, while will be classified separately, and associate it with the i5/OS processes service class.
25. Click **Next** twice to skip the **Partition classes** page and proceed to the **Service policy** page.
26. On the **Service policy** page, click **Finish** to save the domain policy.

Deploy the domain policy and activate the service policy:

After defining the domain policy, you will need to deploy and activate it so that EWLM will start collecting data.

1. Go to **Set up** → **Domain policies**.
2. Select your domain policy so that it is highlighted.
3. Select **Deploy** from the **Select Action** field and click **Go**.
4. On the **Deploy domain policy** page, select **EWLM Service Policy** in the **Service policy to activate** screen. Or, select the appropriate service policy name depending on what you named it. The previous procedures indicated that you could use the default service policy, which is called **EWLM Service Policy**.
5. Click **Deploy**.

Determine if performance goals are met:

After you deploy the domain policy and activate the service policy, EWLM collects performance data. You can then view reports to see if the performance requirements are being met.

Use the Monitor section of the EWLM Control Center to determine if the transactions processed by the WebSphere Application Server, and the operating system processes from i5/OS are adhering to the performance goals that you defined. To do this, view the following reports:

- **Service classes:** View the performance index for the **WAS transactions** service class.
If the performance index is greater than one, it means that the goal being defined in the policy is being met. If the performance index is less than one, the goal is not being met. A performance index of none means that no transactions are currently associated with the service class.
 - **Service classes:** View the performance index for the **i5OS print processes** service class.
 - **Service classes:** View the performance index for the **i5OS processes** service class.
 - **Transaction classes:** View the **WAS transactions** process class to view performance data for the WebSphere Application Server transactions that meet the criteria specific in the rule for this transaction class.
 - **Process classes:** View the **i5OS processes** and **i5OS print processes** process class to view performance data for the related i5/OS jobs.
- Note:** If you would like to see the processor usage for the i5/OS processes, you must add the processor utilization for both process classes together. This is because the i5/OS print process is also an operating system process, but is classified separately (by the first applicable process class).
- **Managed Server:** View each class of work's contribution to the Central Processing Unit (CPU) usage for the managed server. This allows you to identify which class of work consumes the most of the CPU.
 - Use any other reports and monitors as necessary to troubleshoot performance problems, if they exist.

Scenario: Using the sample banking domain policy

View this scenario to learn how you can use EWLM to monitor work in your EWLM domain. Furthermore, this scenario uses the sample banking domain policy to demonstrate how you can incorporate your business goals into an EWLM domain policy.

Situation

As a system administrator at your company, you support a systems environment for a stock brokerage and banking Web site where customers can transfer funds, perform account queries, buy and sell stock, and handle mortgage loans. You are responsible for defining the Service Level Agreement (SLA) for these applications in terms that your systems environment can use. This scenario describes how a domain policy was created for the banking applications.

Objective

In this scenario, MyCo, Inc. wants to define an EWLM domain policy for its Web banking applications. The applications run on:

- Microsoft Internet Information Services (IIS) on Windows 2003 server
- IBM WebSphere Application Server 5.1.1 on Solaris
- DB2 Universal Database (UDB) Version 8.2 on AIX

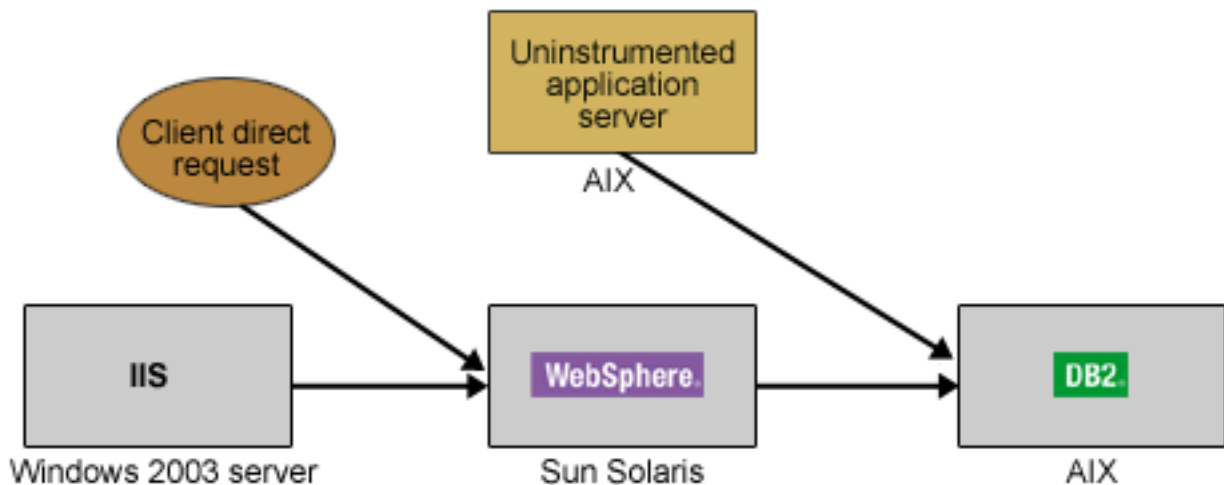
In addition, MyCo has recently merged with another bank and has integrated another Web application server running on AIX.

MyCo's Web banking applications include a funds transfer, an account query, a mortgage rate advisor, stock buying for brokers, and a bank statement processing application. The vice president of finance requires quarterly reports on the performance of each of the bank region's brokerage buyers. They serve the northeast and northwest regions of the United States.

You, as the performance system administrator, are responsible for making sure the performance objectives for the Web banking applications are met. The objectives for this scenario are:

- To define service classes for the different applications offered on their Web site.
- To specify which work should be assigned to the service classes.
- To report the performance of brokerage buyers for the northeast and northwest regions.
- To include the application running on the other Web application server.
- To use the above to define a domain policy representing the Web banking environment.

Details



EWLM managed servers:

- | • AIX 5L™
- | • Microsoft Windows 2003 Server
- | • SUN Solaris 8

| **Applications:**

- | • Microsoft Internet Information Services (IIS) plug-in to WebSphere, with ARM-instrumentation enabled
- | • WebSphere Application Server 5.1.1, with ARM-instrumentation enabled
- | • IBM DB2 Universal Database Version 8.2, with ARM-instrumentation enabled
- | • Other Web application server, not ARM-instrumented

| **Bank Web site applications defined in MyCo's Service Level Agreement (SLA):**

- | • Account queries
- | • Brokerage buyers
- | • Mortgage rates
- | • Statement processing
- | • Stock quotes

| **Configuration details**

| In this scenario, you will define an EWLM domain policy for MyCo's bank Web site. Assuming that the EWLM domain manager has been installed and configured on MyCo's system and the EWLM Control Center has been started, complete the following:

| **Task 1: Define performance objectives in service classes:**

| View this task of the scenario to learn how this company defined the performance objectives in service classes.

| After you have installed and configured the EWLM domain manager and started the EWLM Control Center, you can begin defining a domain policy for the Web banking environment. You start by defining a domain policy name: Sample Banking Domain Policy. Then, you can proceed to the next steps:

| **Step 1: Define the business applications that run in the Web banking environment**

| From the business applications that process the banking Web Site, determine the following:

- | • Whether the work is transaction based or process based
- | • On which middleware the business applications run
- | • Which of the middleware applications are ARM 4.0 instrumented

| The Web banking applications include the following functions:

- | • Funds transfer
- | • Account query
- | • Mortgage rate advisor
- | • Brokerage stock buying
- | • Static Web page content that includes graphics and text
- | • WebSphere Application Server Enterprise Java Beans (EJB) application, which processes bank statements

| All are transaction based work. All run on the ARM 4.0 instrumented middleware applications except for the mortgage rate advisor application, which runs on the Web application server from the bank merger (herein known as the non-instrumented Web application server). The non-instrumented Web application

server does, however, pass requests to DB2 UDB, which is ARM 4.0 instrumented. This means that the non-instrumented Web application server can be included in the end-to-end domain policy only as process-oriented work because transaction-level performance data is not available for a non-instrumented application. Because the non-instrumented Web application server passes data to DB2 UDB, which is instrumented, the DB2 UDB portion of transactions can be monitored and managed by EWLM and therefore, can be included in the domain policy.

Step 2: Define the performance objectives for the work

In this step, you will define service classes based on the performance objectives for the Web banking work and assign a business importance to the work. *Importance* is a value that defines how important it is to the business that the performance objective be met. The performance objectives depend on whether the work is considered transaction based or process based. Response time objectives usually apply to transaction-based work. For process-based work, EWLM provides a goal type called *velocity*. Velocity goals are appropriate for long-running work such as system daemons, batch jobs, and server processes. Velocity defines how fast work should run when ready, without being delayed for processor, storage, and I/O.

The Web banking Web site has the following service classes with response times defined in their service level agreement:

Service class	Importance	Performance goal
WebSphere EJB Service Class	Highest	85% complete in 4 seconds
Account Queries	High	80% complete in 3 seconds
Brokerage Buyers	Highest	80% complete in 1 seconds
Stock Quotes	Medium	80% complete in 10 seconds
Static Web Serving	Low	90% complete in 20 seconds

For the mortgage rate advisor application, consider the end-to-end response time and the time the transactions normally spend in the database query. Because the Web application server is not instrumented, the performance objectives defined in the domain policy represent only the portion of time the transactions spend in DB2 UDB. The application is not quite real-time because the Web site provides a wizard to query mortgage rates with the rates sent as an e-mail, so the turnaround need not be instantaneous. The service level agreement states that an e-mail will be sent within the next hour after a query is made; the portion of the time spent in DB2 for the query is estimated to be about 20 seconds, so the response time objective is set for 1 minute.

Service class	Importance	Performance goal
Mortgage Rates	Medium	1 minute average response time

For the mortgage rate advisor application, EWLM can manage and monitor the non-instrumented Web application server regions. To do this, you must define a service class for the non-instrumented Web application server region with a velocity goal. Specify a general name for this service class. Then, if another bank merger occurs and they would like to integrate a non-instrumented application, it can be included in this service class. Velocity is specified as a category: Fastest, Fast, Moderate, Slow, or Slowest. You might also want to manage the processes running on Web Browsers in the banks. The browsers, Internet Explorer and Netscape Navigator are processes, and are also assigned a velocity goal.

Service class	Importance	Performance goal
Non-Instrumented Web Applications	High	Fast Velocity
Web Browsers	Medium	Moderate Velocity

| For work not otherwise categorized, EWLM provides a service class in every domain policy called EWLM Service Class. It is assigned a discretionary goal. Discretionary goals specify that work is to be completed when resources are available. You can assign a discretionary goal type to a service class for low priority work that does not require a particular performance goal. EWLM processes discretionary work using resources not required to meet the goals of other service classes.

| **Step 3: Define the workloads**

| Based on the service classes, define the following workloads for reporting purposes:

- | • **WebSphere Applications**, for the WebSphere EJB application that processes the bank statements.
- | • **DB2 Applications**, for the mortgage advisor transactions originating from the non-instrumented application running in DB2 UDB.
- | • **Non-Instrumented Applications**, for the Web application server process and the Web browser processes.
- | • **Web Site Applications**, for the funds transfer, account query, mortgage rate advisor, and brokerage buyer transactions, and the static Web serving work.

| EWLM provides a workload, *EWLM Workload*, which is assigned to the EWLM service class.

| **Task 2: Define work to service classes using transaction and process classes:**

| After the Web banking service classes are defined, you must define which work should be assigned to each service class by using transaction classes and process classes. The transaction classes define the transactions that should be assigned to a service class. The process classes define the processes that should be assigned to a service class.

| In the transaction class or process class, you define the rules to identify the work. The attributes of incoming work are compared to these rules and, if there is a match, the rule is used to assign a transaction or process class to the work. For example, a transaction class may contain a rule that specifies that all transactions from server name `www.mycobank.com` should be assigned to the **Static Web Serving** service class. The attributes of the work are called filters, and include URI, system name, host name, user name, and so on. EWLM provides some general filters, and the applications or platforms provide some filters specific to their environment, such as EJB name for WebSphere or Executable Path for AIX. You can also use wildcard (`*`) and masking (`\?`) to define general rules for a group of transactions.

| EWLM monitors and manages resources to meet the goals defined for service classes, and monitors based on transaction classes and process classes. You can view performance data on transaction classes and process classes in addition to performance data on service classes. You can use transaction classes and process classes to not only define which work should be assigned to a service class, but to define which work within a service class you want to specifically monitor.

| With the service classes defined, you can define your transaction classes and process classes in the next steps.

| **Step 1: Define the entry applications for the service classes and add the applications to the domain policy**

| To begin defining transaction classes, start with a service class. Determine the first ARM-instrumented application that processes the transactions representing that service class in the Web Banking environment. The first ARM-instrumented application that processes the transactions specifies the start of the end-to-end performance for the service class. The first application is known as the entry application. The entry application is where EWLM looks up the classification rules defined in the transaction classes.

For each of the service classes representing transaction-based work, you determine the entry application.

Service class	Entry application
Static Web Serving	IIS plug-in to WebSphere
Account Queries	IIS plug-in to WebSphere
Brokerage Buyers	IIS plug-in to WebSphere
Stock Quotes	IIS plug-in to WebSphere
WebSphere EJB Service Class	WebSphere Application Server
Mortgage Rates	IBM DB2 Universal Database

The entry application for the Static Web Serving, Account Queries, Brokerage Buyers, and Stock Quotes service classes is IIS, which, in EWLM, is included in the general IBM Web serving plug-ins.

The entry application for the WebSphere EJB service class that processes the bank statements is WebSphere Application Server, since the EJB is from a client direct request.

The entry application for Mortgage Rates service class is IBM DB2 Universal Database. While the transaction originates in the Web application server from the bank merger, the Web application server is not ARM-instrumented, so it cannot be the entry application.

In the EWLM Control Center, you must add the entry applications to the domain policy. When you add an application to the domain policy, EWLM defines a default transaction class for the application. If a transaction is not defined in any other classification rules of transaction classes in that application, the transaction is assigned to the default transaction class. The default transaction class has a wildcard classification rule, `(*) stringMatch (*)`, that indicates that all filters will match. The default application transaction class is assigned to the EWLM provided service class, called EWLM Service Class. The EWLM Service Class has a discretionary goal. You can change the service class assigned to the default transaction class.

If unidentified transactions come into the EWLM domain from an ARM-instrumented middleware application that has not been defined in the domain policy, EWLM defines a default application with a default transaction class assigned to the EWLM Service Class. The default application environment does not have a name; it is defined as a wildcard application and you cannot edit it from the Control Center.

Step 2: Define the transaction classes under the entry application for the service classes

MyCo Inc. is very particular about the naming conventions it uses for its systems and transactions, which is useful for defining the classification rules. The following tables identifies the applications that EWLM will monitor. In addition, it specifies the details of the domain policy for each application.

IBM Web serving plug-in as an entry Application

The IBM Web serving plug-in is the entry application for most of MyCo's three-tiered banking applications. The Static Web Serving service class defines the static Web pages that appear on the banking Web site. They can be identified by the request for the Web site's host name:

Transaction Class	Description	Service Class	Rules
Static Web Serving	Pages and graphics for the bank Web site	Static Web Serving	HostInfo == www.mycobank.com

The stock queries can be identified by the query string passed to DB2 specifying the stock ID:

Transaction Class	Description	Service Class	Rules
Stock Query	Stock ID passed as DB2 query	Stock Quotes	QueryString == WBstock="stockID"

The Vice President of finance wants to have reporting on the performance of the northeast and northwest brokerage regions. The IIS server names include the region, so two service classes are defined for the two regions using the System Name filter:

Transaction Class	Description	Service Class	Rules
NorthWest	NW region brokerage buyers	Stock Quotes	EWLM:System Name == myco.northwest.com
NorthEast	NE region brokerage buyers	Stock Quotes	EWLM:System Name == myco.northeast.com

The account queries can be identified by the query string passed to DB2. The string includes the user name and the account ID number:

Transaction Class	Description	Service Class	Rules
Account Query	Account query transactions to UDB	Account Queries	QueryString == WBname="username%" AND WBaccount="accountID%"

When the IBM Webserving plug-in application was added to the domain policy, EWLM defined a default transaction class with the assigned service class EWLM Service Class. Any transactions that come into the IBM Web serving plug-in that do not match on the rules defined for the other transaction classes will be assigned to 'Default Web serving plugin' application. The default transaction class for an application always has the wildcard rule to match on all filters. You cannot edit the rule for the default transaction class:

Transaction Class	Description	Service Class	Rules
Default - IBM Web serving plug-in	Default transaction class for IIS	EWLM Service Class	(*) == (*)

WebSphere as Entry Application

WebSphere is the entry application for the EJBs that support the banking applications. Because the only service class for which WebSphere is the entry application is the WebSphere EJB class, use the default transaction class for WebSphere to assign the WebSphere EJB Service Class. All transactions originating in WebSphere are assigned the WebSphere EJB service class:

Transaction Class	Description	Service Class	Rules
Default - WebSphere	EJBs supporting the banking applications	WebSphere EJB Service Class	(*) == (*)

IBM DB2 Universal Database as Entry Application

The IBM DB2 Universal Database is the entry application for the non-instrumented Web application server mortgage rate transactions. Those transaction are the only ones with DB2 UDB as the entry, so the default transaction class can also be used for the Mortgage Rates service class:

Transaction Class	Description	Service Class	Rules
Default - IBM DB2 Universal Database	Default transaction class for DB2	Mortgage Rates	(*) == (*)

Step 3: Define the platforms for the process based service classes

Process-based work almost always runs on a single server or set of servers on a single platform. To define the work in process-based service classes, you start with the platform on which the processes run.

Service class	Platform
Non-Instrumented Web Applications	AIX
Web Browsers	Windows

In the Web Banking environment, the non-instrumented Web application server runs on AIX. In the domain policy, you add the AIX platform. Then, you define a process class to specify the processes to be assigned to a service class. There are no default process classes for platforms. Any processes that you want EWLM to manage and monitor must be defined explicitly in a process class.

If you want to monitor the performance of the Web Browsers that the banking employees use, add the Windows platform.

Step 4: Define the process classes for the process based service classes

For the platforms, you define the process classes.

AIX Platform

For the Non-Instrumented Web Applications service class running on AIX, you define a process class using the Executable Path filter. The application server runs under the webapserver/webpages root.

Process Class	Description	Service Class	Rules
Web Application Server Processes	Example process class for uninstrumented applications.	Non-Instrumented Web Applications	ExecutablePath -- webapserver/webpages

Windows Platform

You define the following process classes for the Web Browsers service class:

Process Class	Description	Service Class	Rules
Netscape Web Browser	Netscape Navigator	Web Browsers	ExecutablePath == C:\ProgramFiles\Netscape\Communicator\Program\netscape.exe=
IE Web Browser	Internet Explorer	Web Browsers	ExecutablePath == C:\Program Files\Internet Explorer \IEXPLORE.EXE

| **Task 3: Review and verify the service policy and deploy the domain policy:**

| You have defined a domain policy with workloads, service classes, applications, transaction classes,
| platforms, and process classes for the Web Banking environment. Before you deploy the domain policy
| and make it available to the domain, you should review the service policies and verify the domain policy.

| **Step 1: Review the service policy**

| When the banking domain policy was created, EWLM defined a service policy called 'Service Policy'.
| Because MyCo's 2005 SLA specifies one set of performance objectives for the service classes, only one
| service policy is required. To make it specific to the Web banking environment, you can rename Service
| Policy to Banking 2005 Service Policy.

| You should review the goals defined for the service classes in the 'Banking 2005 Service Policy'. In the
| EWLM Control Center, you can view the 'Sample Banking Domain Policy' and review the 'Banking 2005
| Service Policy'.

Service Policies		
Name: Banking 2005 Service Policy Description: SLA for 2005 banking applications	Service policy goals	
	Non-instrumented Web applications	
	Velocity	Importance: High Level: Fast
	Web browsers	
	Velocity	Importance: Medium Level: Moderate
	WebSphere EJB service class	
	Percentile Response Time	Importance: Highest Response time: 00:00:04.000 Percentile: 85
	Account queries	
	Percentile Response Time	Importance: High Response time: 00:00:03.000 Percentile: 80
	Brokerage buyers	
	Percentile Response Time	Importance: Highest Response time: 00:00:01.000 Percentile: 80
	Stock quotes	
	Percentile Response Time	Importance: Medium Response time: 00:00:10.000 Percentile: 80
	Static Web serving	
	Percentile Response Time	Importance: Low Response time: 00:00:20.000 Percentile: 90
	Mortgage rates	
Average Response Time	Importance: Medium Response time: 00:01:00.000	
EWLM service class		
Discretionary		

Step 2: Verify the domain policy in the Web banking domain

When you have reviewed your service policy, you should verify the domain policy in the Web banking domain. When verifying a domain policy, EWLM determines whether the policy is formatted properly, whether it contains valid syntax, and whether the managed servers will activate it successfully. To verify the domain policy, follow these steps:

- | 1. In the EWLM Control Center, select **Set up** → **Domain Policies**.
- | 2. Select **Sample Banking Domain Policy** from the list of domain policies.
- | 3. Select **Verify** from the **Select Action** field.
- | 4. Click **GO**.

| The EWLM Control Center verifies the domain policy on all servers defined in the domain. After the domain policy successfully verifies, you are ready to deploy it and activate the *Banking 2005 Service Policy*.

| **Step 3: Deploy the domain policy and activate the service policy**

| To deploy the domain policy and activate *Banking 2005 Service Policy*, follow these steps.


- | 1. In the EWLM Control Center, select **Set up** → **Domain Policies** .
- | 2. Select *Sample Banking Domain Policy* from the list of domain policies.
- | 3. Select **Deploy** from the **Select Action** field.
- | 4. Click **GO**.
- | 5. On the Deploy Domain Policy page, select **Banking 2005 Service Policy** for the **Service Policy to Activate** option.

| EWLM displays the **Deploy Domain Policy and Activate Service Policy Status** page. If the domain policy is deployed successfully and the service policy is activated successfully, you have completed this task. You have successfully defined, verified, and deployed a domain policy, and activated a service policy for the Web Banking environment. You can begin to view the Monitor section of the EWLM Control Center to view the performance of the work requests in your EWLM domain.

Specified operating environments for EWLM

View this topic to ensure that your system meets the hardware and software requirements for each component of EWLM.

The hardware and software required to use EWLM varies by platform and function of EWLM that you want to use. Make sure that you have applied the most recent fix packages. You need to install required operating system fixes for EWLM base function as well as conditional fixes depending on what features of EWLM you want to use. For required and recommended fixes, go to the IBM Virtualization Engine

Systems Services Fixes  Web site (<http://www14.software.ibm.com/webapp/set2/sas/f/VirtualizationEngine/home2.html>). View the following for more information:

- Domain manager hardware requirements
- Managed server hardware requirements
- Domain manager and managed server software requirements
- Control Center Web browser requirements
- Application requirements

Note: The requirement data in this topic was obtained or projected based on performance measurements in controlled environments. Results obtained in other environments may vary significantly and do not predict a specific customer's environment.

Table 5. Domain manager hardware requirements.

Domain manager hardware requirements ¹						
Environment	Processor capacity	Memory (RAM) ²	Disk space ³	WebSphere Application Server maximum heap size ⁴	Domain manager JVM maximum heap size ⁵	Comments
AIX on System 5xx, eServer™ 5xx, or pSeries 6xx	Relative Performance >1.8	640 MB	6 GB	256 MB	512 MB	
i5/OS on eServer 5xx, or iSeries 8xx	≥1300 CPW	1.5 GB	6 GB	256 MB	512 MB	
Linux on System 5xx, eServer 5xx, OpenPower™, iSeries 8xx, or pSeries 6xx	Relative Performance >1.8	640 MB	6 GB	256 MB	512 MB	
Linux on xSeries	2 GHz	640 MB	6 GB	256 MB	512 MB	
Linux on zSeries	Processing capacity of a zSeries 890 2086-310 or greater	640 MB	6 GB	256 MB	512 MB	
Windows on xSeries	2 GHz	512 MB	6 GB	256 MB	512 MB	
Windows on eServer i5 5xx, or iSeries 8xx	2 GHz	512 MB	6 GB	256 MB	512 MB	
z/OS on zSeries	Processing capacity of a zSeries 890 2086-210 or greater that supports z/OS	1.25 GB	6 GB	512 MB	512 MB	The memory requirement is based on an unconstrained 6 GB memory system. Actual working set memory requirements are likely less.

Table 5. Domain manager hardware requirements. (continued)

Domain manager hardware requirements ¹						
Environment	Processor capacity	Memory (RAM) ²	Disk space ³	WebSphere Application Server maximum heap size ⁴	Domain manager JVM maximum heap size ⁵	Comments
<p>Note:</p> <ol style="list-style-type: none"> These hardware requirements are sized for a moderate number of policy definitions that supports 10 to 50 managed servers Specifies the additional memory required for the domain manager and the Virtualization Engine WebSphere Application Server. This does not include base operating system memory requirements. Specifies the recommended amount of disk space to install and run EWLM. Depending on the complexity of your EWLM domain, you might need additional space. For EWLM requirements, this is the default maximum heap (-maxHeap) size. Monitor the memory and the processor resource usage as the configuration increases in size to more accurately project your requirements. WebSphere Application Server maximum heap (-maxHeap) settings potentially need to be increased to avoid an out-of-memory condition for Java heap. To change the WebSphere Application Server maximum heap size, use the WebSphere Application Server. For more information, see the Troubleshooting EWLM Control Center. These are the default values for the domain manager's Java Virtual Machine (JVM) maximum heap sizes, except for some i5/OS environments, which have a WebSphere maximum heap size of *NOMAX. Note that as the size and complexity of your EWLM domain increases, you may need to adjust this value accordingly. Performance metrics indicate that these default values provide above satisfactory performance ratings for small EWLM configurations. For medium configurations, the default values provide satisfactory performance ratings. For larger configurations that contain more than 100 managed servers, you may need to increase this value. 						

Table 6. Managed server hardware requirements.

Managed server hardware requirements			
Environment ^{1, 3}	Processor capacity ²	Memory (RAM)	Disk space
Any supported EWLM managed server environment. ¹	Varies by application usage.	128 MB	128 MB
<p>Note:</p> <ol style="list-style-type: none"> Managed server hardware requirements are not available for EWLM managed servers that run on a Hewlett-Packard or SUN system. The EWLM managed server function increases the processor usage approximately 2 to 12 percent of the application processing requirements, depending on the type of workload. Workloads with many small interactions between systems will see the largest amount of additional resource requirements. 			

Table 7. Domain manager and managed server software requirements and list of supported functions.

Domain manager and managed server software requirements					
Environment		Domain manager and EWL Control Center	Managed server and firewall broker	Managed server - load balancing	Partition management ³
Platform	Operating system ¹				
iSeries 8xx	IBM i5/OS V5R4	X	X		
	IBM i5/OS V5R3	X	X		
	RedHat Enterprise Linux AS 4.0	X			
	SUSE Linux Enterprise Server (SLES) 9	X	X ⁸	X ⁸	
	Windows ²	X ⁴	X	X	
System 5xx, eServer 5xx, OpenPower ⁷	IBM i5/OS V5R4	X	X		X
	IBM i5/OS V5R3	X	X		X
	IBM AIX 5L Version 5.2		X	X	
	IBM AIX 5D Version 5.3	X	X	X	X
	RedHat Enterprise Linux AS 4.0	X			
	SUSE Linux Enterprise Server (SLES) 9	X	X ⁸	X ⁸	X ⁸
	Windows ^{2,9}	X ⁴	X	X	
pSeries 6xx	IBM AIX 5L Version 5.2		X	X	
	IBM AIX 5D Version 5.3	X	X	X	
	RedHat Enterprise Linux AS 4.0	X			
	SUSE Linux Enterprise Server (SLES) 9	X	X ⁸	X ⁸	
xSeries	RedHat Enterprise Linux AS 4.0	X			
	SUSE Linux Enterprise Server (SLES) 9	X	X ^{6,8}	X ⁸	
	Microsoft Windows ²	X ⁴	X ⁶	X	
zSeries	RedHat Enterprise Linux AS 4.0	X			
	SUSE Linux Enterprise Server (SLES) 9	X	X ⁸	X ⁸	
	z/OS V1R6	X	X		
	z/OS V1R7	X	X		
Hewlett-Packard 9000 and Integrity	HP-UX 11iV1		X	X ⁵	
SUN	Sun Microsystems Solaris 9 (SPARC Platform Edition)		X	X ⁵	
Other systems that are certified for IA-32 or compatible processors for Microsoft Windows	Microsoft Windows ²		X	X	

Table 7. Domain manager and managed server software requirements and list of supported functions. (continued)


Domain manager and managed server software requirements					
Environment		Domain manager and EWLM Control Center	Managed server and firewall broker	Managed server - load balancing	Partition management ³
Platform	Operating system ¹				
<p>Note:</p> <ol style="list-style-type: none"> Make sure that you apply the most recent fix packages. EWLM will not function properly if you do not install the required operating system fixes. For required fixes, go to the IBM Virtualization Engine Systems Services Fixes  Web site (http://www14.software.ibm.com/webapp/set2/sas/f/VirtualizationEngine/home2.html). Select a product name and the most recent fix period. Click By OS → Additional fix downloads → Required OS fixes. View the Required fixes to determine which operating system updates are required for you to use EWLM. Support for Windows implies support for Windows 2000 Server, Windows 2000 Advanced Server, Windows Server 2003 Standard Edition, and Windows Server 2003 Enterprise Edition, unless stated otherwise. EWLM partition management requires server firmware release 2.3.0 for EWLM to manage the partitions on the system. Domain manager is supported only on Windows 2003 Enterprise Edition and Windows 2003 Standard Edition. Load balancing support on this operating system requires that you install the LSOF utility on the managed server. View the EWLM managed server installation topic collection for your operating system for more details. For xSeries systems, EWLM also supports guest operating systems that run on VMware ESX 2.5 Server. To learn more about this, see Virtualizing a system. OpenPower systems do not support AIX or i5/OS; they support only Linux (RedHat Enterprise Linux AS 4.0 and SUSE Linux Enterprise Server (SLES) 9). EWLM managed server support for Linux is not available with the base function shipped with Version 2 Release 1 of IBM Enterprise Workload Manager. Ensure that you acquire and install the latest EWLM managed server fix package. For more information about how to identify which Virtualization Engine software fixes are available, see Fixes and Updates in the Virtualization Engine Management collection topic. For EWLM to monitor work that runs on a Linux operating system, the work must not run in a LinuxThreads environment. For more details, see "Troubleshooting: Applications" on page 326. For the operating environments specified, Windows is only supported on eServer i5 5xx. 					

Table 8. Control Center Web browser requirements.



Control Center Web browser requirements
<ul style="list-style-type: none"> Internet Explorer (IE) 6.0 with Service Pack 1. Java Runtime Environment (JRE) 1.4.2 with Service Release 2. Adobe SVG viewer. EWLM prompts you to install this function. If you cancel the install and choose to install it later, you can download the appropriate SVG viewer at the Adobe Web site . Pop-up windows enabled. Add the domain manager host name to the Web browser's local intranet settings to avoid security errors.

Table 9. Application requirements for EWLM to monitor application-level transactions.

Application requirements
Application¹
IBM DB2 Universal Database, Version 8.2 ²
IBM WebSphere Application Server, Version 5.1.1.1 and Version 6.0 ⁴
IBM Web serving plug-ins, which include applications provided by WebSphere Application Server and other independent applications. ³

Table 9. Application requirements for EWLM to monitor application-level transactions. (continued)

Application requirements
<p>Note:</p> <ol style="list-style-type: none"> The applications that EWLM can monitor are not limited to this list. These are the applications that are predefined in the EWLM Control Center. Make sure that you apply the most recent fix packages for these applications. Refer to your product documentation to ensure that your application has the most recent fixes installed. In addition, some applications require EWLM specific fixes. EWLM will not monitor these applications unless you install the conditionally required application fixes. They are conditionally required in that you need to install the fix only if you want EWLM to monitor that application. For conditionally required fixes that allow EWLM to monitor a specific application, go to the IBM Virtualization Engine systems services fixes  Web site (http://www14.software.ibm.com/webapp/set2/sas/f/VirtualizationEngine). Select a product name and the most recent fix period. Click By OS → Additional fix downloads → Required OS fixes. View the Conditionally required fixes to determine which application fixes are required for the applications that you want EWLM to monitor. i5/OS supports DB2 UDB for iSeries Version 5 Release 3. The EWLM Control Center uses a common application name of IBM Web serving plug-ins. To monitor one of the applications that is associated with this application name, you must add the IBM Web serving plug-in application to a domain policy. Then, create transaction classes for with this application name. Use a combination of transaction classes to distinguish between the various applications that the Web serving plug-in application name includes. The WebSphere Application Server plug-ins are not ARM-instrumented to support z/OS. To monitor WebSphere Application Server on z/OS, you must use WebSphere Application Server 6.0 or later.

Related concepts

“Partition workload group” on page 16

A *partition workload group* identifies a set of partitions that reside on the same physical system. The partition profile specifies the name of the partition workload group that it belongs to, if applicable.

“EWLM domain manager” on page 12

The *domain manager* is the central point of control for an EWLM management domain.

“EWLM managed server” on page 12

A *managed server* is a server or server instance that EWLM monitors.

“Firewall broker” on page 15

A *firewall broker* provides the ability to allow communication through a firewall.

Related information

Fixes and updates

IBM Virtualization Engine systems services fixes

Planning for EWLM

Use these planning worksheets before you install and configure EWLM.

These planning worksheets contain the tasks and considerations that you must address before beginning the installation of IBM Enterprise Workload Manager (EWLM). Print this page, record your answers, and use the completed worksheet to assist you as you install and configure EWLM.

Complete the following worksheets as applicable to your environment:

- Prerequisite planning worksheet
- User account information worksheet
- Domain topology information worksheet
- Domain manager planning worksheet

- Managed server planning worksheet
- Firewall broker planning worksheet

Prerequisites planning worksheet

-
- Complete the Virtualization Engine planning advisor for installation guidance if you have not already done so. It provides topology recommendations for deploying EWLM and other Virtualization Engine systems services across your enterprise.
-
- Determine if you want to install additional Virtualization Engine system services. Note that the items that you determine that you want to use, you must add to your installation plan. For more information on how various Virtualization Engine products work with one another, see Virtualization Engine products: Overview.
-
- | — If you plan to install other Virtualization Engine system services, review the following topics:
- | • General requirements across Virtualization Engine products
- | • Supported operating environments
-

User account information worksheet

Note: You can use the same user ID for multiple tasks.

Description of user ID	User ID
<p>Installation user ID: Ensure the system administrator responsible for installing EWLM has the following user authorities on each of the applicable operating systems. This is the user that will install the domain manager, managed servers, and firewall brokers, if applicable.</p> <p>AIX: Requires root authority.</p> <p>HP-UX: Requires root authority.</p> <p>i5/OS: Requires *ALLOBJ, *SECADM, *JOBCTL,*IOSYSCFG, and *SAVSYS special authority.</p> <p>Linux: Requires root authority.</p> <p>Solaris: Requires root authority.</p> <p>Windows: Must be a member of the Administrators group and have Act as part of the operating system user right. If the system is part of a domain, then the user must also be added to the Domain Admin group.</p> <p>z/OS: Requires:</p> <ul style="list-style-type: none"> • The ID is defined to z/OS UNIX[®] System Services • UID(0) (superuser authority), or READ or higher access to the RACF[®] profile BPX.SUPERUSER in the FACILITY class • READ or higher access to the RACF profiles BPX.FILEATTR.PROGCTL, BPX.FILEATTR.APF and BPX.FILEATTR.SHARELIB in the FACILITY class 	

Description of user ID	User ID
<p>Configuration user ID: Ensure the system administrator responsible for configuring EWLM has the following user authorities on each of the applicable operating systems. This is the user that will configure the domain manager, managed servers, and firewall brokers, if applicable.</p> <p>AIX: Requires root authority and the process under which the WebSphere Application Server runs must also run as root.</p> <p>HP-UX: Requires root authority.</p> <p>i5/OS: Requires *ALLOBJ, *SECADM, *JOBCTL, and *IOSYSCFG special authority</p> <p>Linux: Requires root authority and the process under which the WebSphere Application Server runs must also run as root.</p> <p>Solaris: Requires root authority.</p> <p>Windows: Must be a member of the Administrators group and have Act as part of the operating system user right. If the system is part of a domain, then the user must also be added to the Domain Admin group.</p> <p>z/OS: Requires superuser authority.</p>	
<p>WebSphere Application Server user ID: On the system acting as the domain manager, create a user for administering the WebSphere Application Server instance with the following authorities. This is the user ID that you will specify for the <i>adminUser</i> parameter on the EWLM commands that use this parameter.</p> <p>AIX: Requires no special authority, but the process under which the WebSphere Application Server runs must run as root.</p> <p>i5/OS: Requires no special authorities.</p> <p>Linux: Requires no special authority, but the process under which the WebSphere Application Server runs must run as root.</p> <p>Windows: Requires no special authorities.</p> <p>z/OS: Requires READ or higher access to the RACF profile <i>cellname.ADMINISTRATOR</i> in the EJBROLE class. For more information, see “Configuring a domain manager on z/OS” on page 168</p>	
<p>User ID that starts the WebSphere Application Server: Determine under which user ID the WebSphere Application Server process will run. The process runs under the user ID that starts the WebSphere Application Server instance by running the startWAS command, which is part of the process to start EWLM, unless otherwise noted.</p> <p>AIX: Requires root authority to run the command and the process under which the WebSphere Application Server runs must also run as root.</p> <p>i5/OS: Requires no special authorities to run the command. The WebSphere Application Server processes run under the QEJBSVR user ID.</p> <p>Linux: Requires root authority to run the command and the process under which the WebSphere Application Server runs must also run as root.</p> <p>Windows: Must be a member of the Administrators group to run the command. The WebSphere Application Server processes run under the Local System Account.</p> <p>z/OS: This user ID is automatically created by the createWAS configuration command. For more information, see “Configuring a domain manager on z/OS” on page 168</p>	

Description of user ID	User ID
<p>EWLM Control Center user IDs: Determine which user or groups will have access to the EWLM Control Center. You will then assign these users or groups to an appropriate role for administering and monitoring EWLM from the EWLM Control Center. For more information on assigning user IDs to EWLM Control Center roles, see “Assigning users to EWLM Control Center roles” on page 117.</p> <p>These users or groups do not require any special authorities or rights on the server where the domain manager and EWLM Control Center functions run. However, you should verify that they are able to log on to the operating system interfaces before they attempt to sign on to the EWLM Control Center.</p>	
<p>User ID that starts the managed server: Determine under which user or group the managed server process will run. This is the user that will start the managed servers. The managed server must be able to call the EWLM APIs, which require that users have the following authorities or rights:</p> <p>AIX: Requires either root authority or you can give a nonroot user authority. To give a nonroot user authority, see “Authorizing AIX non-root users to start EWLM on an AIX managed server” on page 118.</p> <p>HP-UX: Requires root authority.</p> <p>i5/OS: Requires All Object (*ALLOBJ) authority or authority to the startMS command. The managed server processes run under the QWLM user profile. The QWLM user profile is created by EWLM.</p> <p>Linux: Requires root authority.</p> <p>Solaris: Requires root authority.</p> <p>Windows: Must be a member of the Administrators group to start the managed server. The managed server processes run under the Local System Account.</p> <p>z/OS: Requires that the managed server run under the user profile that the superuser sets up with the necessary access rights. For more information on how a superuser sets this up, see “Authorizing z/OS users to start EWLM” on page 185.</p>	
<p>User ID that starts the domain manager: Determine under which user or group the domain manager processes will run. These processes include starting and stopping the domain manager, which require that the user or group have the following authorities or rights:</p> <p>AIX: Requires root authority.</p> <p>i5/OS: Requires All Object (*ALLOBJ) authority or authority to the startDM command. The managed server processes run under the QWLM user profile. The QWLM user profile is created by EWLM.</p> <p>Linux: Requires root authority.</p> <p>Windows: Must be a member of the Administrators group.</p> <p>z/OS: Requires superuser authority and other access rights. For more information, see Authorizing z/OS users to start the managed server</p>	

Description of user ID	User ID
<p>User ID that starts an application that EWLM monitors: Ensure that the users, under which ARM-instrumented applications run, can access the appropriate ARM libraries, which require the following authorities and rights. These are the user IDs that start the applications in which EWLM monitors.</p> <p>AIX: Requires either root authority or you can give a nonroot user authority. To give a nonroot user authority, see “Authorizing AIX non-root users to start EWLM on an AIX managed server” on page 118.</p> <p>HP-UX: Requires either root authority or the user must be listed in the applications section of the authorization file. To grant a user authorization, see “Authorizing HP-UX users to ARM services” on page 126</p> <p>i5/OS: Requires that the application runs under a user that has been granted authority to use the QSYS2/LIBARM4 service program. Or, the application must be owned by a user that has the required authority so that it can adopt the authority of the owner.</p> <p>Linux: Requires root authority.</p> <p>Solaris: Requires either root authority or the user must be listed in the applications section of the authorization file. To grant a user authorization, see “Authorizing Solaris users to ARM services” on page 156.</p> <p>Windows: Requires that the application run under the Local System Account, or under a user that belongs to either the Administrators group or the EWLMArm4Users group. It is recommended that you use the EWLMArm4Users account to help avoid security exposures.</p> <p>z/OS: The user ID’s address space must be permitted to resource BPX.WLMSEVER in the FACILITY class, if the BPX.WLMSEVER resource is defined. If BPX.WLMSEVER is not defined, the calling process must be defined as a superuser (UID=0). For more information, see “Authorizing z/OS users to ARM services” on page 188.</p>	

Domain topology planning worksheet

Description	Your value / answer / % complete
<p>Obtain or create SSL certificates for securing communications between whichever of the following apply to your particular environment:</p> <ul style="list-style-type: none"> • Web browsers and the EWLM Control Center • The domain manager and its managed servers • The managed servers and firewall brokers • The firewall brokers and the domain manager • The domain manager and the load balancers <p>Record the location and password for each certificate keystore. Use file system protections to ensure the keystore location cannot be accessed by non-privileged users. You will be asked to provide this information when you configure or customize the domain manager, managed servers, and firewall brokers, if applicable.</p>	
<p>Draw a diagram of your EWLM domain; include the location of the domain manager, firewalls, load balancers, and each managed server.</p>	

Description	Your value / answer / % complete
Determine where you need to place firewall brokers in the domain. If one or more managed server uses a SOCKS server to communicate with the domain manager, you must create a firewall broker. The firewall broker is required to allow communication between these systems. List the names of the managed servers that must connect to the domain manager through a firewall broker.	
List the ARM-instrumented applications installed on each of the managed servers that EWLM is to monitor.	
Define performance objectives for work that will flow through the EWLM domain.	

Domain manager planning worksheet

- ___ Ensure the system on which you plan to install the domain manager runs a supported operating system with the latest fixes applied. For a list of supported operating systems, see Specified operating environments.
-
- ___ Check the hardware and software requirements to make sure that your system complies.
-
- ___ Ensure that the server has valid Internet Protocol (IP) version 4 addresses. IP version 4 addresses are required for communication between the domain manager and managed servers.
-
- ___ Determine if you are to use an existing LDAP server or install a new LDAP server. The domain manager installation requires that you select to use an existing LDAP server or to select to install one. For more information, see Directory server.
-

Description	Your value / answer / % complete
<i>Configuration information</i>	
Enter a name for the EWLM domain.	
Enter the IP address or host name of the domain manager. The IP address or host name must be a valid IP version 4 address or resolve to a valid IP version 4 address. This is the value for the -ma parameter on the createDM command.	
Enter the configuration ID (configID) for the domain manager. The configuration ID must be unique in EWLM. This is the value for the configID parameter on the createDM command.	
Enter the port on the domain manager that will be used for communication with the managed servers. Use SSL to protect the data that flows across this connection and to authenticate the servers. This is the value for the -mp parameter on the createDM command.	
Enter the port that the EWLM Control Center and domain manager will use to communicate. If possible, you should secure this port by blocking external access to this port and only allow the user accounts under which the EWLM Control Center and domain manager runs to access it. This is the value for the -jp parameter on the createDM command.	
Enter the first port in a range of ports that the WebSphere Application Server instance will use. The port that you specify should be the first of a contiguous range of 15 open ports. If a port in the range is not open, the configuration will fail. This is the value for the -wasPorts parameter on the createDM command. Note for z/OS the parameter is called -wasProfile .	

Description	Your value / answer / % complete
<i>Configuration information</i>	
<p>Enter which type of SSL you plan to use to protect connections coming into the domain manager, if any. The possible communication streams are listed in the prerequisite planning worksheet:</p> <ul style="list-style-type: none"> • Server SSL • Client/server SSL • None <p>If you plan to use either server SSL or client/server SSL, enter the location of the certificate keystore on the domain manager and its password.</p> <p>Note: You must use a single certificate keystore on the domain manager to store all of your SSL trusted keys for managed servers and load balancers. In addition, the type of SSL you plan to use on the domain manager must be the same as the SSL-type you use on the managed servers and firewall brokers, if applicable.</p> <p>This is the value for the -auth parameter on the createDM command.</p>	
<i>Customization information</i>	
<p>Will any of the managed servers in the EWLM domain connect to the domain manager through a SOCKS server?</p> <p>If the answer is yes, then you will need to configure the domain manager to identify the firewall broker. Gather the information requested in the next four cells.</p>	
<p>Enter the port that the domain manager will use to communicate with any firewall brokers in the EWLM domain. Use client/server SSL to protect the data that flows across this connection and to authenticate the servers. This is the port number that you will specify in the -dp parameter when you run the create firewall broker (createFB) command.</p> <p>This is the value for the -fb parameter on the changeDM command.</p>	
<p>List each of the firewall brokers in the EWLM domain. You should record the IP address or host name of each firewall broker, along with the port that this domain manager will use to communicate with the firewall broker, and whether the domain manager must access the firewall broker through a SOCKS server.</p> <p>These are the values for the -fb parameter on the changeDM command that uses the format <i>addr:port[:SOCKS] [,addr:port[:SOCKS]]</i></p>	
<p>Enter the IP address or host name of the SOCKS server that data from the domain manager must flow through to access the managed servers via the firewall broker. You only have to set this value on the domain manager if the SOCKS server is protecting the zone where the managed server and firewall broker are located. This is the value for the -sa parameter on the changeDM command.</p>	
<p>Enter the port that the domain manager will use to communicate with the SOCKS server. You only have to set this value on the domain manager if the SOCKS server is protecting the zone where the managed server and firewall broker are located. This is the value for the -sp parameter on the changeDM command.</p>	

Description	Your value / answer / % complete
<i>Configuration information</i>	
<p>Do you plan to use a load balancer to provide recommendations about how to distribute work in the EWLM domain?</p> <p>If the answer is yes, then you must configure the domain manager to identify the port on the domain manager from which it listens for communications from load balancers. Gather the applicable information requested in the next two cells.</p>	
<p>Enter the non-secure port that the domain manager will use to listen for communication from load balancers. This is the value for the -lbp parameter on the changeDM command.</p>	
<p>Enter the secure port that the domain manager will use to listen for communication from load balancers.</p> <p>Note: You could potentially have both the non-SSL and SSL ports active if you have multiple load balancers in your environment.</p> <p>This is the value for the -lbs parameter on the changeDM command.</p>	
<p>Do you want to enable EWLM partition management? If so, identify the partitions that EWLM is to manage and to which partition workload group each belongs.</p>	

Managed server planning worksheet

- ___ Ensure any system on which you plan to install a managed server is running on a supported operating system with the latest fixes applied.

- ___ Check the hardware and software requirements to make sure that your system complies.

- ___ Ensure that the server has a valid Internet Protocol (IP) version 4 address.

- ___ Ensure that the path specified in the TEMP environment variable has at least 100 MB of disk space available to complete the installation, except HP-UX which requires 250 MB.

- ___ (HP-UX, Solaris, and Windows) Verify the date and time that you will perform the installation because the install requires a restart of the system. Notify users of the system that it will be down temporarily.

- ___ Determine how you will distribute and install the EWLM managed servers to endpoint systems. You can use one of the following options:
 - **IBM Director:** This method is available with the Virtualization Engine; however, you must install and configure IBM Director separate from EWLM. Allows you to remotely deploy and install the EWLM managed server to multiple endpoint systems simultaneously from a central location.
 - **Remote package deployer:** This method is available with the Virtualization Engine and allows you to remotely deploy and install the EWLM managed server from a central location to one endpoint system at a time.
 - **EWLM manual install:** This method is available with EWLM and allows you to install the EWLM managed server to one endpoint system at a time. You must distribute the installation image to each endpoint system, and then install the managed server. This installation method supports interactive, silent, or console modes.

Description	Your value / answer / % complete
<i>Configuration information</i>	
<p>Enter the IP address or host name of the domain manager to which this managed server will connect. This should be the same IP address or host name that you specified in the domain manager planning worksheet. If you need to have a firewall broker between the domain manager and the managed server, then this will be the IP address or host name you specify in the firewall broker planning worksheet. This is the value for the <code>-ma</code> parameter on the <code>createMS</code> command.</p>	
<p>Enter the configuration ID (<i>configID</i>) for the managed server. The configuration ID must be unique within the system on which you install the EWLM managed server. This is the value for the configID parameter on the <code>createMS</code> command.</p>	
<p>Enter the port that the domain manager listens for managed server communications. This should be the same port you specified in the domain manager planning worksheet. If you need to have a firewall broker between the domain manager and the managed server, then this will be the IP address or host name you specify in the firewall broker planning worksheet. This is the value for the -mp parameter on the <code>createMS</code> command.</p>	
<p>Enter which type of SSL you plan to use to protect connections to either a firewall broker or the domain manager, whichever applies.</p> <ul style="list-style-type: none"> • Server SSL • Client/server SSL • None <p>If you plan to use either server SSL or client/server SSL, enter the location of the certificate keystore on this managed server and its password.</p> <p>Note: You must use the same type of SSL on the domain manager and firewall broker that you plan to use on this managed server.</p> <p>This is the value for the -auth parameter on the <code>createMS</code> command.</p>	
<p>If EWLM is to manage the resources of the managed server and the managed server is a partition, specify to which Partition workload group it belongs.</p>	
<i>Customization information</i>	
<p>Note: In an EWLM environment, a managed server can access its domain manager through either an HTTP proxy server or a SOCKS server via a firewall broker. The managed server can never access the domain manager through both. Answer only the questions in the customization information section that apply to your particular situation.</p>	
<p>Will this managed server access the domain manager through an HTTP proxy server?</p> <p>If the answer is yes, then you will need to configure the managed server to identify the proxy server. Gather the information requested in the next two cells.</p>	
<p>Enter the IP address or host name of the HTTP proxy server. This is the value for the -va parameter on the <code>changeMS</code> command.</p>	
<p>Enter the port that this HTTP proxy server uses to listen for managed server communications. This is the value for the -vp parameter on the <code>changeMS</code> command.</p>	

Description	Your value / answer / % complete
<i>Configuration information</i>	
<p>Will this managed server access the domain manager through a SOCKS server via a firewall broker?</p> <p>If the answer is yes, then you will need to configure the managed server to identify the firewall broker which is required to allow the managed server to access the domain manager through the SOCKS server. Gather the information requested in the next two cells.</p>	
Enter the IP address or host name of the firewall broker. This is the value for the -ma parameter on the createMS command.	
Enter the port that this firewall broker uses to listen for managed server communications. This is the value for the -mp parameter on the createMS command.	

Firewall broker planning worksheet

- Determine if you need to create a firewall broker. Firewall broker support is made available to you when you install the EWLM managed server. Whether or not you deploy the firewall broker in your EWLM domain depends on your particular environment. See “Accessing the domain manager through a firewall” on page 198 for more information about firewall brokers, when to use one, and how to set it up.

Description	Your value / answer / % complete
<i>Configuration information</i>	
Enter the IP address or host name of the firewall broker. This is the value for the -ma parameter on the createFB command.	
Enter the configuration ID (<i>configID</i>) for the firewall broker. The configuration ID must be unique within the system on which you create the firewall broker. This is the value for the configID parameter on the createFB command.	
Enter the port that the firewall broker uses to listen for managed server communications. This is the value for the -mp parameter on the createFB command.	
Enter the port that the firewall broker uses to listen for domain manager communications. This is the value for the -fp parameter on the createFB command.	
Enter the IP address or host name of the domain manager. This should be the same address you specified on the domain manager planning worksheet. This is the value for the -da parameter on the createFB command.	
Enter the port that the domain manager uses to listen for firewall broker communications. This should be the same port you specified on the domain manager planning worksheet. This is the value for the -dp parameter on the createFB command.	

Description	Your value / answer / % complete
<i>Configuration information</i>	
Enter which type of SSL you plan to use to protect connections in your EWLM domain. <ul style="list-style-type: none"> • Server SSL • Client/server SSL • None If you plan to use either server SSL or client/server SSL, enter the location of the certificate keystore on the firewall broker and its password. Note: You must use the same type of SSL on the domain manager and managed servers that you plan to use on this firewall broker. This is the value for the -auth parameter on the createFB command.	
<i>Customization information</i>	
Enter the IP address or host name of the SOCKS server that the managed servers must pass through to access the firewall broker. This is the value for the -sa parameter on the changeFB command.	
Enter the port on which the SOCKS server listens for connections. This is the value for the -sp parameter on the createFB command.	

Installing EWLM domain manager

View this topic to learn how to install EWLM.

Depending on which Virtualization Engine package and operating system you have purchased, you can install an EWLM domain manager, EWLM managed servers for IBM platforms, EWLM managed servers for non-IBM platforms, or some combination of these. The installation suite for each operating system includes the EWLM domain manager, and installation images for either the IBM managed servers, or the non-IBM managed servers, depending on which package you choose.

The IBM Virtualization Engine EWLM packages available through the Virtualization Engine installation wizard include the following:

Domain manager packages

- EWLM domain manager for AIX
- EWLM domain manager for i5/OS
- | • EWLM domain manager for Linux on POWER
- EWLM domain manager for Linux on xSeries
- | • EWLM domain manager for Linux on zSeries
- EWLM domain manager for Windows on xSeries

Note: The EWLM Control Center is installed as part of the domain manager.

As part of the EWLM domain manager installation, you must select which EWLM managed server packages that you will also need in your EWLM environment. The EWLM managed server packages include the following:

Managed server packages

- | • EWLM managed servers for AIX, HP-UX, i5/OS, and Linux
- | • EWLM managed servers for Windows and Solaris

Note: The managed server installation image includes an optionally installable firewall broker installation image.

Note: z/OS: To install either the EWLM domain manager or EWLM managed server on a z/OS operating system, see “Installing EWLM on z/OS” on page 94.

To install the EWLM domain manager, follow these steps:

1. Ensure that your systems meet the minimum EWLM software and hardware requirements for the domain manager and managed servers.
2. Complete the EWLM planning worksheets. The planning worksheets provide a list of tasks that you must complete before installing and configuring EWLM.
3. Review the Virtualization Engine installation notes for the operating system on which you will install the EWLM domain manager:
 - Installing the Virtualization Engine: AIX
 - Installing the Virtualization Engine: i5/OS
 - Installing the Virtualization Engine: Linux
 - Installing the Virtualization Engine: Windows
 - Installing the Virtualization Engine: z/OS
4. Launch and complete the Virtualization Engine installation wizard to install the EWLM domain manager and the managed server installation images on your system. Use the extensive online help system as necessary to answer questions you might have as you install EWLM.

When you use the Virtualization Engine installation wizard and you select **IBM Enterprise Workload Manager**, the entry expands and reveals the following preselected subcomponents. If you do not want a particular subcomponent, you can deselect it:

- **EWLM agents for AIX, i5/OS, Linux, HP-UX**
 - Select and expand this to copy installation images of the EWLM managed servers on the local system. After you complete the wizard, you will need to distribute, install, and configure these images on the managed servers.
- **EWLM agents for Windows and Solaris**
 - Select and expand this to copy installation images of the EWLM managed servers on the local system. After you complete the wizard, you will need to distribute, install, and configure these images on the managed servers.
- **IBM Enterprise Workload Manager Domain Manager**
 - Select and expand this to install the domain manager and EWLM Control Center on your system.
- **IBM Enterprise Workload Manager Domain Manager bridge to IBM Virtualization Engine console**
 - Select this if you plan to install the Virtualization Engine console now or in the future. The EWLM bridge to IBM Virtualization Engine console is software that enables the Virtualization Engine console to communicate with the EWLM domain manager.
For more information on how to install, configure, and use the Virtualization Engine console bridge, see Virtualization Engine console installation and configuration.

5. Install the EWLM managed server onto endpoint systems. For more information on how to install the EWLM managed server, see the following subtopics:
 - AIX, HP-UX, i5/OS, Linux, Solaris, Windows: “Installing the EWLM managed servers” on page 95
 - z/OS: “Installing EWLM on z/OS” on page 94
6. Ensure that you have installed the latest Virtualization Engine fix packages.
 - Ensure that you adhere to the EWLM prerequisite, installation, and post-installation notes before installing a fix package. For more information, see “Fix package installation considerations for EWLM” on page 102.

7. After you finish installing EWLM, you must configure the components of EWLM that you installed. You must configure the domain manager and managed servers. The specific configuration tasks vary from one operating system to another. Ensure that you complete the configuration tasks for the operating system on which the domain manager is installed and for the operating systems that are EWLM managed servers.

Related tasks

“Configuring EWLM” on page 105

View this topic to learn how to configure EWLM. You must configure the components of EWLM that exist in your EWLM environment before you can begin to use EWLM. For example, you must configure the EWLM domain manager and configure each EWLM managed server in your domain before you can start EWLM.

Command usage considerations

View this topic to learn about the syntax required when issuing an EWLM command. As you complete the EWLM installation and configuration tasks, you may need to run an EWLM command. When you specify an EWLM command, the location from which you run the command and the syntax used to format the command varies by operating system. In addition, the commands that are supported by each operating system vary.

The following is a list of EWLM configuration commands that are available, depending on which functions of EWLM you install. The operating system usage notes indicate if a particular operating system supports a particular command and special considerations when issuing the commands.

- **Domain manager:** The installation path includes the following:

- configWizardDM
- createDM
- changeDM
- displayDM
- startDM
- stopDM
- startWAS
- stopWAS

- **Managed server and firewall broker:** The installation path includes the following:

- configWizardMS
- createMS
- changeMS
- displayMS
- startMS
- stopMS

If you select to install the firewall broker the installation path also includes the following commands:

- createFB
- changeFB
- displayFB
- startFB
- stopFB

Operating system usage notes

- AIX

- | – If you run the command from */bin* location where the command resides, prefix the command with *./* unless you have *."* listed in the PATH environment variable. If you run the command from another directory, you must provide the absolute or relative path to the command that you want to use.
- | – Add the extension *.sh* to the command.
- | – Run the command by either changing to the directory where the command resides or by passing the relative or absolute path to the directory where the command resides. The default installation location of the EWLM commands are as follows. If you installed to a different directory, then go to the */bin* subdirectory of that directory.
 - | - **Domain manager:** */opt/IBM/VE2/EWLM/bin*
 - | - **Managed server and firewall broker:** */opt/IBM/VE2/EWLMMS/bin*
- | • **HP-UX**
 - | – If you run the command from */bin* location where the command resides, prefix the command with *./* unless you have *."* listed in the PATH environment variable. If you run the command from another directory, you must provide the absolute or relative path to the command that you want to use.
 - | – Add the extension *.sh* to the command.
 - | – Run the command by either changing to the directory where the command resides or by passing the relative or absolute path to the directory where the command resides. The default installation location of the EWLM commands are as follows. If you installed to a different directory, then go to the */bin* subdirectory of that directory.
 - | - **Domain manager:** The domain manager is not supported by this operating system.
 - | - **Managed server and firewall broker:** */opt/IBM/VE2/EWLMMS/bin*
- | • **i5/OS**
 - | – If you run the command from */bin* location where the command resides, prefix the command with *./* unless you have *."* listed in the PATH environment variable. If you run the command from another directory, you must provide the absolute or relative path to the command that you want to use.
 - | – Add the extension *.sh* to the command (from a QSHELL session).
 - | – Run the command by either changing to the directory where the command resides or by passing the relative or absolute path to the directory where the command resides. The default installation location of the EWLM commands are as follows. If you installed to a different directory, then go to the */bin* subdirectory of that directory.
 - | - **Domain manager:** */QIBM/ProdData/VE2/EWLM/bin*
 - | - **Managed server and firewall broker:** */QIBM/ProdData/VE2/EWLMMS/bin*
- | • **Linux**
 - | – If you run the command from */bin* location where the command resides, prefix the command with *./* unless you have *."* listed in the PATH environment variable. If you run the command from another directory, you must provide the absolute or relative path to the command that you want to use.
 - | – Add the extension *.sh* to the command.
 - | – Run the command by either changing to the directory where the command resides or by passing the relative or absolute path to the directory where the command resides. The default installation location of the EWLM commands are as follows. If you installed to a different directory, then go to the */bin* subdirectory of that directory.
 - | - **Domain manager:** */opt/IBM/VE2/EWLM/bin*
 - | - **Managed server and firewall broker:** */opt/IBM/VE2/EWLMMS/bin*
- | • **Solaris**

- | – If you run the command from */bin* location where the command resides, prefix the command with *./* unless you have *."* listed in the PATH environment variable. If you run the command from another directory, you must provide the absolute or relative path to the command that you want to use.
- | – Add the extension *.sh* to the command.
- | – Run the command by either changing to the directory where the command resides or by passing the relative or absolute path to the directory where the command resides. The default installation location of the EWLM commands are as follows. If you installed to a different directory, then go to the */bin* subdirectory of that directory.
 - | - **Domain manager:** The domain manager is not supported by this operating system.
 - | - **Managed server and firewall broker:** */opt/IBM/VE2/EWLMMS/bin*

- | • **Windows**

- | – Add the extension *.bat* to the command.
- | – Run the command by either changing to the directory where the command resides or by passing the relative or absolute path to the directory where the command resides. The default installation location of the EWLM commands are as follows. If you installed to a different directory, then go to the */bin* subdirectory of that directory.
 - | - **Domain manager:** *C:\Program Files\IBM\VE2\EWLM\bin*
 - | - **Managed server and firewall broker:** *C:\Program Files\IBM\VE2\EWLMMS\bin*
- | – When installing the managed server, you can either double click the installation image file or go to **Start** → **Run** *C:\temp\EWLMMSWin.exe* where *C:\temp* is the path to which you copied the installation image from the domain manager to the target system that contains EWLMMSWin.exe .

- | • **z/OS**

| When entering a command from a z/OS Unix Systems Services command line:

- | – If you run the command from */bin* location where the command resides, prefix the command with *./* unless you have *."* listed in the PATH environment variable. If you run the command from another directory, you must provide the absolute or relative path to the command that you want to use.
- | – Add the extension *.sh* to the command.
- | – To run the *createdM*, *changeCC*, *changeCCapp*, *displayCC*, or *stopWAS* commands, you must be logged on with the user ID defined in the *configWAS* command as the ID for the WebSphere Application Server instance.
- | – Run the command by either changing to the directory where the command resides or by passing the relative or absolute path to the directory where the command resides. The default installation location of the EWLM commands are as follows. If you installed to a different directory, then go to the */bin* or *scripts*, depending on what command you want to use, subdirectory of that directory.
 - | - **Virtualization Engine global configuration:** */usr/lpp/VE_R2/VE_LIB/scripts*
 - | - **Virtualization Engine WebSphere Application Server:** */usr/lpp/VE_R2/VEWAS/scripts*
 - | - **Domain manager:** */usr/lpp/VE_R2/EWLM/bin*
 - | - **Managed server and firewall broker:** */usr/lpp/VE_R2/EWLMMS/bin*
- | – The following commands are not supported: *configWizardDM*, *configWizardMS*, *stopDM*, *stopFB* and *stopMS*.

- | **Related tasks**

| “Configuring EWLM” on page 105

| View this topic to learn how to configure EWLM. You must configure the components of EWLM that exist in your EWLM environment before you can begin to use EWLM. For example, you must configure the EWLM domain manager and configure each EWLM managed server in your domain before you can start EWLM.



Installing EWLM on z/OS

View this topic to learn how to install EWLM on z/OS.

Depending on which Virtualization Engine package and operating system you have purchased, you can install an EWLM domain manager, EWLM managed servers for IBM platforms, EWLM managed servers for non-IBM platforms, or some combination of these. Refer to the planning advisor for guidance about which package or packages you will need for your particular environment.

IBM Virtualization Engine Enterprise Workload Manager for z/OS, which is a separately-orderable package, includes an EWLM domain manager and managed server. The EWLM Control Center is installed as part of the domain manager installation. The managed server installation image includes a firewall broker installation image.

To install EWLM on z/OS, follow these steps:

1. Ensure that your system meets the minimum software requirements
2. Complete the EWLM planning worksheets. The planning worksheets provide a list of tasks that you must complete before installing and configuring EWLM.
3. Install Virtualization Engine Enterprise Workload Manager onto your z/OS operating system. If you obtain Virtualization Engine Enterprise Workload Manager for z/OS in a custom-built product delivery offering (CBPDO, 5751-CS3), use the installation instructions contained in the program directory that is provided with the order. If you obtained the Virtualization Engine Enterprise Workload Manager for z/OS with SystemPac® or ServerPac, use the installation jobs and documentation supplied with those offerings.
4. Ensure that you have installed the latest service for Virtualization Engine Enterprise Workload Manager. Contact the IBM Support Center, use S/390® SoftwareXcel to obtain the current preventive service planning (PSP) "bucket," or obtain the current PSP from the Internet at PSP390  (<https://techsupport.services.ibm.com/server/390.psp390>). For program support, see Software Support  (<http://www.ibm.com/software/support/>). The PSP bucket UPGRADE is VEV2R1, and the specific subset information is described in the program directory provided with the order.
5. After you finish installing EWLM, you must configure the components of EWLM that you installed. You must configure the domain manager and managed servers. The specific configuration tasks vary from one operating system to another. Ensure that you complete the configuration tasks for the operating system on which the domain manager is installed and for the operating systems that are EWLM managed servers.

Managed servers

Note these differences between Installing EWLM on z/OS and installing EWLM on other operating systems:

- You do not copy and deploy the managed server installation images from z/OS onto other endpoint systems. Instead, to install managed servers on other operating systems, follow the procedure described in Installing EWLM
- You do not deploy a managed server installation image from another operating system to z/OS. Instead, to install a manager server on z/OS, you install Virtualization Engine Enterprise Workload Manager for z/OS as described here.

Related tasks

"Installing the EWLM managed servers" on page 95

During the EWLM domain manager install, the EWLM managed server installation images are copied to the domain manager file system. To finish the managed server install, you must copy and deploy the managed server installation images onto the endpoint system that will be an EWLM managed server.

Related information

Installing the EWLM managed servers

During the EWLM domain manager install, the EWLM managed server installation images are copied to the domain manager file system. To finish the managed server install, you must copy and deploy the managed server installation images onto the endpoint system that will be an EWLM managed server.

1. Install EWLM onto the endpoint system that is to be an EWLM managed server. To install the EWLM managed server, select one of the following methods:
 - a. "Using IBM Director for managed server installation"
 - b. "Using Remote Package Deployer for managed server installation" on page 96
 - c. "Manually distributing and installing the managed server" on page 97
2. AIX: Load the netWLM fileset on each endpoint system. For more information, see "AIX: Loading the netWLM fileset" on page 101.
3. HP-UX: Install the LSOF utility for load balancing support. For more information, see "HP-UX: Installing LSOF utility for load balancing support" on page 102.
4. Solaris: Install the LSOF utility for load balancing support. For more information, see "Solaris: Installing LSOF utility for load balancing support" on page 102.
5. Repeat this procedure for each system that you want to be an EWLM managed server.

Related tasks

"Installing EWLM on z/OS" on page 94

View this topic to learn how to install EWLM on z/OS.

"Fix package installation considerations for EWLM" on page 102

Using IBM Director for managed server installation

View this topic to learn how to use IBM Director to distribute the appropriate managed server installation images to each of the servers that will be part of the EWLM domain. IBM Director is the preferred method for distributing and installing endpoint software because it enables you to install all of the endpoint software from one centralized location. This method provides an interactive user interface to assist you during the installation. You must have IBM Director installed to use this method.

IBM Director allows you to install the EWLM managed server on multiple systems simultaneously. In addition, you can install the managed server onto different operating systems.

When you use IBM Director to install the EWLM managed server, you must identify the location of the EWLM managed server installation files. IBM Director needs these files to complete the managed server installation. These managed server installation files were copied to the domain manager's file system during the domain manager installation. IBM Director will copy the installation files from the domain manager's file system and install the managed server onto the systems that you identify as endpoint systems.

IBM Director uses the installable unit deployment descriptor (IUDD) files to install the EWLM managed server onto endpoint systems. To locate the IUDD files, use the file paths documented in the following table. Append the EWLM managed server path to the default Virtualization Engine path to get a complete path to the IUDD files.

Table 10. Virtualization Engine default installation paths and paths to IUDD files.

Operating system	Default Virtualization Engine installation paths	Path to EWLM managed server
AIX	/opt/IBM/VE2/	Managed Nodes/EWLM/ <i>operating system name</i> /FILES/Metainf where <i>operating system name</i> represents the name of the managed server operating system
i5/OS ¹	/QIBM/ProdData/VE2/	
Linux	/opt/IBM/VE2/	
Windows	C:\Program Files\IBM\VE2\	
Note:		
1. If the i5/OS endpoint system is a Level-2 managed system, which means that the system contains the IBM Director Agent, IBM Director on the endpoint system must run under a user profile that has a password. The default user profile does not have a password. If the user profile does not have a password, the EWLM managed server install will fail. For more information, see Authorizing users for i5/OS in the IBM Director topic collection.		

Use the following information to distribute Virtualization Engine software to your endpoint systems using IBM Director:

1. Creating software packages to distribute

Use the IBM Update Assistant to create the software packages for distribution to the endpoint systems.

2. Distributing a software package

After you create the software package, use these instructions to distribute the software to the endpoint systems.

Using Remote Package Deployer for managed server installation

View this topic to learn how to use Remote Package Deployer to distribute the appropriate managed server installation images to each of the servers that will be part of the EWLM domain. This method provides an interactive user interface to assist you during the installation. The Remote Package Deployer is shipped with the Virtualization Engine and is installed when you install the EWLM domain manager. No additional setup is required.

Remote package deployer allows you to remotely distribute and install the EWLM managed server on endpoint systems from a central location. This managed server installation method requires that you distribute and install each managed server individually.

When you use remote package deployer to install the EWLM managed server, you must identify the location of the EWLM managed server installation files. Remote package deployer needs these files to complete the managed server installation. These managed server installation files are copied to the domain manager's file system during the domain manager installation. Remote package deployer will copy the installation files from the domain manager's file system and install the managed server onto the systems that you identify as endpoint systems.

The remote package deployer uses the IUDD files to install the EWLM managed server onto endpoint systems. To locate the IUDD files, use the file paths documented in the following table. Append the EWLM managed server path to the default Virtualization Engine path to get a complete path to the IUDD files.

Table 11. Virtualization Engine default installation paths and paths to IUDD files

Operating system	Default Virtualization Engine installation paths	Path to managed server IUDD files
AIX	/opt/IBM/VE2/	Managed Nodes/EWLM/ <i>operating system name</i> /FILES/MetaInf where <i>operating system name</i> represents the name of the managed server operating system.
i5/OS	/QIBM/ProdData/VE2/	
Linux	/opt/IBM/VE2/	
Windows	C:\Program Files\IBM\VE2\	

For step-by-step instructions on how to use the remote package deployer, see Distributing agent software without IBM Director.

Manually distributing and installing the managed server

View this topic to learn how you can manually distribute and install the managed server onto the system that will be a part of the EWLM domain. Select this method if you do not want to use the Remote Package Deployer or IBM Director methods to complete the managed server installation process.

The process of installing the domain manager also makes available the managed server installation images. To distribute these managed server installation images to endpoint systems, complete the following steps.

1. Navigate to the Virtualization Engine installation directory in the domain manager file system.

The default installation paths are the following:

- **AIX, Linux:** /opt/IBM/VE2/
- **i5/OS :** /QIBM/ProdData/VE2/
- **Windows:** C:\Program Files\IBM\VE2\

2. Navigate to the managed server installation image files that are within the installation directory. From the installation directory, select **Managed Nodes** → **EWLM** → ***operating system*** → **FILES**, where the variable is defined as follows:

- ***operating system*:** Refers to the directory that is specific to the operating system in which you want to install the EWLM managed server. The following is a list of possible directories that vary depending on what you selected to install during the Virtualization Engine installation.
 - **Aix:** AIX
 - **Hpux:** HP-UX
 - **i5OS:** i5/OS
 - **LinPPC:** Linux on POWER
 - **Linx:** Linux on xSeries
 - **Linz:** Linux on zSeries
 - **Solaris:** Solaris
 - **Win:** Windows

3. Copy and distribute the appropriate managed server installation images and response time files to each of the servers that will be a part of the EWLM domain. You can distribute the files by File Transfer Protocol (FTP), by mapping a drive and copying the files to the file system, or by any other distribution mechanism that you might have in place. You must distribute the installation image to a system that runs the corresponding operating system. For example, EWLMMSAix.bin must be distributed to a system that runs the AIX operating system. The following list identifies the name of the installation image file for each operating system. The list of installation images available to you to install on endpoint systems depends the EWLM managed servers that you have purchased. Each installation image has a corresponding response time file that has the *.rsp* file extension. This response time file will be useful when you install the managed server in that it includes default values for the install.

| You can identify which image goes with which platform by the name of the installable file, as follows:

- | • AIX: EWLMMSAix.bin
- | • HP-UX: EWLMMShpux.bin
- | • i5/OS: EWLMMSi5OS.exe or EWLMMSi5OS.jar

| **Note:** If you use a Windows system to install the i5/OS managed server remotely, use the .exe file.
| If you install the i5/OS managed server locally, use the .jar file.

- | • Linux on POWER: EWLMMSLinPPC.bin
- | • Linux on xSeries: EWLMMSLinx.bin
- | • Linux on zSeries: EWLMMSLinz.bin
- | • Solaris - EWLMMSSolaris.bin
- | • Windows: EWLMMSSwin.exe

| **Note:** If you plan to use FTP to transfer the images to a system running AIX, HP-UX, Linux, or
| Solaris, note that the FTP method does not maintain the executable bit on the installation image
| file when it is sent to either of these operating systems. You can still use FTP to transfer the
| files to the endpoint system; however, after you do so, you must run `chmod 755` on the
| installation image to set the permission levels so that you can continue the installation process.

After you distribute the installation images onto the systems that will be a part of the domain, you must install the EWLM managed server onto the endpoint system.

Manually install managed server using interactive mode:

View this topic to learn how to complete the managed server installation process installing the EWLM managed servers on the endpoint systems. This process describes the interactive method that provides a graphical user interface.

After you distribute the managed server to the endpoint system, complete the following to install the EWLM managed server.

To install the EWLM managed server by using the installation wizard in **interactive mode**, complete the following:

1. From a command line, go to the directory where you copied the installation image and run the installation.
2. To start the installation, run the installation command. The command is the name of the installation image file name. To view a list of the installation image file names for each operating system, see installation image file names. Ensure that when you specify the install command that you adhere to the syntax requirements as described in Command usage considerations.
3. Complete the Install wizard to install the managed server on the target system.
4. HP-UX, Solaris, Windows only: Restart the system.

Manually install managed server using silent mode:

View this topic to learn how to complete the managed server installation process by installing the EWLM managed servers on the endpoint systems. This process describes the silent method that does not require a graphical user interface.

After you distribute the managed server to the endpoint system, complete the following procedure to install the EWLM managed server.

To install the managed servers in **silent mode**, complete the following steps:

1. From a command line, go to the directory where you copied the installation image.

2. Start the managed server installation by completing either of the following steps:
 - a. Run the following command that uses the installation parameters as defined in the response time file. Edit this file as necessary before issuing this command. The response time file includes default installation values.

```
EWLM_MSfilename [i5OSSystemName i5OSUserID i5OSpwd] -silent -options EWLM_MSfilename.rsp
```

where the possible value is defined as follows:

- *EWLM_MSfilename* is the file name of the installation image for the appropriate platform. To view a list of the installation image file names for each operating system, see installation image filenames.
- *i5OSSystemName* is the name of the i5/OS endpoint system on which you are installing the managed server function. This is only required for i5/OS remote installations.
- *i5OSUserID* is the name of the i5/OS user ID who has the proper user authorities to perform the installation. This is only required for i5/OS remote installations.
- *i5OSpwd* is the password for the i5/OS user ID. This is only required for i5/OS remote installations.

With the response time file, you can install the managed server without having to pass parameters as arguments to the command.

- b. Run the following command that allows you to specify installation parameters within the command rather than using the response time file.

```
EWLM_MSfilename -silent -P managedServerBean.active=[true | false]
-P firewallBrokerBean.active=[true | false]
-P productBean.installLocation=mydirectory
[i5OSSystemName i5OSUserID i5OSpwd]
```

where each of the possible parameters is described as follows:

- *EWLM_MSfilename* is the file name of the installation image for the appropriate platform. To view a list of the installation image file names for each operating system, see installation image filenames.
- **managedServerBean.active** indicates whether to install the managed server or not. Possible values are **true**, to install the managed server, or **false**, to not install the managed server. The default value is false.
- **firewallBrokerBean.active** indicates whether to install the firewall broker or not. Possible values are **true**, to install the firewall broker, or **false**, to not install the firewall broker. The default value is false.
- **productBean.installLocation** indicates the directory into which you want the managed server or firewall broker installed. The default directory is \$D(install)/IBM/VE/EWLMMS.
- *i5OSSystemName* is the name of the i5/OS endpoint system on which you are installing the managed server function. This is only required for i5/OS remote installs.
- *i5OSUserID* is the name of the i5/OS user ID who has the proper user authorities to perform the install. This is only required for i5/OS remote installs.
- *i5OSpwd* is the password for the i5/OS user ID. This is only required for i5/OS remote installs.

3. HP-UX, Solaris, Windows: Restart the system.

Related tasks

“Windows: Managed server installation considerations” on page 100

“i5/OS: Installing i5/OS managed server on local system” on page 100

View this topic to learn how to install the i5/OS managed server on a local system, rather than from a remote system.

Manually install managed server using console mode:

View this topic to learn how to complete the managed server installation process by installing the EWLM managed servers on the endpoint systems. This process describes the console method that allows you to pass installation parameters, but does not require a graphical user interface.

After you distribute the managed server to the endpoint system, complete the following to install the EWLM managed server.

To install the managed servers in **console mode**, do the following:

1. From a command line, go to the directory where you copied the installation image.
2. Run the following command:

```
EWLM_MSfilename -console
```

where the possible value is defined as follows:

- *EWLM_MSfilename* is the file name of the installation image for the appropriate platform. To view a list of the installation image file names for each operating system, see installation image filenames.

3. HP-UX, Solaris, Windows only: Restart the system.
“Windows: Managed server installation considerations”

i5/OS: Installing i5/OS managed server on local system:

View this topic to learn how to install the i5/OS managed server on a local system, rather than from a remote system.

To install the managed servers **locally** on an i5/OS system, do the following:

1. From a 5250 command line, go to the directory where you copied the installation image. The i5/OS installation image filename is EWLMMSi5OS.jar.
2. Install the managed server using either a *silent* mode install or *console* mode install.
 - a. Run the following command to install the i5/OS managed server using *silent* mode
> java **-Dos400.awt.native=true -cp ./EWLMMSi5OS.jar run -silent [-options ./EWLMMSi5OS.rsp]**

where **-options** is optional. Only specify this parameter if you want to use the values specified in the response file during the installation. Edit the response file as needed before issuing the installation command.

- b. Run the following command to install the i5/OS managed server using *console* mode.
> java **-cp ./EWLMMSi5OS.jar run -console**

Related tasks

“Manually install managed server using silent mode” on page 98

View this topic to learn how to complete the managed server installation process by installing the EWLM managed servers on the endpoint systems. This process describes the silent method that does not require a graphical user interface.

Windows: Managed server installation considerations:

Special considerations for unattended installations

To perform an unattended installation of a managed server on a system running Windows, follow these steps:

1. Create a batch file to run the managed server executable after the installation. The batch file is necessary because the switches prevent the unattended installation from completing if it is placed in the unattend.txt file or as a parameter of the /cmd switch because there are too many switches. The file will look similar to the following:

```
start $D(install)\EWLMMSWin.exe -silent -P managedServerBean.active=[true | false]
-P productBean.installLocation=mydirectory
/wait
```

where *\$D(install)* maps to the platform-specific default directory in which products are typically installed. For example, C:\Program Files and *mydirectory* is the full path to where the Managed Server component is to be installed. You should also store the batch file in this directory.

2. Create the unattend.txt file:

- a. Use **Setup Manager**, which can be found in the \Support\Tools\Deploy.cab directory in the operating system installation CD.
- b. Modify the file to include the following parameters in the specified sections:

```
[Unattended]
    UnattendMode=FullUnattended
    OemPreinstall=Yes
    DriverSigningPolicy=Ignore

[GuiRunOnce]
    Command0="mydirectory\%BATCH_FILE%"
```

where *%BATCH_FILE%* is the batch file that was created in step one and *mydirectory* is the location of that batch file.

3. Call the installation executable for the operating system by using the unattended switch that points to the unattend.txt file you created in step two. The function call can be similar to the following:

```
%EXECUTABLE_LOCATION%\I386\winnt32 /s: %EXECUTABLE_LOCATION%\I386 /unattend:%ANSWERFILE%
```

where *%EXECUTABLE_LOCATION%* is the location of the operating system executable, and *%ANSWERFILE%* is the path and the name of the unattend.txt file.

Related tasks

“Manually install managed server using silent mode” on page 98

View this topic to learn how to complete the managed server installation process by installing the EWLM managed servers on the endpoint systems. This process describes the silent method that does not require a graphical user interface.

AIX: Loading the netWLM fileset

Use this procedure to install the netWLM fileset so that you can start a managed server on AIX. The netWLMfileset contains the network load balancing kernel extensions for AIX platform support.

If you do not install this fileset, when you attempt to start the managed server, you will receive the following error message:

```
EWLM execution error: The native Library ewlmnet could not be loaded
```

To check whether the netWLM fileset is already installed on your system, run this command:

```
ls|pp -l|grep ewlm
```

if the fileset is installed you should see something similar to the following:

```
...
bos.net.ewlm.rte 5.2.0.0 COMMITTED netWLM
ewlm.common 1.1.0.0 COMMITTED EWLM Common AIX Support
...
```

Prerequisite:

Ensure that the following file sets are installed before you install the netWLM fileset:

```
bos.rte 5.2.0.30
bos.net.tcp.client 5.2.0.30
```


Install the netWLM fileset:

1. Mount the AIX base operating system CD1 that is at least Version 5.2.2.
2. From the command line, enter:
`smitty install_all`
3. Enter the directory of your installation device, which in this example is the CD, similar to the following:
`/dev/cd0`
4. Press F4 to view or list the table of contents.
5. Use the forward slash (/) to search for netWLM.
6. Press F7 to select netWLM.
7. Press **Enter**. In the SOFTWARE to install field, you should see netWLM.
8. Press **Enter** to install.

HP-UX: Installing LSOF utility for load balancing support

View this topic to learn how to obtain the LSOF utility which is required by certain operating systems if you want to use EWLM load balancing capabilities.


Complete the following to install the LSOF utility:

1. Download and install the LSOF utility for the operating system that runs on the managed server. To do this, go to the following Web site:
 - Hewlett-Packard Open Source Tools List  (http://h21007.www2.hp.com/dspp/tech/tech_TechSoftwareDetailPage_IDX/1,1703,4682,00.html).
2. Download the LSOF download for the version of the operating system that you are using.
3. Complete the LSOF installation as directed. When installing LSOF, note that LSOF must have root authority. In addition, the path to the LSOF executable must be in the PATH environment variable for the root user ID that will start the managed server.

Solaris: Installing LSOF utility for load balancing support

View this topic to learn how to obtain the LSOF utility which is required by certain operating systems if you want to use EWLM load balancing capabilities.

Complete the following to install the LSOF utility:

1. Download and install the LSOF utility for the operating system that runs on the managed server. To do this, go to the following Web site:
 - Freeware for Solaris  (<http://www.sunfreeware.com/>)
2. Download the LSOF download for the version of the operating system that you are using.
3. Complete the LSOF installation as directed. When installing LSOF, note that LSOF must have root authority. In addition, the path to the LSOF executable must be in the PATH environment variable for the root user ID that will start the managed server.

Fix package installation considerations for EWLM

When installing a fix package, whether it be a Virtualization Engine fix package for a system service or a fix package for a product that a system service is dependent on, there are some items to consider. These special considerations are specific to the Enterprise Workload Manager (EWLM) systems service.

IBM recommends installing WebSphere Application Server fixes before any Virtualization Engine fixes. Read WebSphere Application Server fix package to learn about EWLM specific considerations before installing a WebSphere Application Server fix.

EWLM fix package considerations

When installing a fix package for the Virtualization Engine Enterprise Workload Manager, consider the following:

- **Prerequisites**

1. Review the readme file for the fix package, if applicable. The readme file contains special instructions that apply to only the fix package for that fix period. The following instructions in this topic apply to any EWLM fix package and for any fix period.
2. AIX, HP-UX, Solaris: Create a directory and save the auth and limits files to that directory by specifying the following commands on the managed server:


```
mkdir /etc/EWLMSAVE
cp /etc/ewlm/auth /etc/EWLMSAVE/auth
cp /etc/ewlm/limits /etc/EWLMSAVE/limits
```
3. If you are installing a domain manager fix, ensure that EWLM on the domain manager is stopped. For more information, see “Stopping EWLM on the domain manager” on page 194.
4. If you are installing a domain manager fix, ensure that the EWLM Control Center is stopped, if applicable. For more information, see “Stopping the EWLM Control Center” on page 196.
5. If you are installing a managed server fix, ensure that EWLM on the managed servers and firewall brokers are stopped, whichever apply. For more information, see “Stopping EWLM on the managed server” on page 194 and “Stopping the firewall broker” on page 195.
6. If you are installing a managed server fix, ensure that you stop the ARM-enabled applications that run on the system.

- **Installation notes**

- If installing a domain manager fix package, ensure that you specify the WebSphere Application Server user ID and password. During the install, it prompts you to specify the user ID and password. If specified incorrectly, you are not able to access the EWLM Control Center. If this occurs, you need to perform the fix package install again and specify the correct user ID and password.
- Use the following table to help determine how you will acquire and install an EWLM fix package:

Table 12. Summary of supported tools to acquire and install EWLM domain manager or managed server fix packages.

	Managed server fixes	Domain manager fixes	Notes®
Acquisition methods			
Virtualization Engine console Updates portlet	Yes	Yes	Requires the Virtualization Engine console installed on AIX, i5/OS, or Linux.
Virtualization Engine Fixes Web site	Yes	Yes	Allows you to download the fix package. Then, you must route the fix package to the system on which you want to install it, using a tool of your choice.
Installation methods			
Update Installer	No	Yes	This tool is part of the Virtualization Engine common runtime environment that is installed as part of your EWLM domain manager installation.
Manual method	Yes	No	Allows you to launch the install in interactive, console, or silent mode.

- For detailed instructions on how to obtain and install an EWLM fix package, you must use the instructions as described in the Fixes and updates topic collection.

- **Post-install**

1. AIX, HP-UX, Solaris: Restore the auth and limits files that you saved in the prerequisite step. Specify the following commands on the managed server:

```

| cp /etc/EWLMSAVE/auth /etc/ewlm/auth
| cp /etc/EWLMSAVE/limits /etc/ewlm/limits
| 2. AIX, HP-UX, Linux, Solaris, and Windows: Restart the system. On Windows, the install prompts
| you to restart the system after the install completes.
| 3. If you are installing a fix package on a domain manager, you need to update the EWLM Control
| Center. To do this, run the following command from the directory in which the domain manager
| resides:
| On AIX, HP-UX, i5/OS, Linux, Solaris, and z/OS:
| changeCCapp.sh configID -adminUser userid
| -adminPW userpw -earLocation installPath
| On Windows:
| changeCCapp.bat configID -adminUser userid
| -adminPW userpw -earLocation installPath


```

where the parameters are defined as follows:

Parameter	Description
-adminPW	The password associated with the administrator user ID.
-adminUser	The user ID of the administrator of this WebSphere Application Server instance. This user ID must conform to the WebSphere Application Server requirements for an administrator.
configID	The configuration identifier, from the createDM command, used to identify the EWLM domain manager.
-earLocation	Specify the path to where the fix package was installed. The domain manager default installation paths are the following: The default installation paths are the following: <ul style="list-style-type: none"> • AIX, Linux: /opt/IBM/VE2/ • i5/OS : /QIBM/ProdData/VE2/ • Windows: C:\Program Files\IBM\VE2\ Use this parameter if you install a domain manager fix package and want to update the Control Center to use the latest fix package.

WebSphere Application Server

The EWLM product is dependent on the WebSphere Application Server. The WebSphere Application Server also has fix packages available. Because the EWLM product is dependent on the WebSphere Application Server, you need to install the WebSphere Application Server fix packages.

IBM recommends that you install the WebSphere Application Server fix packages before a Virtualization Engine fix package. For more information about how to obtain WebSphere Application Server fix packages, see the IBM Virtualization Engine systems services fixes  (<http://www14.software.ibm.com/webapp/set2/sas/f/VirtualizationEngine/>) Web site and select a product and fix period. Then, select the **By product** or **By operating system** view.

When installing a fix package for the WebSphere Application Server, consider the following:

- **Prerequisites**

1. Review the readme file for the WebSphere Application Server fix packages.

2. Identify all WebSphere Application Server instances created by EWLM. An instance of the WebSphere Application Server is started when you start the EWLM Control Center.
 3. Stop EWLM on the domain manager. For more information, see “Stopping EWLM on the domain manager” on page 194.
 4. Stop the EWLM Control Center. For more information, see “Stopping the EWLM Control Center” on page 196.
 5. Apply the fix package.
 6. Apply Virtualization Engine fix packages, if applicable.
- **Post-install**
 1. Start EWLM on the domain manager. To start EWLM on the domain manager, see “Starting EWLM on the systems in your domain” on page 189.
 2. Start the Control Center. To start the Control Center, see “Starting the EWLM Control Center” on page 192.

Related concepts

“Stopping EWLM” on page 194

View this topic to learn how to stop the EWLM domain manager, managed server, and firewall broker. This task describes how to stop the EWLM function that runs on the system that contains the EWLM domain manager, managed server, or firewall broker. This task does not shutdown or stop the system.

Related tasks

“Stopping the EWLM Control Center” on page 196

Use this topic to learn how to stop the EWLM Control Center. Even if you close the Web session for the EWLM Control Center, the EWLM Control Center processes are still running. To stop it completely, you must complete the task outlined in this topic.

“Starting EWLM on the systems in your domain” on page 189

View this topic to learn how to start EWLM on the components of EWLM that you have installed and configured in your EWLM domain. The components of EWLM that you can install and configure include the domain manager, managed servers, and firewall brokers.

“Starting the EWLM Control Center” on page 192

View this topic to learn how to start the EWLM Control Center.

Related information

Fixes and updates

IBM Virtualization Engine systems services fixes

Configuring EWLM

View this topic to learn how to configure EWLM. You must configure the components of EWLM that exist in your EWLM environment before you can begin to use EWLM. For example, you must configure the EWLM domain manager and configure each EWLM managed server in your domain before you can start EWLM.

The specific tasks you must complete to configure EWLM vary, depending on the operating system on which EWLM is installed. Ensure that you complete the configuration tasks for each operating system that exists in your EWLM domain. For example, if you installed the EWLM domain manager on a Windows operating system, you must complete the domain manager configuration tasks described in the Windows topic collection. Similarly, if you installed the EWLM managed server on an AIX operating system, you must complete the managed server configuration tasks described in the AIX topic collection.

As you complete the EWLM configuration tasks, you can use the EWLM commands or configuration wizards, if supported by your operating system.

| To complete EWLM configuration, you will need to run EWLM commands or EWLM configuration wizards or a combination of both, depending on which methods you choose to use.

| **Related concepts**

| “Command usage considerations” on page 91

| View this topic to learn about the syntax required when issuing an EWLM command. As you complete the EWLM installation and configuration tasks, you may need to run an EWLM command. When you specify an EWLM command, the location from which you run the command and the syntax used to format the command varies by operating system. In addition, the commands that are supported by each operating system vary.

| “Managing EWLM” on page 198

| View this topic collection to work with additional tasks that you can perform when using EWLM. In addition, this topic collection describes how to enable optional functions of EWLM such as partition management and load balancing capabilities.

| **Related tasks**

| “Starting EWLM” on page 189

| After you install EWLM and configure the components of EWLM that exist in your environment, you are ready to start EWLM.

| **Configuration wizard usage considerations**

| View this topic to learn about the operating systems that support the EWLM configuration wizards. In addition, learn about the syntax required when you run an EWLM command that launches an EWLM configuration wizard. As you complete the EWLM installation and configuration tasks, you may choose to use the configuration wizards.

| The following is a list of configuration wizards, depending on which functions of EWLM you install.

| **Domain manager configuration wizard (configWizardDM):**

| Allows you to create and change the domain manager configuration. In addition, you can give users access to the EWLM Control Center. To launch this wizard, you must use the configWizardDM command.

| **Managed server configuration wizard (configWizardMS):**

| Allows you to create and change the managed server configuration. In addition, if a managed server accesses the domain manager via a SOCKS server, you can create and configure a firewall broker. To launch this wizard, you must use the configWizardMS command.

| **Operating system usage notes**

| These notes indicate which operating systems support the configuration wizards and denotes special considerations when issuing the EWLM command that launches the configuration wizard:

| **AIX**

- | • Supports the domain manager and managed server configuration wizards.
- | • If you run the command from */bin* location where the command resides, prefix the command with *./* unless you have *."* listed in the PATH environment variable. If you run the command from another directory, you must provide the absolute or relative path to the command that you want to use.
- | • Add the extension *.sh* to the command.
- | • Requires a local x-desktop environment to run the wizard as a graphical user interface (GUI).
- | • Run the command by either changing to the directory where the command resides or by passing the relative or absolute path to the directory where the command resides. The default installation location of the EWLM commands are as follows. If you installed to a different directory, then go to the */bin* subdirectory of that directory.
 - | – **Domain manager:** */opt/IBM/VE2/EWLM/bin*

- **Managed server and firewall broker:** /opt/IBM/VE2/EWLMMS/bin

HP-UX

- Supports only the managed server configuration wizard.
- If you run the command from */bin* location where the command resides, prefix the command with *./* unless you have *."* listed in the PATH environment variable. If you run the command from another directory, you must provide the absolute or relative path to the command that you want to use.
- Add the extension *.sh* to the command.
- Run the command by either changing to the directory where the command resides or by passing the relative or absolute path to the directory where the command resides. The default installation location of the EWLM commands are as follows. If you installed to a different directory, then go to the */bin* subdirectory of that directory.
 - **Domain manager:** The domain manager is not supported on this operating system.
 - **Managed server and firewall broker:** /opt/IBM/VE2/EWLMMS/bin

i5/OS

- Does not support the configuration wizards. You must use the configuration commands. For more information on issuing the EWLM commands, see “Command usage considerations” on page 91.

Linux

- Supports the domain manager and managed server configuration wizards.
- If you run the command from */bin* location where the command resides, prefix the command with *./* unless you have *."* listed in the PATH environment variable. If you run the command from another directory, you must provide the absolute or relative path to the command that you want to use.
- Add the extension *.sh* to the command.
- Requires an x-desktop environment to run the wizard as a graphical user interface (GUI).
- Run the command by either changing to the directory where the command resides or by passing the relative or absolute path to the directory where the command resides. The default installation location of the EWLM commands are as follows. If you installed to a different directory, then go to the */bin* subdirectory of that directory.
 - **Domain manager:** /opt/IBM/VE2/EWLM/bin
 - **Managed server and firewall broker:** /opt/IBM/VE2/EWLMMS/bin

Solaris

- Supports only the managed server configuration wizard.
- If you run the command from */bin* location where the command resides, prefix the command with *./* unless you have *."* listed in the PATH environment variable. If you run the command from another directory, you must provide the absolute or relative path to the command that you want to use.
- Add the extension *.sh* to the command.
- Run the scripts found in the installation path. If you installed the managed server to a different directory, run the command from the EWLMMS/bin subdirectory of that directory. See below for the default installation path.
 - **Domain manager:** The domain manager is not supported on this operating system.
 - **Managed server and firewall broker:** /opt/IBM/VE2/EWLMMS/bin

Windows

- Supports the domain manager and managed server configuration wizards.
- Add the extension *.bat* to the command.

- To launch the configuration wizard, either double click the configuration wizard file or go to **Start** → **Run** and type and run C:\Program Files\IBM\VE2\EWLM\bin\configWizardMS.bat or C:\Program Files\IBM\VE2\EWLMMS\bin\configWizardDM.bat.
- Alternatively, you can run the command to launch the wizard by either changing to the directory where the command resides or by passing the relative or absolute path to the directory where the command resides. The default installation location of the EWLM commands are as follows. If you installed to a different directory, then go to the */bin* subdirectory of that directory.
 - **Domain manager:** C:\Program Files\IBM\VE2\EWLM\bin
 - **Managed server and firewall broker:** C:\Program Files\IBM\VE2\EWLMMS\bin

z/OS

- Does not support the configuration wizards. You must use the configuration commands. For more information on issuing the EWLM commands, see “Command usage considerations” on page 91.

Configuring EWLM on AIX

View this topic to learn how to configure the components of EWLM that you installed. In addition, learn how to enable functions of EWLM that are optional.

The following topics provide step-by-step instructions on how to configure and enable various functions of EWLM. Note that not all of these tasks are required. You need to complete only the configuration tasks that are applicable to your environment.

Configuring a domain manager

View this topic to learn how to configure a domain manager.

Ensure that you have completed the domain manager worksheets found in the topic “Planning for EWLM” on page 79. Use the values from the planning worksheet when you configure the domain manager. In addition, ensure that the user responsible for configuring the domain manager has the proper user authorities.

After you install the domain manager on your system, you must run either the **EWLM domain manager configuration wizard** or the createDM configuration command to create and configure the domain manager and its associated WebSphere Application Server instance. The WebSphere Application Server instance includes the EWLM Control Center function.

Note: As part of the domain manager configuration, EWLM creates a default target user ID and password that is used to display EWLM in the Virtualization Engine (VE) Console’s Health Center. Therefore, you will notice that you are not prompted for a target user ID and password when you access EWLM in the Health Center. Due to this, ensure that you do not delete or edit this default target user ID and password that EWLM creates as part of the EWLM domain manager configuration.

You can choose to configure the domain manager as part of the installation or choose to configure the domain manager after the installation. If you need to configure the domain manager outside of the installation process, use one of the following methods.

- Run the EWLM domain manager configuration wizard
- Run createDM

Run the EWLM domain manager configuration wizard:

1. Go to the $\$D(install)/IBM/VE2/EWLM/bin$ directory where $\$D(install)$ maps to the platform-specific default directory in which products are typically installed. If you installed the domain manager to a different directory, then run the command from the EWLM/bin subdirectory of that directory.

2. Run the following command:

```
configWizardDM
```

For more information about starting the wizard on your operating system, if supported, see, "Configuration wizard usage considerations" on page 106.
3. Complete the wizard to configure the domain manager. Click **Help** if you need assistance.

Run createDM:

Run the following command, replacing the example values shown in italics with your own.

```
createDM configID -adminUser userid -adminPW password -wasPorts port -jp port
      -ma address -mp port -dn domainName -auth [None | ServerSSL | ClientServerSSL]
      -sslks path -sslpw password
```

For more information about issuing the command on your operating system, if supported, see, "Command usage considerations" on page 91.

The following table describes each of the parameters and provides a place for you to enter your specific values when you print this page for reference.

Table 13. Parameters for createDM command

Parameter	Description	Value
-adminUser	The WebSphere Application Server administrator user ID. This user ID must conform to the WebSphere Application Server requirements for an administrator.	
-adminPW	The password associated with the -adminUser parameter.	
-auth	<p>The authority level used to secure communications between the domain manager and the managed servers and, if they exist, the firewall brokers. Possible values follow:</p> <p>None Specify this to provide a default level of security between the managed servers and the domain manager. If you specify None, communication between the managed servers and the domain manager is encrypted and no application other than EWLM can communicate with the domain manager. However, it does not prevent a managed server from connecting to a domain manager to which it should not have access. As a general rule, you should not use this value in a production environment. To provide a greater level of security, you should create and use your own SSL certificates.</p> <p>ServerSSL Specify this if you are using server SSL to secure communications between the managed servers and the domain manager. Server SSL authenticates the domain manager to the managed servers and encrypts the data that flows between the servers.</p> <p>ClientServerSSL Specify this if you are using client/server SSL to secure communications between a managed server and the domain manager. Client/server SSL authenticates the domain manager to the managed server, authenticates the managed server to the domain manager, and encrypts the data that flows between the servers.</p> <p>The -auth value you specify on the createDM command must match the -auth value that you specify when you configure each managed server and firewall broker, if applicable.</p>	

Table 13. Parameters for createDM command (continued)

Parameter	Description	Value
configID	A unique name for this domain manager instance. The configuration ID must be an alphanumeric character string. Valid characters are uppercase letters A-Z, lowercase letters a-z, and numbers 0-9. The length of the ID must not exceed 32 characters.	
-dn	The name of the EWLM domain. The EWLM domain name must be a character string. All characters are valid except the colon (:), question mark (?) symbol, double quote (") symbol, greater than (>) symbol, less than (<) symbol, or bar () symbol. The length of the ID must not exceed 64 characters.	
-jp	The port that the EWLM Control Center uses to communicate with the domain manager. Ensure that this port is not currently in use and is not the same port that you specify for -mp or one of the 15 contiguous ports that -wasPorts requires.	
-ma	The IP address or host name of the domain manager server. The -ma value you specify on the createDM command must match the -ma value that you specify when you configure each managed server. This allows the managed server to properly identify and communicate with its domain manager.	
-mp	The port that the domain manager uses to listen for managed server connection requests. The -mp value that you specify on the createDM command must match the -mp value you specify when you configure each managed server.	
-sslks	The path of the certificate keystore that contains the SSL keys that are used to secure the following communications: <ul style="list-style-type: none"> Between the domain manager and the managed servers when -auth is ServerSSL or ClientServerSSL. Between the domain manager and firewall brokers, if they exist in the EWLM domain, when -auth is ServerSSL or ClientServerSSL. Between the domain manager and load balancers if you provide a port value for the -lbs parameter. Note: You set the -lbs parameter on the changedM command. Note: This parameter is only required if you specify ServerSSL or ClientServerSSL on the -auth parameter.	
-sslpw	The password used to access the keystore you specified on the -sslks parameter, if applicable.	

Table 13. Parameters for createDM command (continued)

Parameter	Description	Value
-wasPorts	<p>The first port in a range of ports that the WebSphere Application Server instance uses. The port you specify should be the first of a contiguous range of 15 open ports. The output of the createDM command displays which ports are assigned to selected WebSphere Application Server functions, such as the EWLM Control Center, for example. You need to remember or record the HTTP ports assigned for the EWLM Control Center and the WebSphere Application Server administration console so that you can connect your Web browser to these applications. You can also use the displayCC command to view the ports currently used by the EWLM Control Center.</p> <p>Note: WebSphere Application Server randomly assigns the port numbers that it requires and does not identify ports that are already in use. You should make certain that all of the ports in the range are not in use or createDM will fail. If a port conflict occurs after you have created the domain manager, the WebSphere Application Server instance and EWLM Control Center will fail to start.</p>	

Related concepts

“Planning for EWLM” on page 79

Use these planning worksheets before you install and configure EWLM.

“Command usage considerations” on page 91

View this topic to learn about the syntax required when issuing an EWLM command. As you complete the EWLM installation and configuration tasks, you may need to run an EWLM command. When you specify an EWLM command, the location from which you run the command and the syntax used to format the command varies by operating system. In addition, the commands that are supported by each operating system vary.

“Configuration wizard usage considerations” on page 106

View this topic to learn about the operating systems that support the EWLM configuration wizards. In addition, learn about the syntax required when you run an EWLM command that launches an EWLM configuration wizard. As you complete the EWLM installation and configuration tasks, you may choose to use the configuration wizards.

“EWLM domain manager” on page 12

The *domain manager* is the central point of control for an EWLM management domain.

Related tasks

“Configuring a domain manager on z/OS” on page 168

View this topic to learn how to configure a domain manager on z/OS.

Configuring a managed server

View this topic to learn how to configure a managed server.

Ensure that you have completed the planning worksheet for the managed server. Use the values from the planning worksheet when you configure the managed server. In addition, ensure that the user responsible for configuring the managed server has the proper user authorities.

After you install the managed servers, you must either run the **EWLM managed server configuration wizard** or the createMS configuration command to create and configure each managed server in the management domain.

You can choose to configure the managed server as part of the installation or choose to configure the managed servers after the installation. If you need to configure the managed server outside of the installation process, use one of the following methods.

- | • “Run the EWLM managed server configuration wizard”
- | • “Run createMS”

| **Run the EWLM managed server configuration wizard:**

- | 1. Go to the `$D(install)/IBM/VE2/EWLMMS/bin` directory where `$D(install)` maps to the platform-specific default directory in which products are typically installed. If you installed the managed server to a different directory, then run the command from the `EWLMMS/bin` subdirectory of that directory.
- | 2. Run the following command:
| `configWizardMS`
| For more information about starting the wizard on your operating system, if supported, see, “Configuration wizard usage considerations” on page 106.
- | 3. Complete the wizard to configure the managed server. Click **Help** if you need assistance.

| **Run createMS:**

| Run the following command, replacing the example values shown in italics with your own.

| `createMS configID -ma address -mp port -auth [None | ServerSSL | ClientServerSSL]`
| `-sslks path -sslpw password`

| For more information about issuing the command on your operating system, if supported, see, “Command usage considerations” on page 91.

| The following table describes each of the parameters and provides a place for you to enter your specific values if you print this page for reference.

Table 14. Parameters for createMS command

Parameter	Description	Value
-auth	<p>The authority level used to secure communications between the domain manager and the managed servers and, if they exist, the firewall brokers. Possible values follow:</p> <p>None Specify this to provide a default level of security between the managed servers and the domain manager. If you specify <i>None</i>, communication between the managed servers and the domain manager is encrypted and no application other than EWLM can communicate with the domain manager. However, it does not prevent a managed server from connecting to a domain manager to which it should not have access. As a general rule, you should not use this value in a production environment. To provide a greater level of security, you should create and use your own SSL certificates.</p> <p>ServerSSL Specify this if you are using server SSL to secure communications between the managed servers and the domain manager. Server SSL authenticates the domain manager to the managed servers and encrypts the data that flows between the servers.</p> <p>ClientServerSSL Specify this if you are using client/server SSL to secure communications between a managed server and the domain manager. Client/server SSL authenticates the domain manager to the managed server, authenticates the managed server to the domain manager, and encrypts the data that flows between the servers.</p> <p>The -auth value you specify on the createMS command must match the -auth value you specify when you configure the domain manager.</p>	
configID	<p>A unique name for this managed server instance. The configuration ID must be an alphanumeric character string. Valid characters are uppercase letters A-Z, lowercase letters a-z, and numbers 0-9. The length of the ID must not exceed 32 characters.</p>	
-ma	<p>The IP address or host name of the domain manager. If the managed server accesses the domain manager through a SOCKS server, then this value is the IP address or host name of the firewall broker.</p> <p>This value must match the -ma value that you specify when you configure the domain manager or firewall broker, depending on which applies. This allows the managed server to properly identify and communicate with its domain manager either directly or through the firewall broker.</p>	
-mp	<p>The port that the domain manager or firewall broker uses to listen for managed server connection requests. This value must match the -mp value that you specify when you configure the domain manager or firewall broker, depending on which applies.</p>	

Table 14. Parameters for createMS command (continued)

Parameter	Description	Value
-sslks	The path to the certificate keystore that contains the SSL keys that are used for securing the communications between the managed servers and the domain manager or firewall broker, if applicable, when -auth is ServerSSL or ClientServerSSL. Note: This parameter is only required if you specify ServerSSL or ClientServerSSL on the -auth parameter.	
-sslpw	The password used to access the keystore that you specified on the -sslks parameter.	

Related concepts

“Planning for EWLM” on page 79

Use these planning worksheets before you install and configure EWLM.

“Command usage considerations” on page 91

View this topic to learn about the syntax required when issuing an EWLM command. As you complete the EWLM installation and configuration tasks, you may need to run an EWLM command. When you specify an EWLM command, the location from which you run the command and the syntax used to format the command varies by operating system. In addition, the commands that are supported by each operating system vary.

“Configuration wizard usage considerations” on page 106

View this topic to learn about the operating systems that support the EWLM configuration wizards. In addition, learn about the syntax required when you run an EWLM command that launches an EWLM configuration wizard. As you complete the EWLM installation and configuration tasks, you may choose to use the configuration wizards.

“EWLM managed server” on page 12

A *managed server* is a server or server instance that EWLM monitors.

Related tasks

“Configuring a managed server on z/OS” on page 179

Configuring an EWLM managed server on z/OS is different from configuring a managed server on other platforms. View this topic to learn how to configure the EWLM managed server on z/OS.

Creating a firewall broker

View this topic to learn how to create a firewall broker. You need to create a firewall broker to allow EWLM communication between the domain manager and managed server, if the managed server accesses the domain manager by using a SOCKS server.

Ensure that you have completed the firewall broker planning worksheet. Use the values from the worksheet when you create the firewall broker. In addition, ensure the user responsible for configuring the firewall broker has the proper user authorities.

If a managed server must access its domain manager through a SOCKS server, you must either run the **EWLM managed server configuration wizard**, which provides a path for firewall broker setup, or the createFB configuration command to create and configure a firewall broker. You can create the firewall broker on one of the managed servers or some other server in the same trusted zone as the managed servers.

Use one of the following methods to create a firewall broker:

- Run the EWLM managed server configuration wizard
- Run createFB

Run the EWLM managed server configuration wizard:

1. Go to the `$(install)/IBM/VE2/EWLMMS/bin` directory where `$(install)` maps to the platform-specific default directory in which products are typically installed. If you installed the firewall broker to a different directory, then run the command from the `EWLMMS/bin` subdirectory of that directory.
2. Run the following command:

```
configWizardMS
```

For more information about starting the wizard on your operating system, if supported, see, "Configuration wizard usage considerations" on page 106.
3. Complete the wizard to configure the firewall broker. Click **Help** if you need assistance.

Run createFB:

Run this command, replacing the example parameters shown in italics with your own.

```
createFB configID -ma address -mp port
          -da address -dp port -fp port
          -auth [None | ServerSSL | ClientServerSSL]
          -sslks path -sslpw password
```

For more information about issuing the command on your operating system, if supported, see, "Command usage considerations" on page 91.

The following table describes each of the parameters and provides a place for you to enter your specific values when you print this page for reference.

Table 15. Parameters for createFB command

Parameter	Description	Value
-auth	<p>The authority level used to secure communications between the firewall broker and the managed servers as well as between the firewall broker and the domain manager. Possible values follow:</p> <p>None Specify this to provide a default level of security between the firewall broker and the managed servers. If you specify <i>None</i>, communication between the firewall brokers and the managed servers is encrypted and no application other than EWLM can communicate with the firewall broker. However, it does not prevent a managed server from connecting to a firewall broker to which it should not have access. As a general rule, you should not use this value in a production environment. To provide a greater level of security, you should create and use your own SSL certificates.</p> <p>ServerSSL Specify this if you are using server SSL to secure communications between the managed servers and the firewall broker. Server SSL authenticates the firewall broker to the managed servers and encrypts the data that flows between the servers.</p> <p>ClientServerSSL Specify this if you are using client/server SSL to secure communications between a managed server and the firewall broker. Client/server SSL authenticates the firewall broker to the managed server, authenticates the managed server to the firewall broker, and encrypts the data that flows between the servers.</p> <p>This value must match the -auth value you specify when you configure each managed server that will use this firewall broker.</p>	

Table 15. Parameters for createFB command (continued)

Parameter	Description	Value
configID	A unique name for the firewall broker. The configuration ID must be an alphanumeric character string. Valid characters are uppercase letters A-Z, lowercase letters a-z, and numbers 0-9. The length of the ID must not exceed 32 characters.	
-da	The IP address or host name of the domain manager. This value must match the -ma parameter you specify on the createDM command.	
-dp	The port that the domain manager uses to listen for connection requests from the firewall broker. This value must match the -fp parameter you specify on the changeDM command.	
-fp	The port that the firewall broker uses to listen for connection requests from the domain manager. This value must match the port you specify when you list this firewall broker on the -fb parameter of the changeDM command.	
-ma	The IP address or the host name that the firewall broker uses for connections requests from managed servers in the same trusted zone. The -ma value that you specify here must match the -ma value that you will specify when you configure each managed server that will use this firewall broker to access the domain manager through a SOCKS server.	
-mp	The port that the firewall broker uses to listen for connection requests from the managed servers in the same trusted zone. The -mp value that you specify here must match the -mp value that you will specify when you configure each managed server that will use this firewall broker to access the domain manager through a SOCKS server.	
-sslks	The path to the certificate keystore that contains the SSL keys that are used for securing the following communications: <ul style="list-style-type: none"> Between the firewall brokers and the managed server when -auth is ServerSSL or ClientServerSSL. Between the firewall brokers and domain manager when -auth is ServerSSL or ClientServerSSL. <p>Note: This parameter is only required if you specify ServerSSL or ClientServerSSL on the -auth parameter.</p>	
-sslpw	The password used to access the keystore that you specified on the -sslks parameter.	

Related concepts

“Planning for EWLM” on page 79

Use these planning worksheets before you install and configure EWLM.

“Command usage considerations” on page 91

View this topic to learn about the syntax required when issuing an EWLM command. As you complete the EWLM installation and configuration tasks, you may need to run an EWLM command. When you specify an EWLM command, the location from which you run the command and the syntax used to format the command varies by operating system. In addition, the commands that are supported by each operating system vary.

“Configuration wizard usage considerations” on page 106

View this topic to learn about the operating systems that support the EWLM configuration wizards. In addition, learn about the syntax required when you run an EWLM command that launches an EWLM configuration wizard. As you complete the EWLM installation and configuration tasks, you may choose to use the configuration wizards.

“Firewall broker” on page 15

A *firewall broker* provides the ability to allow communication through a firewall.

Assigning users to EWLM Control Center roles

View this topic to learn how to assign user IDs that exist on the domain manager’s local user registry to access the EWLM Control Center.

To administer EWLM with the EWLM Control Center, users must have a valid user ID on the server where the domain manager and EWLM Control Center functions run.

Then, you must assign each user ID to an EWLM role based on the function the owner of the user ID will perform when using the EWLM Control Center. The user ID must be granted access to the Control Center by assigning it to an EWLM role. Otherwise, the user cannot log in to the EWLM Control Center. A user can also belong to a group, which can then, in turn, be assigned an EWLM role. All users in a group are assigned the role of the group. The EWLM roles follow:

Administrator

This role has access to all functions. A user assigned to this role can perform setup, manage, and monitor tasks

Operator

This role provides access to the managing and monitoring tasks listed in the EWLM Control Center. A user assigned to this role can manage service policies, view the status of the managed servers in an EWLM domain, and view reports.

Monitor

This role provides access to the monitoring tasks. A user assigned to this role can view reports that contain performance data for the EWLM domain. This role has access to only the monitoring functions of the EWLM Control Center.

To assign user IDs to EWLM Control Center roles, use one of the following methods:

- Run the EWLM domain manager configuration wizard
- Run `changeCC`

If the WebSphere Application Server instance is active when you run the command, then the EWLM Control Center is restarted so that your changes take effect immediately. If the WebSphere Application Server is stopped when you run the command, then the server is left in the stopped state when the command finishes running.

• Run the EWLM domain manager configuration wizard

1. Go to the `$(install)/IBM/VE2/EWLM/bin` directory where `$(install)` maps to the platform-specific default directory in which products are typically installed. If you installed the domain manager to a different directory, then run the command from the `EWLM/bin` subdirectory of that directory.
2. Run the following command:
`configWizardDM`
For more information about starting the wizard on your operating system, if supported, see, “Configuration wizard usage considerations” on page 106.
3. Select the **Add or remove users to EWLM Control Center roles** task. Complete the wizard as indicated. Click **Help** if you need assistance.

• Run `changeCC`

To assign users to EWLM Control Center roles, run the following command, replacing the example parameters shown in *italics* with your own and adding the appropriate command extension.

```
changeCC -addUser configID -adminUser userid -adminPW password  
-roleUser user -role [Administrator | Monitor | Operator]
```

To assign groups to EWLM Control Center roles, run this command, replacing the example parameters shown in *italics* with your own and adding the appropriate command extension.

```
changeCC -addGroup configID -adminUser userid -adminPW password
        -roleGroup group -role [Administrator | Monitor | Operator]
```

For more information about issuing the command on your operating system, if supported, see, “Command usage considerations” on page 91.

The following table describes each of the parameters and provides a place for you to enter your specific values if you print this page for reference.

Table 16. Parameters for changeCC -addUser and changeCC -addGroup scripts

Parameter	Description	Value
-adminPW	The password associated with the -adminUser parameter.	
-adminUser	The WebSphere Application Server administrator user ID. This must be the same user ID that you specified when you ran the createDM command or EWLM domain manager configuration wizard.	
-role	The role you want to assign to the users or groups you specified in the -roleUser or -roleGroup parameter. Possible values are: Administrator, Operator, or Monitor. These values are case-sensitive.	
-roleGroup	The group or groups that you want to access the EWLM Control Center. The same syntactical rules apply to the -roleGroup parameter as for the -roleUser parameter.	
-roleUser	The user or users that you want to access the EWLM Control Center. Specify multiple user IDs by separating each user ID with a colon (:). For example, user1:user2:user3. When a user ID is used to log into the EWLM Control Center, you must specify the same user ID as provided for this parameter. Note that the EWLM Control Center log in field is case sensitive.	
configID	The configuration identifier that EWLM uses to identify the EWLM domain manager. You defined this value when you create the domain manager. You cannot change this value.	

Related concepts

“EWLM Control Center” on page 13

The EWLM Control Center is a Web-based user interface that allows you to manage and monitor the performance of servers in an EWLM management domain. It is installed with and hosted by the domain manager.

Related tasks

“Starting EWLM on the systems in your domain” on page 189

View this topic to learn how to start EWLM on the components of EWLM that you have installed and configured in your EWLM domain. The components of EWLM that you can install and configure include the domain manager, managed servers, and firewall brokers.

“Starting the EWLM Control Center” on page 192

View this topic to learn how to start the EWLM Control Center.

“Removing EWLM Control Center users or groups” on page 281

Use this topic to learn how to remove EWLM Control Center access for a user ID.

Authorizing AIX non-root users to start EWLM on an AIX managed server

View this topic to learn how to authorize non-root users to start EWLM on an AIX managed server. Root users, by default, have authority to start EWLM on an AIX managed server.

Before you begin, you must know the username of those users for whom you are granting access and be logged into the system as the root user.

| If users are not granted the correct authorizations, they will not be able to run managed server processes.
| The root user is the only user who can grant this authorization to additional users.

| Non-root users must be granted the following permissions and abilities:

- | • CAP_EWLM_AGENT and CAP_PROPAGATE ability.
- | • Write access to the /ewlmData/ewlmms/servers directory. This is the default managed server information directory. If you specified a different directory while you were installing the managed server, then you must grant the non-root user access to the servers subdirectory of the directory you specified.
- | • Read, write and execute access to the /ewlmData/ewlmms/servers/configID directory, where *configID* is the configuration ID specified when a managed server is created.
- | • Write access to the /var/adm/ewlm directory.
- | • Write access to the /var/adm/ewlm/ewlmms directory, including its subdirectories and files.
- | • Execute access to the /opt/IBM/VE2/EWLMMS/_jvm directory, including its subdirectories and files. This is the default installation directory. If you installed the managed server to a different directory, then you must grant the non-root user access to the _jvm subdirectory of the directory to which you installed the managed server.

| **Note:** The default permissions for the /opt/IBM/VE2/EWLMMS directory and its other subdirectories and files give non-root users the access needed for running the managed server process.

| To create and use a group to grant multiple non-root users the ability to start a managed server, complete the following:

- | 1. Log on to the managed server as a root user.
- | 2. Create a group. In these instructions, the group name is *ewlmms*, but you can use any name. Use the user and group management tool or commands on your system in compliance with your system security policy. For example, use the SMIT or SMITTY tool, or use the `mkgroup -A ewlmms` command.
- | 3. Run the following commands to give the group you created in Step 2 access to the necessary directory and files:

```
| mkdir /ewlmData/ewlmms/servers  
| mkdir /var/adm/ewlm  
| mkdir /var/adm/ewlm/ewlmms  
| mkdir /var/adm/ewlm/ewlmms/Diagnostics  
| touch /var/adm/ewlm/ewlmms/config.log
```

| **Note:** These directories might already exist on your system if you used EWLM commands, such as the `createMS.sh` command. If you find that these directories do already exist as you complete these steps, ignore the cannot create error returned by the `mkdir` command.

```
| chgrp -R ewlmms /opt/IBM/VE2/EWLMMS/_jvm  
| chmod -R g+X /opt/IBM/VE2/EWLMMS/_jvm
```

```
| chgrp -R ewlmms /ewlmData  
| chmod g+ws,/ewlmData/ewlmms/servers
```

```
| chgrp -R ewlmms /var/adm/ewlm  
| chmod g+w,/var/adm/ewlm  
| chmod -R g+ws,/var/adm/ewlm/ewlmms
```

- | 4. Add the user ID you will use to create and start managed servers to the group created in Step 2, and to the existing groups with access to the /var and /var/adm directories on your system. Use the user and group management tool or commands on your system to do this in compliance with your system security policy. For example, use the SMIT or SMITTY tool, or the following commands:

```
| chgrpmem -m + userID bin  
| chgrpmem -m + userID adm  
| chgrpmem -m + userID ewlmms
```

- | **Note:** In this example, members of the bin group have execute access to the /var directory, and member of the adm group have execute access to the /var/adm directory. Access to these directories might be different on your system. Adjust this step as necessary for your system.
- | 5. Grant the user ID access to the EWLM agent interfaces and information available on your system by running the following command, where *userID* is the ID of the user that you added to the group:
| `chuser capabilities=CAP_EWLM_AGENT,CAP_PROPAGATE userID`
 - | 6. Log on to the managed server system as the user ID you created to create and start a managed server.

| You can add as many user IDs enabled to create and run managed servers as you like. Each user ID will be able to start and run the managed servers it creates. A system can have only one managed server running at a time.

| You can also enable users to run managed servers created by other users. Set group permissions on the managed server data whenever the server stops if you want another user to be able to start it later. To set these permissions, complete the following after the server stops:

- | 1. Log on to the managed server system as a root user.
- | 2. Run the following command, where *configID* is the configuration ID specified when the managed server was created:
| `chmod -R g+rwx /ewlmdata/ewlms/servers/configID`

| **Note:** If a non-root user other than the user who started the managed server attempts to use the stopMS.sh command to stop the server, the command will fail because the non-root user does not have the authority to stop a process started by another non-root user. However, the failed attempt will also make it impossible for any user, even a root user, to use the stopMS.sh command to stop the server. If this occurs, the user who started the managed server or a root user can use the `ps -ef | grep configID` command to find the server's process ID, then use the kill command to kill the process.

| **Authorizing AIX users to ARM services**

| View this topic to learn how to authorize non-root users to ARM services. ARM services must be enabled for the EWLM managed server to function properly. The user ID under which ARM services run must have access to ARM services.

| The root user is automatically authorized to use ARM services. However, non-root users must be granted authorization to use these ARM services. If a non-authorized user attempts to access ARM services, a not authorized code is returned. The root user is the only user who can grant this authorization with the chuser command to additional users.

| To grant authorizations to non-root users to the ARM services, run the following command:

| `chuser capabilities=CAP_ARM_APPLICATION,CAP_PROPAGATE username`

| where *username* is the name of the user to whom you are granting access.

| **Enabling ARM services on AIX**

| View this topic to learn how to enable and disable EWLM on AIX. You must enable ARM services on systems that are EWLM managed servers for the EWLM managed server to function properly. After you install the managed server, you must enable ARM services.

| To enable EWLM on AIX, log in as the root user, or as a user with root authority, and use the series of SMIT panels available by typing `smitty ewlm` at the command line. The SMIT panels will lead you through the following process to enable EWLM on AIX:

- | 1. Select **Change/Show Status of the EWLM services** and press Enter.

2. Select **Enable the EWLM services** and press Enter. You are prompted to confirm your request to enable EWLM. Select Enter to continue or F3 to cancel. You can elect to enable EWLM services immediately, at the next restart, or both.

Use the following procedure to disable EWLM on AIX:

1. Select **Change/Show Status of the EWLM services** and press Enter.
2. Select **Disable the EWLM services** and press Enter. You are prompted to confirm your request to disable EWLM. Select Enter to continue or F3 to cancel.

Configuring EWLM on HP-UX

View this topic to learn how to configure the components of EWLM that you installed. In addition, learn how to enable functions of EWLM that are optional.

The following topics provide step-by-step instructions on how to configure and enable various functions of EWLM. Note that not all of these tasks are required. You need to complete only the configuration tasks that are applicable to your environment.

Configuring a managed server

View this topic to learn how to configure a managed server.

Ensure that you have completed the planning worksheet for the managed server. Use the values from the planning worksheet when you configure the managed server. In addition, ensure that the user responsible for configuring the managed server has the proper user authorities.

After you install the managed servers, you must either run the **EWLM managed server configuration wizard** or the `createMS` configuration command to create and configure each managed server in the management domain.

You can choose to configure the managed server as part of the installation or choose to configure the managed servers after the installation. If you need to configure the managed server outside of the installation process, use one of the following methods.

- “Run the EWLM managed server configuration wizard” on page 112
- “Run `createMS`” on page 112

Run the EWLM managed server configuration wizard:

1. Go to the `$(install)/IBM/VE2/EWLMMS/bin` directory where `$(install)` maps to the platform-specific default directory in which products are typically installed. If you installed the managed server to a different directory, then run the command from the `EWLMMS/bin` subdirectory of that directory.
2. Run the following command:
`configWizardMS`
For more information about starting the wizard on your operating system, if supported, see, “Configuration wizard usage considerations” on page 106.
3. Complete the wizard to configure the managed server. Click **Help** if you need assistance.

Run `createMS`:

Run the following command, replacing the example values shown in italics with your own.

```
createMS configID -ma address -mp port -auth [None | ServerSSL | ClientServerSSL]  
-sslks path -sslpw password
```

For more information about issuing the command on your operating system, if supported, see, “Command usage considerations” on page 91.

The following table describes each of the parameters and provides a place for you to enter your specific values if you print this page for reference.

Table 17. Parameters for createMS command

Parameter	Description	Value
-auth	<p>The authority level used to secure communications between the domain manager and the managed servers and, if they exist, the firewall brokers. Possible values follow:</p> <p>None Specify this to provide a default level of security between the managed servers and the domain manager. If you specify <i>None</i>, communication between the managed servers and the domain manager is encrypted and no application other than EWLM can communicate with the domain manager. However, it does not prevent a managed server from connecting to a domain manager to which it should not have access. As a general rule, you should not use this value in a production environment. To provide a greater level of security, you should create and use your own SSL certificates.</p> <p>ServerSSL Specify this if you are using server SSL to secure communications between the managed servers and the domain manager. Server SSL authenticates the domain manager to the managed servers and encrypts the data that flows between the servers.</p> <p>ClientServerSSL Specify this if you are using client/server SSL to secure communications between a managed server and the domain manager. Client/server SSL authenticates the domain manager to the managed server, authenticates the managed server to the domain manager, and encrypts the data that flows between the servers.</p> <p>The -auth value you specify on the createMS command must match the -auth value you specify when you configure the domain manager.</p>	
configID	<p>A unique name for this managed server instance. The configuration ID must be an alphanumeric character string. Valid characters are uppercase letters A-Z, lowercase letters a-z, and numbers 0-9. The length of the ID must not exceed 32 characters.</p>	
-ma	<p>The IP address or host name of the domain manager. If the managed server accesses the domain manager through a SOCKS server, then this value is the IP address or host name of the firewall broker.</p> <p>This value must match the -ma value that you specify when you configure the domain manager or firewall broker, depending on which applies. This allows the managed server to properly identify and communicate with its domain manager either directly or through the firewall broker.</p>	
-mp	<p>The port that the domain manager or firewall broker uses to listen for managed server connection requests. This value must match the -mp value that you specify when you configure the domain manager or firewall broker, depending on which applies.</p>	

Table 17. Parameters for createMS command (continued)

Parameter	Description	Value
-sslks	The path to the certificate keystore that contains the SSL keys that are used for securing the communications between the managed servers and the domain manager or firewall broker, if applicable, when -auth is ServerSSL or ClientServerSSL. Note: This parameter is only required if you specify ServerSSL or ClientServerSSL on the -auth parameter.	
-sslpw	The password used to access the keystore that you specified on the -sslks parameter.	

Related concepts

“Planning for EWLM” on page 79

Use these planning worksheets before you install and configure EWLM.

“Command usage considerations” on page 91

View this topic to learn about the syntax required when issuing an EWLM command. As you complete the EWLM installation and configuration tasks, you may need to run an EWLM command. When you specify an EWLM command, the location from which you run the command and the syntax used to format the command varies by operating system. In addition, the commands that are supported by each operating system vary.

“Configuration wizard usage considerations” on page 106

View this topic to learn about the operating systems that support the EWLM configuration wizards. In addition, learn about the syntax required when you run an EWLM command that launches an EWLM configuration wizard. As you complete the EWLM installation and configuration tasks, you may choose to use the configuration wizards.

“EWLM managed server” on page 12

A *managed server* is a server or server instance that EWLM monitors.

Related tasks

“Configuring a managed server on z/OS” on page 179

Configuring an EWLM managed server on z/OS is different from configuring a managed server on other platforms. View this topic to learn how to configure the EWLM managed server on z/OS.

Creating a firewall broker

View this topic to learn how to create a firewall broker. You need to create a firewall broker to allow EWLM communication between the domain manager and managed server, if the managed server accesses the domain manager by using a SOCKS server.

Ensure that you have completed the firewall broker planning worksheet. Use the values from the worksheet when you create the firewall broker. In addition, ensure the user responsible for configuring the firewall broker has the proper user authorities.

If a managed server must access its domain manager through a SOCKS server, you must either run the **EWLM managed server configuration wizard**, which provides a path for firewall broker setup, or the createFB configuration command to create and configure a firewall broker. You can create the firewall broker on one of the managed servers or some other server in the same trusted zone as the managed servers.

Use one of the following methods to create a firewall broker:

- Run the EWLM managed server configuration wizard
- Run createFB

Run the EWLM managed server configuration wizard:

1. Go to the `$(install)/IBM/VE2/EWLMS/bin` directory where `$(install)` maps to the platform-specific default directory in which products are typically installed. If you installed the firewall broker to a different directory, then run the command from the `EWLMS/bin` subdirectory of that directory.
2. Run the following command:

```
configWizardMS
```

For more information about starting the wizard on your operating system, if supported, see, "Configuration wizard usage considerations" on page 106.
3. Complete the wizard to configure the firewall broker. Click **Help** if you need assistance.

Run createFB:

Run this command, replacing the example parameters shown in italics with your own.

```
createFB configID -ma address -mp port
          -da address -dp port -fp port
          -auth [None | ServerSSL | ClientServerSSL]
          -sslks path -sslpw password
```

For more information about issuing the command on your operating system, if supported, see, "Command usage considerations" on page 91.

The following table describes each of the parameters and provides a place for you to enter your specific values when you print this page for reference.

Table 18. Parameters for createFB command

Parameter	Description	Value
-auth	<p>The authority level used to secure communications between the firewall broker and the managed servers as well as between the firewall broker and the domain manager. Possible values follow:</p> <p>None Specify this to provide a default level of security between the firewall broker and the managed servers. If you specify <i>None</i>, communication between the firewall brokers and the managed servers is encrypted and no application other than EWLM can communicate with the firewall broker. However, it does not prevent a managed server from connecting to a firewall broker to which it should not have access. As a general rule, you should not use this value in a production environment. To provide a greater level of security, you should create and use your own SSL certificates.</p> <p>ServerSSL Specify this if you are using server SSL to secure communications between the managed servers and the firewall broker. Server SSL authenticates the firewall broker to the managed servers and encrypts the data that flows between the servers.</p> <p>ClientServerSSL Specify this if you are using client/server SSL to secure communications between a managed server and the firewall broker. Client/server SSL authenticates the firewall broker to the managed server, authenticates the managed server to the firewall broker, and encrypts the data that flows between the servers.</p> <p>This value must match the -auth value you specify when you configure each managed server that will use this firewall broker.</p>	

Table 18. Parameters for createFB command (continued)

Parameter	Description	Value
configID	A unique name for the firewall broker. The configuration ID must be an alphanumeric character string. Valid characters are uppercase letters A-Z, lowercase letters a-z, and numbers 0-9. The length of the ID must not exceed 32 characters.	
-da	The IP address or host name of the domain manager. This value must match the -ma parameter you specify on the createDM command.	
-dp	The port that the domain manager uses to listen for connection requests from the firewall broker. This value must match the -fp parameter you specify on the changeDM command.	
-fp	The port that the firewall broker uses to listen for connection requests from the domain manager. This value must match the port you specify when you list this firewall broker on the -fb parameter of the changeDM command.	
-ma	The IP address or the host name that the firewall broker uses for connections requests from managed servers in the same trusted zone. The -ma value that you specify here must match the -ma value that you will specify when you configure each managed server that will use this firewall broker to access the domain manager through a SOCKS server.	
-mp	The port that the firewall broker uses to listen for connection requests from the managed servers in the same trusted zone. The -mp value that you specify here must match the -mp value that you will specify when you configure each managed server that will use this firewall broker to access the domain manager through a SOCKS server.	
-sslks	The path to the certificate keystore that contains the SSL keys that are used for securing the following communications: <ul style="list-style-type: none"> Between the firewall brokers and the managed server when -auth is ServerSSL or ClientServerSSL. Between the firewall brokers and domain manager when -auth is ServerSSL or ClientServerSSL. <p>Note: This parameter is only required if you specify ServerSSL or ClientServerSSL on the -auth parameter.</p>	
-sslpw	The password used to access the keystore that you specified on the -sslks parameter.	

Related concepts

“Planning for EWLM” on page 79

Use these planning worksheets before you install and configure EWLM.

“Command usage considerations” on page 91

View this topic to learn about the syntax required when issuing an EWLM command. As you complete the EWLM installation and configuration tasks, you may need to run an EWLM command. When you specify an EWLM command, the location from which you run the command and the syntax used to format the command varies by operating system. In addition, the commands that are supported by each operating system vary.

“Configuration wizard usage considerations” on page 106

View this topic to learn about the operating systems that support the EWLM configuration wizards. In addition, learn about the syntax required when you run an EWLM command that launches an EWLM configuration wizard. As you complete the EWLM installation and configuration tasks, you may choose to use the configuration wizards.

“Firewall broker” on page 15

A *firewall broker* provides the ability to allow communication through a firewall.

Authorizing HP-UX users to ARM services

View this topic to learn how to authorize users to ARM services. ARM services must be enabled for the EWLM managed server to function properly. The user ID under which ARM services run must have access to ARM services.

You must grant authorization to specific users for them to be able to access ARM-application-services. You can authorize as many as 1024 users in the `/etc/ewlm/auth` file. Similarly, the number of ARM-instrumented processes available is limited to 1024. This limit cannot be changed. If you choose to not authorize other users in this file, only the root user is granted access.

The only type of authorization that is granted in the `/etc/ewlm/auth` file is **application**, which allows the user to access ARM-application-instrumentation services.

You must edit the authorization file to grant users the authorization needed to access the ARM services. Edit this file at one of the following times:

- After the EWLM managed server is installed, but before the system is restarted to enable ARM services. If you choose to edit the file before you restart your system, the authorization changes you make are effective as soon as your system restarts.
- While the ARM services are running. If you choose to edit the file while ARM services are running, you must run the following command after you edit the file to make the changes active:

```
/opt/IBM/VE2/EWLM/usr/sbin/ewlm_init -m
```

To add users to the authorization list for either the application or agent authorizations, complete the following steps:

1. Open the `/etc/ewlm/auth` file.
2. Scroll through the file to the line that shows the type of authorization you want to grant, which appears as `application_auth` in the file.
3. Using commas to separate entries, add the user name for each user you want to be authorized to use ARM application services.

Enabling ARM services on HP-UX

View this topic to learn how to enable ARM services on HP-UX. You must enable ARM services on HP-UX for the EWLM managed server to function properly.

After you install the managed server, you must enable ARM services. You can enable ARM services by either of the following methods:

- **Restart the system**

ARM services is automatically enabled when you restart the system.

- **Run a command**

Without having to restart the system, you can manually enable ARM services by issuing the following command:

```
/opt/IBM/VE2/EWLM/usr/sbin/ewlminit -c
```

Disabling ARM services on HP-UX

View this topic to learn how to disable ARM services on HP-UX. If ARM services is disabled, the EWLM managed server function will not work.

To disable ARM services on HP-UX, run the following command:

Note: You cannot disable ARM services on HP-UX if the managed server is active.

| `/opt/IBM/VE2/EWLM/usr/sbin/ewlm_init -u`

| **Verifying the status of ARM services on HP-UX**

| View this topic to learn how to verify the status of ARM services on HP-UX.

| To verify the status of the ARM services on HP-UX, run the following command:

| `/opt/IBM/VE2/EWLM/usr/sbin/ewlm_init -q`

| **Configuring EWLM on i5/OS**

| View this topic to learn how to configure the components of EWLM that you installed. In addition, learn how to enable functions of EWLM that are optional.

| The following topics provide step-by-step instructions on how to configure and enable various functions of EWLM. Note that not all of these tasks are required. You need to complete only the configuration tasks that are applicable to your environment.

| **Add EWLM library to your system library list**

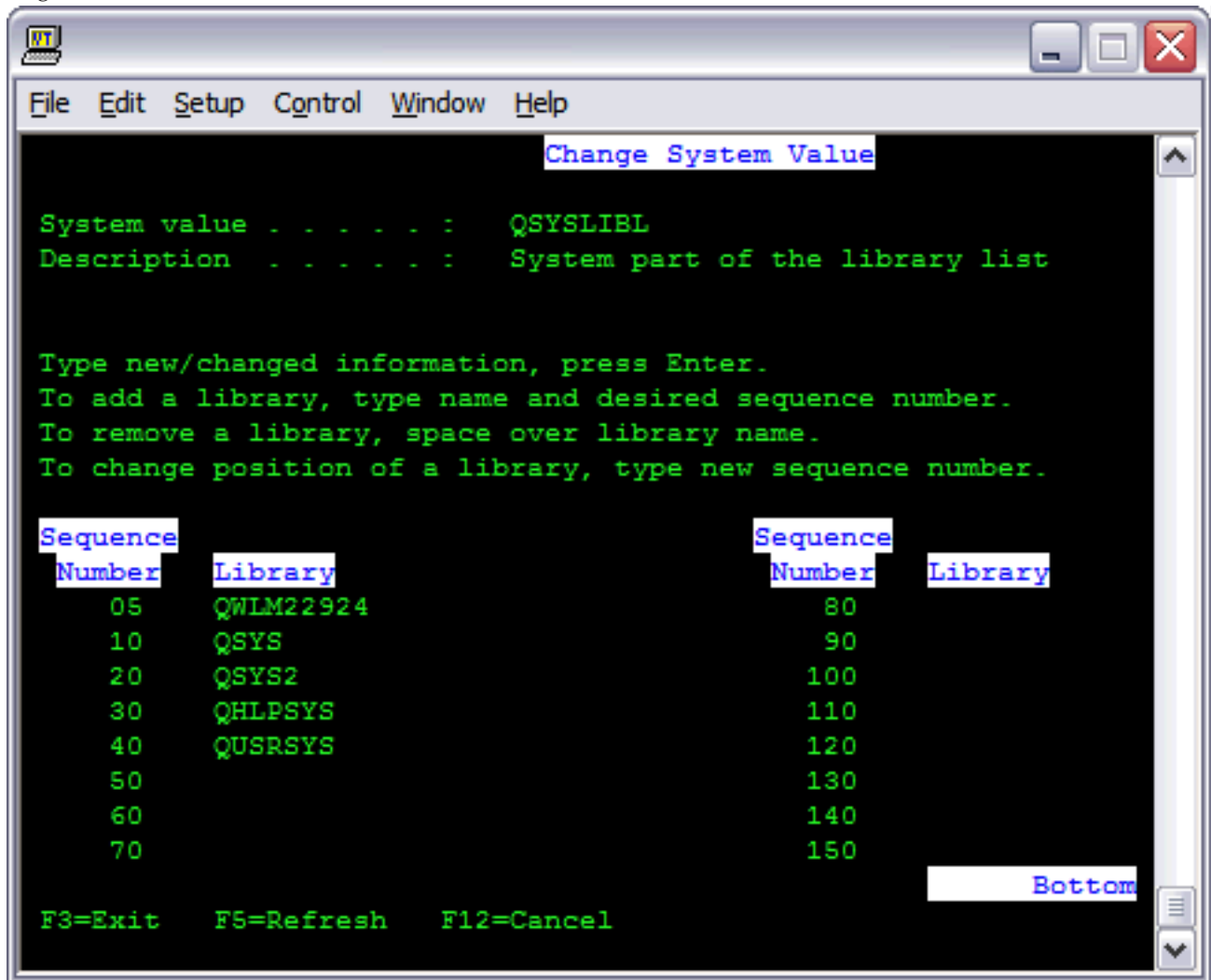
| Learn how to add the EWLM library to your system library list. You must complete this step if English is not the primary language installed on the system.

| If the primary language installed on your i5/OS system is not English, you must add the EWLM library to your system library list. You must perform this step before you configure EWLM on your system. Note that you only need to perform this step once for each i5/OS system on which you install EWLM.

| Complete the following to add the EWLM library to your system library list and ensure that the EWLM library is specified before the QSYS library.

- | 1. Log on to the i5/OS system.
- | 2. Use the WRKSYSVAL QSYSLIBL to change the system library system value.

- Specify library QWLM22924 in a sequence number that precedes the QSYS library. The EWLM library must precede the QSYS library because the QSYS library contains the primary language, which is not English.



Configuring a domain manager

View this topic to learn how to configure a domain manager.

Ensure that you have completed the domain manager worksheets found in the topic “Planning for EWLM” on page 79. Use the values from the planning worksheet when you configure the domain manager. In addition, ensure that the user responsible for configuring the domain manager has the proper user authorities.

After you install the domain manager on your system, you must run either the **EWLM domain manager configuration wizard** or the createDM configuration command to create and configure the domain manager and its associated WebSphere Application Server instance. The WebSphere Application Server instance includes the EWLM Control Center function.

Note: As part of the domain manager configuration, EWLM creates a default target user ID and password that is used to display EWLM in the Virtualization Engine (VE) Console’s Health Center. Therefore, you will notice that you are not prompted for a target user ID and password when you access EWLM in the Health Center. Due to this, ensure that you do not delete or edit this default target user ID and password that EWLM creates as part of the EWLM domain manager configuration.

You can choose to configure the domain manager as part of the installation or choose to configure the domain manager after the installation. If you need to configure the domain manager outside of the installation process, use one of the following methods.

- Run the EWLM domain manager configuration wizard
- Run createDM

Run the EWLM domain manager configuration wizard:

1. Go to the $\$D(install)/IBM/VE2/EWLM/bin$ directory where $\$D(install)$ maps to the platform-specific default directory in which products are typically installed. If you installed the domain manager to a different directory, then run the command from the EWLM/bin subdirectory of that directory.
2. Run the following command:

```
configWizardDM
```

For more information about starting the wizard on your operating system, if supported, see, "Configuration wizard usage considerations" on page 106.
3. Complete the wizard to configure the domain manager. Click **Help** if you need assistance.

Run createDM:

Run the following command, replacing the example values shown in italics with your own.

```
createDM configID -adminUser userid -adminPW password -wasPorts port -jp port
      -ma address -mp port -dn domainName -auth [None | ServerSSL | ClientServerSSL]
      -sslks path -sslpw password
```

For more information about issuing the command on your operating system, if supported, see, "Command usage considerations" on page 91.

The following table describes each of the parameters and provides a place for you to enter your specific values when you print this page for reference.

Table 19. Parameters for createDM command

Parameter	Description	Value
-adminUser	The WebSphere Application Server administrator user ID. This user ID must conform to the WebSphere Application Server requirements for an administrator.	
-adminPW	The password associated with the -adminUser parameter.	

Table 19. Parameters for createDM command (continued)

Parameter	Description	Value
-auth	<p>The authority level used to secure communications between the domain manager and the managed servers and, if they exist, the firewall brokers. Possible values follow:</p> <p>None Specify this to provide a default level of security between the managed servers and the domain manager. If you specify None, communication between the managed servers and the domain manager is encrypted and no application other than EWLM can communicate with the domain manager. However, it does not prevent a managed server from connecting to a domain manager to which it should not have access. As a general rule, you should not use this value in a production environment. To provide a greater level of security, you should create and use your own SSL certificates.</p> <p>ServerSSL Specify this if you are using server SSL to secure communications between the managed servers and the domain manager. Server SSL authenticates the domain manager to the managed servers and encrypts the data that flows between the servers.</p> <p>ClientServerSSL Specify this if you are using client/server SSL to secure communications between a managed server and the domain manager. Client/server SSL authenticates the domain manager to the managed server, authenticates the managed server to the domain manager, and encrypts the data that flows between the servers.</p> <p>The -auth value you specify on the createDM command must match the -auth value that you specify when you configure each managed server and firewall broker, if applicable.</p>	
configID	A unique name for this domain manager instance. The configuration ID must be an alphanumeric character string. Valid characters are uppercase letters A-Z, lowercase letters a-z, and numbers 0-9. The length of the ID must not exceed 32 characters.	
-dn	The name of the EWLM domain. The EWLM domain name must be a character string. All characters are valid except the colon (:), question mark (?) symbol, double quote (") symbol, greater than (>) symbol, less than (<) symbol, or bar () symbol. The length of the ID must not exceed 64 characters.	
-jp	The port that the EWLM Control Center uses to communicate with the domain manager. Ensure that this port is not currently in use and is not the same port that you specify for -mp or one of the 15 contiguous ports that -wasPorts requires.	
-ma	The IP address or host name of the domain manager server. The -ma value you specify on the createDM command must match the -ma value that you specify when you configure each managed server. This allows the managed server to properly identify and communicate with its domain manager.	
-mp	The port that the domain manager uses to listen for managed server connection requests. The -mp value that you specify on the createDM command must match the -mp value you specify when you configure each managed server.	

Table 19. Parameters for createDM command (continued)

Parameter	Description	Value
-sslks	<p>The path of the certificate keystore that contains the SSL keys that are used to secure the following communications:</p> <ul style="list-style-type: none"> Between the domain manager and the managed servers when <code>-auth</code> is <code>ServerSSL</code> or <code>ClientServerSSL</code>. Between the domain manager and firewall brokers, if they exist in the EWLM domain, when <code>-auth</code> is <code>ServerSSL</code> or <code>ClientServerSSL</code>. Between the domain manager and load balancers if you provide a port value for the <code>-lbs</code> parameter. <p>Note: You set the <code>-lbs</code> parameter on the <code>changeDM</code> command.</p> <p>Note: This parameter is only required if you specify <code>ServerSSL</code> or <code>ClientServerSSL</code> on the <code>-auth</code> parameter.</p>	
-sslpw	The password used to access the keystore you specified on the <code>-sslks</code> parameter, if applicable.	
-wasPorts	<p>The first port in a range of ports that the WebSphere Application Server instance uses. The port you specify should be the first of a contiguous range of 15 open ports. The output of the <code>createDM</code> command displays which ports are assigned to selected WebSphere Application Server functions, such as the EWLM Control Center, for example. You need to remember or record the HTTP ports assigned for the EWLM Control Center and the WebSphere Application Server administration console so that you can connect your Web browser to these applications. You can also use the <code>displayCC</code> command to view the ports currently used by the EWLM Control Center.</p> <p>Note: WebSphere Application Server randomly assigns the port numbers that it requires and does not identify ports that are already in use. You should make certain that all of the ports in the range are not in use or <code>createDM</code> will fail. If a port conflict occurs after you have created the domain manager, the WebSphere Application Server instance and EWLM Control Center will fail to start.</p>	

Related concepts

“Planning for EWLM” on page 79

Use these planning worksheets before you install and configure EWLM.

“Command usage considerations” on page 91

View this topic to learn about the syntax required when issuing an EWLM command. As you complete the EWLM installation and configuration tasks, you may need to run an EWLM command. When you specify an EWLM command, the location from which you run the command and the syntax used to format the command varies by operating system. In addition, the commands that are supported by each operating system vary.

“Configuration wizard usage considerations” on page 106

View this topic to learn about the operating systems that support the EWLM configuration wizards. In addition, learn about the syntax required when you run an EWLM command that launches an EWLM configuration wizard. As you complete the EWLM installation and configuration tasks, you may choose to use the configuration wizards.

“EWLM domain manager” on page 12

The *domain manager* is the central point of control for an EWLM management domain.

Related tasks

“Configuring a domain manager on z/OS” on page 168

View this topic to learn how to configure a domain manager on z/OS.

| **Configuring a managed server**

| View this topic to learn how to configure a managed server.

| Ensure that you have completed the planning worksheet for the managed server. Use the values from the planning worksheet when you configure the managed server. In addition, ensure that the user responsible for configuring the managed server has the proper user authorities.

| After you install the managed servers, you must either run the **EWLM managed server configuration wizard** or the `createMS` configuration command to create and configure each managed server in the management domain.

| You can choose to configure the managed server as part of the installation or choose to configure the managed servers after the installation. If you need to configure the managed server outside of the installation process, use one of the following methods.

- | • “Run the EWLM managed server configuration wizard” on page 112
- | • “Run `createMS`” on page 112

| **Run the EWLM managed server configuration wizard:**

| 1. Go to the `$(install)/IBM/VE2/EWLMMS/bin` directory where `$(install)` maps to the platform-specific default directory in which products are typically installed. If you installed the managed server to a different directory, then run the command from the `EWLMMS/bin` subdirectory of that directory.

| 2. Run the following command:

```
| configWizardMS
```

| For more information about starting the wizard on your operating system, if supported, see, “Configuration wizard usage considerations” on page 106.

| 3. Complete the wizard to configure the managed server. Click **Help** if you need assistance.

| **Run `createMS`:**

| Run the following command, replacing the example values shown in italics with your own.

```
| createMS configID -ma address -mp port -auth [None | ServerSSL | ClientServerSSL]  
| -sslks path -sslpw password
```

| For more information about issuing the command on your operating system, if supported, see, “Command usage considerations” on page 91.

| The following table describes each of the parameters and provides a place for you to enter your specific values if you print this page for reference.

Table 20. Parameters for createMS command

Parameter	Description	Value
-auth	<p>The authority level used to secure communications between the domain manager and the managed servers and, if they exist, the firewall brokers. Possible values follow:</p> <p>None Specify this to provide a default level of security between the managed servers and the domain manager. If you specify <i>None</i>, communication between the managed servers and the domain manager is encrypted and no application other than EWLM can communicate with the domain manager. However, it does not prevent a managed server from connecting to a domain manager to which it should not have access. As a general rule, you should not use this value in a production environment. To provide a greater level of security, you should create and use your own SSL certificates.</p> <p>ServerSSL Specify this if you are using server SSL to secure communications between the managed servers and the domain manager. Server SSL authenticates the domain manager to the managed servers and encrypts the data that flows between the servers.</p> <p>ClientServerSSL Specify this if you are using client/server SSL to secure communications between a managed server and the domain manager. Client/server SSL authenticates the domain manager to the managed server, authenticates the managed server to the domain manager, and encrypts the data that flows between the servers.</p> <p>The -auth value you specify on the createMS command must match the -auth value you specify when you configure the domain manager.</p>	
configID	<p>A unique name for this managed server instance. The configuration ID must be an alphanumeric character string. Valid characters are uppercase letters A-Z, lowercase letters a-z, and numbers 0-9. The length of the ID must not exceed 32 characters.</p>	
-ma	<p>The IP address or host name of the domain manager. If the managed server accesses the domain manager through a SOCKS server, then this value is the IP address or host name of the firewall broker.</p> <p>This value must match the -ma value that you specify when you configure the domain manager or firewall broker, depending on which applies. This allows the managed server to properly identify and communicate with its domain manager either directly or through the firewall broker.</p>	
-mp	<p>The port that the domain manager or firewall broker uses to listen for managed server connection requests. This value must match the -mp value that you specify when you configure the domain manager or firewall broker, depending on which applies.</p>	

Table 20. Parameters for createMS command (continued)

Parameter	Description	Value
-sslks	The path to the certificate keystore that contains the SSL keys that are used for securing the communications between the managed servers and the domain manager or firewall broker, if applicable, when -auth is ServerSSL or ClientServerSSL. Note: This parameter is only required if you specify ServerSSL or ClientServerSSL on the -auth parameter.	
-sslpw	The password used to access the keystore that you specified on the -sslks parameter.	

Related concepts

“Planning for EWLM” on page 79

Use these planning worksheets before you install and configure EWLM.

“Command usage considerations” on page 91

View this topic to learn about the syntax required when issuing an EWLM command. As you complete the EWLM installation and configuration tasks, you may need to run an EWLM command.

When you specify an EWLM command, the location from which you run the command and the syntax used to format the command varies by operating system. In addition, the commands that are supported by each operating system vary.

“Configuration wizard usage considerations” on page 106

View this topic to learn about the operating systems that support the EWLM configuration wizards. In addition, learn about the syntax required when you run an EWLM command that launches an EWLM configuration wizard. As you complete the EWLM installation and configuration tasks, you may choose to use the configuration wizards.

“EWLM managed server” on page 12

A *managed server* is a server or server instance that EWLM monitors.

Related tasks

“Configuring a managed server on z/OS” on page 179

Configuring an EWLM managed server on z/OS is different from configuring a managed server on other platforms. View this topic to learn how to configure the EWLM managed server on z/OS.

Creating a firewall broker

View this topic to learn how to create a firewall broker. You need to create a firewall broker to allow EWLM communication between the domain manager and managed server, if the managed server accesses the domain manager by using a SOCKS server.

Ensure that you have completed the firewall broker planning worksheet. Use the values from the worksheet when you create the firewall broker. In addition, ensure the user responsible for configuring the firewall broker has the proper user authorities.

If a managed server must access its domain manager through a SOCKS server, you must either run the **EWLM managed server configuration wizard**, which provides a path for firewall broker setup, or the createFB configuration command to create and configure a firewall broker. You can create the firewall broker on one of the managed servers or some other server in the same trusted zone as the managed servers.

Use one of the following methods to create a firewall broker:

- Run the EWLM managed server configuration wizard
- Run createFB

Run the EWLM managed server configuration wizard:

1. Go to the `$(install)/IBM/VE2/EWLMMS/bin` directory where `$(install)` maps to the platform-specific default directory in which products are typically installed. If you installed the firewall broker to a different directory, then run the command from the `EWLMMS/bin` subdirectory of that directory.
2. Run the following command:

```
configWizardMS
```

For more information about starting the wizard on your operating system, if supported, see, "Configuration wizard usage considerations" on page 106.
3. Complete the wizard to configure the firewall broker. Click **Help** if you need assistance.

Run createFB:

Run this command, replacing the example parameters shown in italics with your own.

```
createFB configID -ma address -mp port
          -da address -dp port -fp port
          -auth [None | ServerSSL | ClientServerSSL]
          -sslks path -sslpw password
```

For more information about issuing the command on your operating system, if supported, see, "Command usage considerations" on page 91.

The following table describes each of the parameters and provides a place for you to enter your specific values when you print this page for reference.

Table 21. Parameters for createFB command

Parameter	Description	Value
-auth	<p>The authority level used to secure communications between the firewall broker and the managed servers as well as between the firewall broker and the domain manager. Possible values follow:</p> <p>None Specify this to provide a default level of security between the firewall broker and the managed servers. If you specify <i>None</i>, communication between the firewall brokers and the managed servers is encrypted and no application other than EWLM can communicate with the firewall broker. However, it does not prevent a managed server from connecting to a firewall broker to which it should not have access. As a general rule, you should not use this value in a production environment. To provide a greater level of security, you should create and use your own SSL certificates.</p> <p>ServerSSL Specify this if you are using server SSL to secure communications between the managed servers and the firewall broker. Server SSL authenticates the firewall broker to the managed servers and encrypts the data that flows between the servers.</p> <p>ClientServerSSL Specify this if you are using client/server SSL to secure communications between a managed server and the firewall broker. Client/server SSL authenticates the firewall broker to the managed server, authenticates the managed server to the firewall broker, and encrypts the data that flows between the servers.</p> <p>This value must match the -auth value you specify when you configure each managed server that will use this firewall broker.</p>	

Table 21. Parameters for createFB command (continued)

Parameter	Description	Value
configID	A unique name for the firewall broker. The configuration ID must be an alphanumeric character string. Valid characters are uppercase letters A-Z, lowercase letters a-z, and numbers 0-9. The length of the ID must not exceed 32 characters.	
-da	The IP address or host name of the domain manager. This value must match the -ma parameter you specify on the createDM command.	
-dp	The port that the domain manager uses to listen for connection requests from the firewall broker. This value must match the -fp parameter you specify on the changeDM command.	
-fp	The port that the firewall broker uses to listen for connection requests from the domain manager. This value must match the port you specify when you list this firewall broker on the -fb parameter of the changeDM command.	
-ma	The IP address or the host name that the firewall broker uses for connections requests from managed servers in the same trusted zone. The -ma value that you specify here must match the -ma value that you will specify when you configure each managed server that will use this firewall broker to access the domain manager through a SOCKS server.	
-mp	The port that the firewall broker uses to listen for connection requests from the managed servers in the same trusted zone. The -mp value that you specify here must match the -mp value that you will specify when you configure each managed server that will use this firewall broker to access the domain manager through a SOCKS server.	
-sslks	The path to the certificate keystore that contains the SSL keys that are used for securing the following communications: <ul style="list-style-type: none"> Between the firewall brokers and the managed server when -auth is ServerSSL or ClientServerSSL. Between the firewall brokers and domain manager when -auth is ServerSSL or ClientServerSSL. <p>Note: This parameter is only required if you specify ServerSSL or ClientServerSSL on the -auth parameter.</p>	
-sslpw	The password used to access the keystore that you specified on the -sslks parameter.	

Related concepts

“Planning for EWLM” on page 79

Use these planning worksheets before you install and configure EWLM.

“Command usage considerations” on page 91

View this topic to learn about the syntax required when issuing an EWLM command. As you complete the EWLM installation and configuration tasks, you may need to run an EWLM command. When you specify an EWLM command, the location from which you run the command and the syntax used to format the command varies by operating system. In addition, the commands that are supported by each operating system vary.

“Configuration wizard usage considerations” on page 106

View this topic to learn about the operating systems that support the EWLM configuration wizards. In addition, learn about the syntax required when you run an EWLM command that launches an EWLM configuration wizard. As you complete the EWLM installation and configuration tasks, you may choose to use the configuration wizards.

“Firewall broker” on page 15

A *firewall broker* provides the ability to allow communication through a firewall.

Assigning users to EWLM Control Center roles

View this topic to learn how to assign user IDs that exist on the domain manager’s local user registry to access the EWLM Control Center.

To administer EWLM with the EWLM Control Center, users must have a valid user ID on the server where the domain manager and EWLM Control Center functions run.

Then, you must assign each user ID to an EWLM role based on the function the owner of the user ID will perform when using the EWLM Control Center. The user ID must be granted access to the Control Center by assigning it to an EWLM role. Otherwise, the user cannot log in to the EWLM Control Center. A user can also belong to a group, which can then, in turn, be assigned an EWLM role. All users in a group are assigned the role of the group. The EWLM roles follow:

Administrator

This role has access to all functions. A user assigned to this role can perform setup, manage, and monitor tasks

Operator

This role provides access to the managing and monitoring tasks listed in the EWLM Control Center. A user assigned to this role can manage service policies, view the status of the managed servers in an EWLM domain, and view reports.

Monitor

This role provides access to the monitoring tasks. A user assigned to this role can view reports that contain performance data for the EWLM domain. This role has access to only the monitoring functions of the EWLM Control Center.

To assign user IDs to EWLM Control Center roles, use one of the following methods:

- Run the EWLM domain manager configuration wizard
- Run `changeCC`

If the WebSphere Application Server instance is active when you run the command, then the EWLM Control Center is restarted so that your changes take effect immediately. If the WebSphere Application Server is stopped when you run the command, then the server is left in the stopped state when the command finishes running.

• Run the EWLM domain manager configuration wizard

1. Go to the `$(install)/IBM/VE2/EWLM/bin` directory where `$(install)` maps to the platform-specific default directory in which products are typically installed. If you installed the domain manager to a different directory, then run the command from the `EWLM/bin` subdirectory of that directory.
2. Run the following command:
`configWizardDM`
For more information about starting the wizard on your operating system, if supported, see, “Configuration wizard usage considerations” on page 106.
3. Select the **Add or remove users to EWLM Control Center roles** task. Complete the wizard as indicated. Click **Help** if you need assistance.

• Run `changeCC`

To assign users to EWLM Control Center roles, run the following command, replacing the example parameters shown in *italics* with your own and adding the appropriate command extension.

```
changeCC -addUser configID -adminUser userid -adminPW password  
-roleUser user -role [Administrator | Monitor | Operator]
```

To assign groups to EWLM Control Center roles, run this command, replacing the example parameters shown in *italics* with your own and adding the appropriate command extension.

```
changeCC -addGroup configID -adminUser userid -adminPW password
        -roleGroup group -role [Administrator | Monitor | Operator]
```

For more information about issuing the command on your operating system, if supported, see, “Command usage considerations” on page 91.

The following table describes each of the parameters and provides a place for you to enter your specific values if you print this page for reference.

Table 22. Parameters for changeCC -addUser and changeCC -addGroup scripts

Parameter	Description	Value
-adminPW	The password associated with the -adminUser parameter.	
-adminUser	The WebSphere Application Server administrator user ID. This must be the same user ID that you specified when you ran the createDM command or EWLM domain manager configuration wizard.	
-role	The role you want to assign to the users or groups you specified in the -roleUser or -roleGroup parameter. Possible values are: Administrator, Operator, or Monitor. These values are case-sensitive.	
-roleGroup	The group or groups that you want to access the EWLM Control Center. The same syntactical rules apply to the -roleGroup parameter as for the -roleUser parameter.	
-roleUser	The user or users that you want to access the EWLM Control Center. Specify multiple user IDs by separating each user ID with a colon (:). For example, user1:user2:user3. When a user ID is used to log into the EWLM Control Center, you must specify the same user ID as provided for this parameter. Note that the EWLM Control Center log in field is case sensitive.	
configID	The configuration identifier that EWLM uses to identify the EWLM domain manager. You defined this value when you create the domain manager. You cannot change this value.	

Related concepts

“EWLM Control Center” on page 13

The EWLM Control Center is a Web-based user interface that allows you to manage and monitor the performance of servers in an EWLM management domain. It is installed with and hosted by the domain manager.

Related tasks

“Starting EWLM on the systems in your domain” on page 189

View this topic to learn how to start EWLM on the components of EWLM that you have installed and configured in your EWLM domain. The components of EWLM that you can install and configure include the domain manager, managed servers, and firewall brokers.

“Starting the EWLM Control Center” on page 192

View this topic to learn how to start the EWLM Control Center.

“Removing EWLM Control Center users or groups” on page 281

Use this topic to learn how to remove EWLM Control Center access for a user ID.

Authorizing i5/OS users to ARM services

View this topic to learn how to authorize users to ARM services. You must complete this task if you want EWLM to monitor an ARM-instrumented application that runs on an i5/OS EWLM managed server.

The user account, under which any ARM-instrumented application will run, must be authorized to the QSYS2/LIBARM4 service program. Alternatively, the application must be owned by a user that has the required authority so that it can adopt the authority of the owner. To grant the proper authority to a user profile, you must use the GRT0BJAUT CL command.

| Run the following command to grant the proper authority to a user profile:

```
| GRTOBJAUT OBJ(QSYS2/LIBARM4) OBJTYPE(*SRVPGM) USER(USERPROFILE) AUT(*USE)
```

| **Enabling ARM services on i5/OS**

| View this topic to learn how to enable ARM services. ARM services must be enabled for the EWLM managed server to function properly.

| EWLM ARM support is included with the i5/OS operating system. When you install the EWLM managed server on i5/OS, ARM services is automatically enabled.

| If for some reason you plan to use ARM on i5/OS without the EWLM managed server function installed, you will need to call the `QpmwChgState()` API to enable it. This API also enables the EWLM APIs.

| **Configuring EWLM on Linux**

| View this topic to learn how to configure the components of EWLM that you installed. In addition, learn how to enable functions of EWLM that are optional.

| The following topics provide step-by-step instructions on how to configure and enable various functions of EWLM. Note that not all of these tasks are required. You need to complete only the configuration tasks that are applicable to your environment.

| **Configuring a domain manager**

| View this topic to learn how to configure a domain manager.

| Ensure that you have completed the domain manager worksheets found in the topic “Planning for EWLM” on page 79. Use the values from the planning worksheet when you configure the domain manager. In addition, ensure that the user responsible for configuring the domain manager has the proper user authorities.

| After you install the domain manager on your system, you must run either the **EWLM domain manager configuration wizard** or the `createDM` configuration command to create and configure the domain manager and its associated WebSphere Application Server instance. The WebSphere Application Server instance includes the EWLM Control Center function.

| **Note:** As part of the domain manager configuration, EWLM creates a default target user ID and password that is used to display EWLM in the Virtualization Engine (VE) Console’s Health Center. Therefore, you will notice that you are not prompted for a target user ID and password when you access EWLM in the Health Center. Due to this, ensure that you do not delete or edit this default target user ID and password that EWLM creates as part of the EWLM domain manager configuration.

| You can choose to configure the domain manager as part of the installation or choose to configure the domain manager after the installation. If you need to configure the domain manager outside of the installation process, use one of the following methods.

- | • Run the EWLM domain manager configuration wizard
- | • Run `createDM`

| **Run the EWLM domain manager configuration wizard:**

| 1. Go to the `$D(install)/IBM/VE2/EWLM/bin` directory where `$D(install)` maps to the platform-specific default directory in which products are typically installed. If you installed the domain manager to a different directory, then run the command from the EWLM/bin subdirectory of that directory.

| 2. Run the following command:

```
| configWizardDM
```

For more information about starting the wizard on your operating system, if supported, see, “Configuration wizard usage considerations” on page 106.

3. Complete the wizard to configure the domain manager. Click **Help** if you need assistance.

Run createDM:

Run the following command, replacing the example values shown in italics with your own.

```
createDM configID -adminUser userid -adminPW password -wasPorts port -jp port
      -ma address -mp port -dn domainName -auth [None | ServerSSL | ClientServerSSL]
      -sslks path -sslpw password
```

For more information about issuing the command on your operating system, if supported, see, “Command usage considerations” on page 91.

The following table describes each of the parameters and provides a place for you to enter your specific values when you print this page for reference.

Table 23. Parameters for createDM command

Parameter	Description	Value
-adminUser	The WebSphere Application Server administrator user ID. This user ID must conform to the WebSphere Application Server requirements for an administrator.	
-adminPW	The password associated with the -adminUser parameter.	
-auth	<p>The authority level used to secure communications between the domain manager and the managed servers and, if they exist, the firewall brokers. Possible values follow:</p> <p>None Specify this to provide a default level of security between the managed servers and the domain manager. If you specify None, communication between the managed servers and the domain manager is encrypted and no application other than EWLM can communicate with the domain manager. However, it does not prevent a managed server from connecting to a domain manager to which it should not have access. As a general rule, you should not use this value in a production environment. To provide a greater level of security, you should create and use your own SSL certificates.</p> <p>ServerSSL Specify this if you are using server SSL to secure communications between the managed servers and the domain manager. Server SSL authenticates the domain manager to the managed servers and encrypts the data that flows between the servers.</p> <p>ClientServerSSL Specify this if you are using client/server SSL to secure communications between a managed server and the domain manager. Client/server SSL authenticates the domain manager to the managed server, authenticates the managed server to the domain manager, and encrypts the data that flows between the servers.</p> <p>The -auth value you specify on the createDM command must match the -auth value that you specify when you configure each managed server and firewall broker, if applicable.</p>	

Table 23. Parameters for createDM command (continued)

Parameter	Description	Value
configID	A unique name for this domain manager instance. The configuration ID must be an alphanumeric character string. Valid characters are uppercase letters A-Z, lowercase letters a-z, and numbers 0-9. The length of the ID must not exceed 32 characters.	
-dn	The name of the EWLM domain. The EWLM domain name must be a character string. All characters are valid except the colon (:), question mark (?) symbol, double quote (") symbol, greater than (>) symbol, less than (<) symbol, or bar () symbol. The length of the ID must not exceed 64 characters.	
-jp	The port that the EWLM Control Center uses to communicate with the domain manager. Ensure that this port is not currently in use and is not the same port that you specify for -mp or one of the 15 contiguous ports that -wasPorts requires.	
-ma	The IP address or host name of the domain manager server. The -ma value you specify on the createDM command must match the -ma value that you specify when you configure each managed server. This allows the managed server to properly identify and communicate with its domain manager.	
-mp	The port that the domain manager uses to listen for managed server connection requests. The -mp value that you specify on the createDM command must match the -mp value you specify when you configure each managed server.	
-sslks	The path of the certificate keystore that contains the SSL keys that are used to secure the following communications: <ul style="list-style-type: none"> Between the domain manager and the managed servers when -auth is ServerSSL or ClientServerSSL. Between the domain manager and firewall brokers, if they exist in the EWLM domain, when -auth is ServerSSL or ClientServerSSL. Between the domain manager and load balancers if you provide a port value for the -lbs parameter. Note: You set the -lbs parameter on the changedM command. Note: This parameter is only required if you specify ServerSSL or ClientServerSSL on the -auth parameter.	
-sslpw	The password used to access the keystore you specified on the -sslks parameter, if applicable.	

Table 23. Parameters for createDM command (continued)

Parameter	Description	Value
-wasPorts	<p>The first port in a range of ports that the WebSphere Application Server instance uses. The port you specify should be the first of a contiguous range of 15 open ports. The output of the createDM command displays which ports are assigned to selected WebSphere Application Server functions, such as the EWLM Control Center, for example. You need to remember or record the HTTP ports assigned for the EWLM Control Center and the WebSphere Application Server administration console so that you can connect your Web browser to these applications. You can also use the displayCC command to view the ports currently used by the EWLM Control Center.</p> <p>Note: WebSphere Application Server randomly assigns the port numbers that it requires and does not identify ports that are already in use. You should make certain that all of the ports in the range are not in use or createDM will fail. If a port conflict occurs after you have created the domain manager, the WebSphere Application Server instance and EWLM Control Center will fail to start.</p>	

Related concepts

“Planning for EWLM” on page 79

Use these planning worksheets before you install and configure EWLM.

“Command usage considerations” on page 91

View this topic to learn about the syntax required when issuing an EWLM command. As you complete the EWLM installation and configuration tasks, you may need to run an EWLM command. When you specify an EWLM command, the location from which you run the command and the syntax used to format the command varies by operating system. In addition, the commands that are supported by each operating system vary.

“Configuration wizard usage considerations” on page 106

View this topic to learn about the operating systems that support the EWLM configuration wizards. In addition, learn about the syntax required when you run an EWLM command that launches an EWLM configuration wizard. As you complete the EWLM installation and configuration tasks, you may choose to use the configuration wizards.

“EWLM domain manager” on page 12

The *domain manager* is the central point of control for an EWLM management domain.

Related tasks

“Configuring a domain manager on z/OS” on page 168

View this topic to learn how to configure a domain manager on z/OS.

Configuring a managed server

View this topic to learn how to configure a managed server.

Ensure that you have completed the planning worksheet for the managed server. Use the values from the planning worksheet when you configure the managed server. In addition, ensure that the user responsible for configuring the managed server has the proper user authorities.

After you install the managed servers, you must either run the **EWLM managed server configuration wizard** or the createMS configuration command to create and configure each managed server in the management domain.

You can choose to configure the managed server as part of the installation or choose to configure the managed servers after the installation. If you need to configure the managed server outside of the installation process, use one of the following methods.

- | • “Run the EWLM managed server configuration wizard” on page 112
- | • “Run createMS” on page 112

| **Run the EWLM managed server configuration wizard:**

- | 1. Go to the `$D(install)/IBM/VE2/EWLMMS/bin` directory where `$D(install)` maps to the platform-specific default directory in which products are typically installed. If you installed the managed server to a different directory, then run the command from the `EWLMMS/bin` subdirectory of that directory.
- | 2. Run the following command:
| `configWizardMS`
| For more information about starting the wizard on your operating system, if supported, see, “Configuration wizard usage considerations” on page 106.
- | 3. Complete the wizard to configure the managed server. Click **Help** if you need assistance.

| **Run createMS:**

| Run the following command, replacing the example values shown in italics with your own.

```
| createMS configID -ma address -mp port -auth [None | ServerSSL | ClientServerSSL]  
| -sslks path -sslpw password
```

| For more information about issuing the command on your operating system, if supported, see, “Command usage considerations” on page 91.

| The following table describes each of the parameters and provides a place for you to enter your specific values if you print this page for reference.

Table 24. Parameters for createMS command

Parameter	Description	Value
-auth	<p>The authority level used to secure communications between the domain manager and the managed servers and, if they exist, the firewall brokers. Possible values follow:</p> <p>None Specify this to provide a default level of security between the managed servers and the domain manager. If you specify <i>None</i>, communication between the managed servers and the domain manager is encrypted and no application other than EWLM can communicate with the domain manager. However, it does not prevent a managed server from connecting to a domain manager to which it should not have access. As a general rule, you should not use this value in a production environment. To provide a greater level of security, you should create and use your own SSL certificates.</p> <p>ServerSSL Specify this if you are using server SSL to secure communications between the managed servers and the domain manager. Server SSL authenticates the domain manager to the managed servers and encrypts the data that flows between the servers.</p> <p>ClientServerSSL Specify this if you are using client/server SSL to secure communications between a managed server and the domain manager. Client/server SSL authenticates the domain manager to the managed server, authenticates the managed server to the domain manager, and encrypts the data that flows between the servers.</p> <p>The -auth value you specify on the createMS command must match the -auth value you specify when you configure the domain manager.</p>	
configID	<p>A unique name for this managed server instance. The configuration ID must be an alphanumeric character string. Valid characters are uppercase letters A-Z, lowercase letters a-z, and numbers 0-9. The length of the ID must not exceed 32 characters.</p>	
-ma	<p>The IP address or host name of the domain manager. If the managed server accesses the domain manager through a SOCKS server, then this value is the IP address or host name of the firewall broker.</p> <p>This value must match the -ma value that you specify when you configure the domain manager or firewall broker, depending on which applies. This allows the managed server to properly identify and communicate with its domain manager either directly or through the firewall broker.</p>	
-mp	<p>The port that the domain manager or firewall broker uses to listen for managed server connection requests. This value must match the -mp value that you specify when you configure the domain manager or firewall broker, depending on which applies.</p>	

Table 24. Parameters for createMS command (continued)

Parameter	Description	Value
-sslks	The path to the certificate keystore that contains the SSL keys that are used for securing the communications between the managed servers and the domain manager or firewall broker, if applicable, when -auth is ServerSSL or ClientServerSSL. Note: This parameter is only required if you specify ServerSSL or ClientServerSSL on the -auth parameter.	
-sslpw	The password used to access the keystore that you specified on the -sslks parameter.	

Related concepts

“Planning for EWLM” on page 79

Use these planning worksheets before you install and configure EWLM.

“Command usage considerations” on page 91

View this topic to learn about the syntax required when issuing an EWLM command. As you complete the EWLM installation and configuration tasks, you may need to run an EWLM command. When you specify an EWLM command, the location from which you run the command and the syntax used to format the command varies by operating system. In addition, the commands that are supported by each operating system vary.

“Configuration wizard usage considerations” on page 106

View this topic to learn about the operating systems that support the EWLM configuration wizards. In addition, learn about the syntax required when you run an EWLM command that launches an EWLM configuration wizard. As you complete the EWLM installation and configuration tasks, you may choose to use the configuration wizards.

“EWLM managed server” on page 12

A *managed server* is a server or server instance that EWLM monitors.

Related tasks

“Configuring a managed server on z/OS” on page 179

Configuring an EWLM managed server on z/OS is different from configuring a managed server on other platforms. View this topic to learn how to configure the EWLM managed server on z/OS.

Creating a firewall broker

View this topic to learn how to create a firewall broker. You need to create a firewall broker to allow EWLM communication between the domain manager and managed server, if the managed server accesses the domain manager by using a SOCKS server.

Ensure that you have completed the firewall broker planning worksheet. Use the values from the worksheet when you create the firewall broker. In addition, ensure the user responsible for configuring the firewall broker has the proper user authorities.

If a managed server must access its domain manager through a SOCKS server, you must either run the **EWLM managed server configuration wizard**, which provides a path for firewall broker setup, or the createFB configuration command to create and configure a firewall broker. You can create the firewall broker on one of the managed servers or some other server in the same trusted zone as the managed servers.

Use one of the following methods to create a firewall broker:

- Run the EWLM managed server configuration wizard
- Run createFB

Run the EWLM managed server configuration wizard:

1. Go to the `$(install)/IBM/VE2/EWLMS/bin` directory where `$(install)` maps to the platform-specific default directory in which products are typically installed. If you installed the firewall broker to a different directory, then run the command from the `EWLMS/bin` subdirectory of that directory.
2. Run the following command:

```
configWizardMS
```

For more information about starting the wizard on your operating system, if supported, see, "Configuration wizard usage considerations" on page 106.
3. Complete the wizard to configure the firewall broker. Click **Help** if you need assistance.

Run createFB:

Run this command, replacing the example parameters shown in italics with your own.

```
createFB configID -ma address -mp port
          -da address -dp port -fp port
          -auth [None | ServerSSL | ClientServerSSL]
          -sslks path -sslpw password
```

For more information about issuing the command on your operating system, if supported, see, "Command usage considerations" on page 91.

The following table describes each of the parameters and provides a place for you to enter your specific values when you print this page for reference.

Table 25. Parameters for createFB command

Parameter	Description	Value
-auth	<p>The authority level used to secure communications between the firewall broker and the managed servers as well as between the firewall broker and the domain manager. Possible values follow:</p> <p>None Specify this to provide a default level of security between the firewall broker and the managed servers. If you specify <i>None</i>, communication between the firewall brokers and the managed servers is encrypted and no application other than EWLM can communicate with the firewall broker. However, it does not prevent a managed server from connecting to a firewall broker to which it should not have access. As a general rule, you should not use this value in a production environment. To provide a greater level of security, you should create and use your own SSL certificates.</p> <p>ServerSSL Specify this if you are using server SSL to secure communications between the managed servers and the firewall broker. Server SSL authenticates the firewall broker to the managed servers and encrypts the data that flows between the servers.</p> <p>ClientServerSSL Specify this if you are using client/server SSL to secure communications between a managed server and the firewall broker. Client/server SSL authenticates the firewall broker to the managed server, authenticates the managed server to the firewall broker, and encrypts the data that flows between the servers.</p> <p>This value must match the -auth value you specify when you configure each managed server that will use this firewall broker.</p>	

Table 25. Parameters for createFB command (continued)

Parameter	Description	Value
configID	A unique name for the firewall broker. The configuration ID must be an alphanumeric character string. Valid characters are uppercase letters A-Z, lowercase letters a-z, and numbers 0-9. The length of the ID must not exceed 32 characters.	
-da	The IP address or host name of the domain manager. This value must match the -ma parameter you specify on the createDM command.	
-dp	The port that the domain manager uses to listen for connection requests from the firewall broker. This value must match the -fp parameter you specify on the changeDM command.	
-fp	The port that the firewall broker uses to listen for connection requests from the domain manager. This value must match the port you specify when you list this firewall broker on the -fb parameter of the changeDM command.	
-ma	The IP address or the host name that the firewall broker uses for connections requests from managed servers in the same trusted zone. The -ma value that you specify here must match the -ma value that you will specify when you configure each managed server that will use this firewall broker to access the domain manager through a SOCKS server.	
-mp	The port that the firewall broker uses to listen for connection requests from the managed servers in the same trusted zone. The -mp value that you specify here must match the -mp value that you will specify when you configure each managed server that will use this firewall broker to access the domain manager through a SOCKS server.	
-sslks	The path to the certificate keystore that contains the SSL keys that are used for securing the following communications: <ul style="list-style-type: none"> • Between the firewall brokers and the managed server when -auth is ServerSSL or ClientServerSSL. • Between the firewall brokers and domain manager when -auth is ServerSSL or ClientServerSSL. <p>Note: This parameter is only required if you specify ServerSSL or ClientServerSSL on the -auth parameter.</p>	
-sslpw	The password used to access the keystore that you specified on the -sslks parameter.	

Related concepts

“Planning for EWLM” on page 79

Use these planning worksheets before you install and configure EWLM.

“Command usage considerations” on page 91

View this topic to learn about the syntax required when issuing an EWLM command. As you complete the EWLM installation and configuration tasks, you may need to run an EWLM command. When you specify an EWLM command, the location from which you run the command and the syntax used to format the command varies by operating system. In addition, the commands that are supported by each operating system vary.

“Configuration wizard usage considerations” on page 106

View this topic to learn about the operating systems that support the EWLM configuration wizards. In addition, learn about the syntax required when you run an EWLM command that launches an EWLM configuration wizard. As you complete the EWLM installation and configuration tasks, you may choose to use the configuration wizards.

| “Firewall broker” on page 15
| A *firewall broker* provides the ability to allow communication through a firewall.

| **Authorizing Linux users to ARM services**

| View this topic to learn how to authorize users to ARM services. If you have an ARM-instrumented application that you want EWLM to monitor, the group ID under which the application runs must be able to access ARM services.

| The group ID under which the application runs can have either root or nonroot authority.

| To allow an application to run under a nonroot group ID and access ARM services, EWLM creates a nonroot group ID during the managed server installation. The group ID is 'ewlm_armusers'. If the group ID already exists, EWLM modifies its properties so that it is a nonroot group ID and has the necessary permission levels set by EWLM. You must run applications under this nonroot group ID if you want the application to access ARM services using a nonroot ID.

| If EWLM fails to create this group ID, all applications that run on the Linux managed server must run with root authority for the application to access ARM services.

| After you make changes to the ewlm_armusers group ID, you must restart ewlmarmd.

| To stop and start ewlmarmd, run the following commands in the order provided:

- | 1. /etc/init.d/ewlm stop
- | 2. /etc/init.d/ewlm start

| To restart ewlmarmd, run the following command:

- | • /etc/init.d/ewlm restart

| **Related concepts**

| “Enabling ARM on IBM-instrumented applications” on page 205
| Use this topic to learn how to enable ARM on supported middleware applications.

| **Enabling ARM services on Linux**

| View this topic to learn how to enable ARM services on Linux. You must enable ARM services on Linux for the EWLM managed server to function properly.

| After you install the managed server, you must enable ARM services. You can enable ARM services by either of the following methods:

- | • **Restart the system**

| ARM services is automatically enabled when you restart the system.

- | • **Run a command** Without having to restart the system, you can manually enable ARM services by issuing the following command:

| /opt/IBM/VE2/EWLMMS/usr/etc/init.d/ewlm start

| **Stopping ARM services on Linux**

| View this topic to learn how to stop ARM services on Linux.

| To stop ARM services, run the following command:


| /opt/IBM/VE2/EWLMMS/usr/etc/init.d/ewlm stop

Installing required RPMs to allow EWLM to manage a Linux partition

View this topic to learn about an additional task that you must complete if you want to enable EWLM partition management. You must install additional functions on your Linux partition, which is also an EWLM managed server, for EWLM partition management to function properly. You must complete this task before starting EWLM on the managed server.

For EWLM to manage the processing units of a Linux partition, you must ensure that the required RPMs are installed on the managed server.

To set up your Linux managed server for partition management, complete the following:

1. Go to Service and Productivity tools  (<http://techsupport.services.ibm.com/server/lopdiags>) and select **Red Hat** or **SUSE Linux**, depending on which Linux version runs on your managed server.
2. Click **HMC-managed servers**
3. Download and install **Platform Enablement Library** onto your system.
4. Download and install **Dynamic Reconfiguration Tools** onto your system.

Related tasks

“Enabling partition management” on page 249

Use this topic to learn how to enable logical partition management in EWLM.

Assigning users to EWLM Control Center roles

View this topic to learn how to assign user IDs that exist on the domain manager’s local user registry to access the EWLM Control Center.

To administer EWLM with the EWLM Control Center, users must have a valid user ID on the server where the domain manager and EWLM Control Center functions run.

Then, you must assign each user ID to an EWLM role based on the function the owner of the user ID will perform when using the EWLM Control Center. The user ID must be granted access to the Control Center by assigning it to an EWLM role. Otherwise, the user cannot log in to the EWLM Control Center. A user can also belong to a group, which can then, in turn, be assigned an EWLM role. All users in a group are assigned the role of the group. The EWLM roles follow:

Administrator

This role has access to all functions. A user assigned to this role can perform setup, manage, and monitor tasks

Operator

This role provides access to the managing and monitoring tasks listed in the EWLM Control Center. A user assigned to this role can manage service policies, view the status of the managed servers in an EWLM domain, and view reports.

Monitor

This role provides access to the monitoring tasks. A user assigned to this role can view reports that contain performance data for the EWLM domain. This role has access to only the monitoring functions of the EWLM Control Center.

To assign user IDs to EWLM Control Center roles, use one of the following methods:

- Run the EWLM domain manager configuration wizard
- Run `changeCC`

If the WebSphere Application Server instance is active when you run the command, then the EWLM Control Center is restarted so that your changes take effect immediately. If the WebSphere Application Server is stopped when you run the command, then the server is left in the stopped state when the command finishes running.

- **Run the EWLM domain manager configuration wizard**

1. Go to the `$(install)/IBM/VE2/EWLM/bin` directory where `$(install)` maps to the platform-specific default directory in which products are typically installed. If you installed the domain manager to a different directory, then run the command from the `EWLM/bin` subdirectory of that directory.
2. Run the following command:

```
configWizardDM
```

For more information about starting the wizard on your operating system, if supported, see, “Configuration wizard usage considerations” on page 106.
3. Select the **Add or remove users to EWLM Control Center roles** task. Complete the wizard as indicated. Click **Help** if you need assistance.

- **Run changeCC**

To assign users to EWLM Control Center roles, run the following command, replacing the example parameters shown in italics with your own and adding the appropriate command extension.

```
changeCC -addUser configID -adminUser userid -adminPW password
          -roleUser user -role [Administrator | Monitor | Operator]
```

To assign groups to EWLM Control Center roles, run this command, replacing the example parameters shown in italics with your own and adding the appropriate command extension.

```
changeCC -addGroup configID -adminUser userid -adminPW password
          -roleGroup group -role [Administrator | Monitor | Operator]
```

For more information about issuing the command on your operating system, if supported, see, “Command usage considerations” on page 91.

The following table describes each of the parameters and provides a place for you to enter your specific values if you print this page for reference.

Table 26. Parameters for `changeCC -addUser` and `changeCC -addGroup` scripts

Parameter	Description	Value
-adminPW	The password associated with the -adminUser parameter.	
-adminUser	The WebSphere Application Server administrator user ID. This must be the same user ID that you specified when you ran the <code>createDM</code> command or EWLM domain manager configuration wizard.	
-role	The role you want to assign to the users or groups you specified in the -roleUser or -roleGroup parameter. Possible values are: Administrator, Operator, or Monitor. These values are case-sensitive.	
-roleGroup	The group or groups that you want to access the EWLM Control Center. The same syntactical rules apply to the -roleGroup parameter as for the -roleUser parameter.	
-roleUser	The user or users that you want to access the EWLM Control Center. Specify multiple user IDs by separating each user ID with a colon (:). For example, <code>user1:user2:user3</code> . When a user ID is used to log into the EWLM Control Center, you must specify the same user ID as provided for this parameter. Note that the EWLM Control Center log in field is case sensitive.	
configID	The configuration identifier that EWLM uses to identify the EWLM domain manager. You defined this value when you create the domain manager. You cannot change this value.	

Related concepts

“EWLM Control Center” on page 13

The EWLM Control Center is a Web-based user interface that allows you to manage and monitor the performance of servers in an EWLM management domain. It is installed with and hosted by the domain manager.

Related tasks

“Starting EWLM on the systems in your domain” on page 189

View this topic to learn how to start EWLM on the components of EWLM that you have installed and configured in your EWLM domain. The components of EWLM that you can install and configure include the domain manager, managed servers, and firewall brokers.

“Starting the EWLM Control Center” on page 192

View this topic to learn how to start the EWLM Control Center.

“Removing EWLM Control Center users or groups” on page 281

Use this topic to learn how to remove EWLM Control Center access for a user ID.

Configuring EWLM on Solaris

View this topic to learn how to configure the components of EWLM that you installed. In addition, learn how to enable functions of EWLM that are optional.

The following topics provide step-by-step instructions on how to configure and enable various functions of EWLM. Note that not all of these tasks are required. You need to complete only the configuration tasks that are applicable to your environment.

Configuring a managed server

View this topic to learn how to configure a managed server.

Ensure that you have completed the planning worksheet for the managed server. Use the values from the planning worksheet when you configure the managed server. In addition, ensure that the user responsible for configuring the managed server has the proper user authorities.

After you install the managed servers, you must either run the **EWLM managed server configuration wizard** or the `createMS` configuration command to create and configure each managed server in the management domain.

You can choose to configure the managed server as part of the installation or choose to configure the managed servers after the installation. If you need to configure the managed server outside of the installation process, use one of the following methods.

- “Run the EWLM managed server configuration wizard” on page 112
- “Run `createMS`” on page 112

Run the EWLM managed server configuration wizard:

1. Go to the `$(install)/IBM/VE2/EWLMMS/bin` directory where `$(install)` maps to the platform-specific default directory in which products are typically installed. If you installed the managed server to a different directory, then run the command from the `EWLMMS/bin` subdirectory of that directory.
2. Run the following command:

```
configWizardMS
```

For more information about starting the wizard on your operating system, if supported, see, “Configuration wizard usage considerations” on page 106.
3. Complete the wizard to configure the managed server. Click **Help** if you need assistance.

Run `createMS`:

Run the following command, replacing the example values shown in *italics* with your own.

```
createMS configID -ma address -mp port -auth [None | ServerSSL | ClientServerSSL]  
-sslks path -sslpw password
```

For more information about issuing the command on your operating system, if supported, see, “Command usage considerations” on page 91.

The following table describes each of the parameters and provides a place for you to enter your specific values if you print this page for reference.

Table 27. Parameters for createMS command

Parameter	Description	Value
-auth	<p>The authority level used to secure communications between the domain manager and the managed servers and, if they exist, the firewall brokers. Possible values follow:</p> <p>None Specify this to provide a default level of security between the managed servers and the domain manager. If you specify <i>None</i>, communication between the managed servers and the domain manager is encrypted and no application other than EWLM can communicate with the domain manager. However, it does not prevent a managed server from connecting to a domain manager to which it should not have access. As a general rule, you should not use this value in a production environment. To provide a greater level of security, you should create and use your own SSL certificates.</p> <p>ServerSSL Specify this if you are using server SSL to secure communications between the managed servers and the domain manager. Server SSL authenticates the domain manager to the managed servers and encrypts the data that flows between the servers.</p> <p>ClientServerSSL Specify this if you are using client/server SSL to secure communications between a managed server and the domain manager. Client/server SSL authenticates the domain manager to the managed server, authenticates the managed server to the domain manager, and encrypts the data that flows between the servers.</p> <p>The -auth value you specify on the createMS command must match the -auth value you specify when you configure the domain manager.</p>	
configID	A unique name for this managed server instance. The configuration ID must be an alphanumeric character string. Valid characters are uppercase letters A-Z, lowercase letters a-z, and numbers 0-9. The length of the ID must not exceed 32 characters.	
-ma	<p>The IP address or host name of the domain manager. If the managed server accesses the domain manager through a SOCKS server, then this value is the IP address or host name of the firewall broker.</p> <p>This value must match the -ma value that you specify when you configure the domain manager or firewall broker, depending on which applies. This allows the managed server to properly identify and communicate with its domain manager either directly or through the firewall broker.</p>	
-mp	The port that the domain manager or firewall broker uses to listen for managed server connection requests. This value must match the -mp value that you specify when you configure the domain manager or firewall broker, depending on which applies.	

Table 27. Parameters for createMS command (continued)

Parameter	Description	Value
-sslks	The path to the certificate keystore that contains the SSL keys that are used for securing the communications between the managed servers and the domain manager or firewall broker, if applicable, when -auth is ServerSSL or ClientServerSSL. Note: This parameter is only required if you specify ServerSSL or ClientServerSSL on the -auth parameter.	
-sslpw	The password used to access the keystore that you specified on the -sslks parameter.	

Related concepts

“Planning for EWLM” on page 79

Use these planning worksheets before you install and configure EWLM.

“Command usage considerations” on page 91

View this topic to learn about the syntax required when issuing an EWLM command. As you complete the EWLM installation and configuration tasks, you may need to run an EWLM command. When you specify an EWLM command, the location from which you run the command and the syntax used to format the command varies by operating system. In addition, the commands that are supported by each operating system vary.

“Configuration wizard usage considerations” on page 106

View this topic to learn about the operating systems that support the EWLM configuration wizards. In addition, learn about the syntax required when you run an EWLM command that launches an EWLM configuration wizard. As you complete the EWLM installation and configuration tasks, you may choose to use the configuration wizards.

“EWLM managed server” on page 12

A *managed server* is a server or server instance that EWLM monitors.

Related tasks

“Configuring a managed server on z/OS” on page 179

Configuring an EWLM managed server on z/OS is different from configuring a managed server on other platforms. View this topic to learn how to configure the EWLM managed server on z/OS.

Creating a firewall broker

View this topic to learn how to create a firewall broker. You need to create a firewall broker to allow EWLM communication between the domain manager and managed server, if the managed server accesses the domain manager by using a SOCKS server.

Ensure that you have completed the firewall broker planning worksheet. Use the values from the worksheet when you create the firewall broker. In addition, ensure the user responsible for configuring the firewall broker has the proper user authorities.

If a managed server must access its domain manager through a SOCKS server, you must either run the **EWLM managed server configuration wizard**, which provides a path for firewall broker setup, or the createFB configuration command to create and configure a firewall broker. You can create the firewall broker on one of the managed servers or some other server in the same trusted zone as the managed servers.

Use one of the following methods to create a firewall broker:

- Run the EWLM managed server configuration wizard
- Run createFB

Run the EWLM managed server configuration wizard:

1. Go to the `$(install)/IBM/VE2/EWLMS/bin` directory where `$(install)` maps to the platform-specific default directory in which products are typically installed. If you installed the firewall broker to a different directory, then run the command from the `EWLMS/bin` subdirectory of that directory.
2. Run the following command:

```
configWizardMS
```

For more information about starting the wizard on your operating system, if supported, see, "Configuration wizard usage considerations" on page 106.
3. Complete the wizard to configure the firewall broker. Click **Help** if you need assistance.

Run createFB:

Run this command, replacing the example parameters shown in italics with your own.

```
createFB configID -ma address -mp port
          -da address -dp port -fp port
          -auth [None | ServerSSL | ClientServerSSL]
          -sslks path -sslpw password
```

For more information about issuing the command on your operating system, if supported, see, "Command usage considerations" on page 91.

The following table describes each of the parameters and provides a place for you to enter your specific values when you print this page for reference.

Table 28. Parameters for createFB command

Parameter	Description	Value
-auth	<p>The authority level used to secure communications between the firewall broker and the managed servers as well as between the firewall broker and the domain manager. Possible values follow:</p> <p>None Specify this to provide a default level of security between the firewall broker and the managed servers. If you specify <i>None</i>, communication between the firewall brokers and the managed servers is encrypted and no application other than EWLM can communicate with the firewall broker. However, it does not prevent a managed server from connecting to a firewall broker to which it should not have access. As a general rule, you should not use this value in a production environment. To provide a greater level of security, you should create and use your own SSL certificates.</p> <p>ServerSSL Specify this if you are using server SSL to secure communications between the managed servers and the firewall broker. Server SSL authenticates the firewall broker to the managed servers and encrypts the data that flows between the servers.</p> <p>ClientServerSSL Specify this if you are using client/server SSL to secure communications between a managed server and the firewall broker. Client/server SSL authenticates the firewall broker to the managed server, authenticates the managed server to the firewall broker, and encrypts the data that flows between the servers.</p> <p>This value must match the -auth value you specify when you configure each managed server that will use this firewall broker.</p>	

Table 28. Parameters for createFB command (continued)

Parameter	Description	Value
configID	A unique name for the firewall broker. The configuration ID must be an alphanumeric character string. Valid characters are uppercase letters A-Z, lowercase letters a-z, and numbers 0-9. The length of the ID must not exceed 32 characters.	
-da	The IP address or host name of the domain manager. This value must match the -ma parameter you specify on the createDM command.	
-dp	The port that the domain manager uses to listen for connection requests from the firewall broker. This value must match the -fp parameter you specify on the changeDM command.	
-fp	The port that the firewall broker uses to listen for connection requests from the domain manager. This value must match the port you specify when you list this firewall broker on the -fb parameter of the changeDM command.	
-ma	The IP address or the host name that the firewall broker uses for connections requests from managed servers in the same trusted zone. The -ma value that you specify here must match the -ma value that you will specify when you configure each managed server that will use this firewall broker to access the domain manager through a SOCKS server.	
-mp	The port that the firewall broker uses to listen for connection requests from the managed servers in the same trusted zone. The -mp value that you specify here must match the -mp value that you will specify when you configure each managed server that will use this firewall broker to access the domain manager through a SOCKS server.	
-sslks	The path to the certificate keystore that contains the SSL keys that are used for securing the following communications: <ul style="list-style-type: none"> Between the firewall brokers and the managed server when -auth is ServerSSL or ClientServerSSL. Between the firewall brokers and domain manager when -auth is ServerSSL or ClientServerSSL. <p>Note: This parameter is only required if you specify ServerSSL or ClientServerSSL on the -auth parameter.</p>	
-sslpw	The password used to access the keystore that you specified on the -sslks parameter.	

Related concepts

“Planning for EWLM” on page 79

Use these planning worksheets before you install and configure EWLM.

“Command usage considerations” on page 91

View this topic to learn about the syntax required when issuing an EWLM command. As you complete the EWLM installation and configuration tasks, you may need to run an EWLM command. When you specify an EWLM command, the location from which you run the command and the syntax used to format the command varies by operating system. In addition, the commands that are supported by each operating system vary.

“Configuration wizard usage considerations” on page 106

View this topic to learn about the operating systems that support the EWLM configuration wizards. In addition, learn about the syntax required when you run an EWLM command that launches an EWLM configuration wizard. As you complete the EWLM installation and configuration tasks, you may choose to use the configuration wizards.

| “Firewall broker” on page 15
| A *firewall broker* provides the ability to allow communication through a firewall.

| **Authorizing Solaris users to ARM services**

| Use this procedure to authorize up to 1024 Solaris users to ARM services by editing the `/etc/ewlm/auth` file.

| You must grant authorization to specific users for them to be able to access ARM application services.
| You can authorize as many as 1024 users in the `/etc/ewlm/auth` file. Similarly, the number of
| ARM-instrumented processes available is limited to 1024. This limit cannot be changed. If you choose to
| not authorize other users in this file, only the root user is granted access.

| The only type of authorization that is granted in the `/etc/ewlm/auth` file is **application**, which allows
| the user to access ARM application instrumentation services.

| You must edit the authorization file to grant users the authorization needed to access the ARM services.
| Edit this file at one of the following times:

- | • After the EWLM managed server is installed, but before the system is restarted to enable ARM
| services. If you choose to edit the file before you restart your system, the authorization changes you
| make are effective as soon as your system restarts.
- | • While the ARM services are running. If you choose to edit the file while ARM services are running,
| you must run the following command after you edit the file to make the changes active:
| `/opt/IBM/EWLM/usr/sbin/ewlm_init -m`

| To add users to the authorization list for either the application or agent authorizations, complete the
| following:

- | 1. Open the `/etc/ewlm/auth` file.
- | 2. Scroll through the file to the line that shows the type of authorization you want to grant, which
| appears as `application_auth` in the file.
- | 3. Using commas to separate entries, add the username for each user you want to be authorized to use
| ARM application services.

| **Enabling ARM services on Solaris**

| View this topic to learn how to enable ARM services on Solaris. You must enable ARM services on
| systems that are EWLM managed servers for the EWLM managed server to function properly.

| After you install the managed server, you must enable ARM services. You can enable ARM services by
| either of the following methods:

- | • **Restart the system**
| ARM services is automatically enabled when you restart the system.
- | • **Run a command**
| Without having to restart the system, you can manually enable ARM services by issuing the following
| command:
| `/opt/IBM/EWLM/usr/sbin/ewlm_init -c`

| **Verifying the status of ARM services on Solaris**

| Use this procedure to verify the status of ARM services.

| To verify the status of the ARM services on Solaris, run the following command:

| `/opt/IBM/EWLM/sbin/ewlm_init -q`

Disabling ARM services on Solaris

Use this procedure to disable ARM services on the managed server only when the managed server is not active.

To disable EWLM on Solaris, run the following command:

Note: You cannot disable EWLM on Solaris if the managed server is active.

```
/opt/IBM/EWLM/sbin/ewlm_init -u
```

Configuring EWLM on Windows

View this topic to learn how to configure the components of EWLM that you installed. In addition, learn how to enable functions of EWLM that are optional.

The following topics provide step-by-step instructions on how to configure and enable various functions of EWLM. Note that not all of these tasks are required. You need to complete only the configuration tasks that are applicable to your environment.

Configuring a domain manager

View this topic to learn how to configure a domain manager.

Ensure that you have completed the domain manager worksheets found in the topic “Planning for EWLM” on page 79. Use the values from the planning worksheet when you configure the domain manager. In addition, ensure that the user responsible for configuring the domain manager has the proper user authorities.

After you install the domain manager on your system, you must run either the **EWLM domain manager configuration wizard** or the createDM configuration command to create and configure the domain manager and its associated WebSphere Application Server instance. The WebSphere Application Server instance includes the EWLM Control Center function.

Note: As part of the domain manager configuration, EWLM creates a default target user ID and password that is used to display EWLM in the Virtualization Engine (VE) Console’s Health Center. Therefore, you will notice that you are not prompted for a target user ID and password when you access EWLM in the Health Center. Due to this, ensure that you do not delete or edit this default target user ID and password that EWLM creates as part of the EWLM domain manager configuration.

You can choose to configure the domain manager as part of the installation or choose to configure the domain manager after the installation. If you need to configure the domain manager outside of the installation process, use one of the following methods.

- Run the EWLM domain manager configuration wizard
- Run createDM

Run the EWLM domain manager configuration wizard:

1. Go to the $\$D(\text{install})/IBM/VE2/EWLM/bin$ directory where $\$D(\text{install})$ maps to the platform-specific default directory in which products are typically installed. If you installed the domain manager to a different directory, then run the command from the EWLM/bin subdirectory of that directory.
2. Run the following command:

```
configWizardDM
```

For more information about starting the wizard on your operating system, if supported, see, “Configuration wizard usage considerations” on page 106.
3. Complete the wizard to configure the domain manager. Click **Help** if you need assistance.

| **Run createDM:**

| Run the following command, replacing the example values shown in italics with your own.

```
| createDM configID -adminUser userid -adminPW password -wasPorts port -jp port
|         -ma address -mp port -dn domainName -auth [None | ServerSSL | ClientServerSSL]
|         -sslks path -sslpw password
```

| For more information about issuing the command on your operating system, if supported, see, “Command usage considerations” on page 91.

| The following table describes each of the parameters and provides a place for you to enter your specific values when you print this page for reference.

| *Table 29. Parameters for createDM command*

Parameter	Description	Value
-adminUser	The WebSphere Application Server administrator user ID. This user ID must conform to the WebSphere Application Server requirements for an administrator.	
-adminPW	The password associated with the -adminUser parameter.	
-auth	<p>The authority level used to secure communications between the domain manager and the managed servers and, if they exist, the firewall brokers. Possible values follow:</p> <p>None Specify this to provide a default level of security between the managed servers and the domain manager. If you specify None, communication between the managed servers and the domain manager is encrypted and no application other than EWLM can communicate with the domain manager. However, it does not prevent a managed server from connecting to a domain manager to which it should not have access. As a general rule, you should not use this value in a production environment. To provide a greater level of security, you should create and use your own SSL certificates.</p> <p>ServerSSL Specify this if you are using server SSL to secure communications between the managed servers and the domain manager. Server SSL authenticates the domain manager to the managed servers and encrypts the data that flows between the servers.</p> <p>ClientServerSSL Specify this if you are using client/server SSL to secure communications between a managed server and the domain manager. Client/server SSL authenticates the domain manager to the managed server, authenticates the managed server to the domain manager, and encrypts the data that flows between the servers.</p> <p>The -auth value you specify on the createDM command must match the -auth value that you specify when you configure each managed server and firewall broker, if applicable.</p>	
configID	A unique name for this domain manager instance. The configuration ID must be an alphanumeric character string. Valid characters are uppercase letters A-Z, lowercase letters a-z, and numbers 0-9. The length of the ID must not exceed 32 characters.	

Table 29. Parameters for createDM command (continued)

Parameter	Description	Value
-dn	The name of the EWLM domain. The EWLM domain name must be a character string. All characters are valid except the colon (:), question mark (?) symbol, double quote (") symbol, greater than (>) symbol, less than (<) symbol, or bar () symbol. The length of the ID must not exceed 64 characters.	
-jp	The port that the EWLM Control Center uses to communicate with the domain manager. Ensure that this port is not currently in use and is not the same port that you specify for -mp or one of the 15 contiguous ports that -wasPorts requires.	
-ma	The IP address or host name of the domain manager server. The -ma value you specify on the createDM command must match the -ma value that you specify when you configure each managed server. This allows the managed server to properly identify and communicate with its domain manager.	
-mp	The port that the domain manager uses to listen for managed server connection requests. The -mp value that you specify on the createDM command must match the -mp value you specify when you configure each managed server.	
-sslks	The path of the certificate keystore that contains the SSL keys that are used to secure the following communications: <ul style="list-style-type: none"> Between the domain manager and the managed servers when -auth is ServerSSL or ClientServerSSL. Between the domain manager and firewall brokers, if they exist in the EWLM domain, when -auth is ServerSSL or ClientServerSSL. Between the domain manager and load balancers if you provide a port value for the -lbs parameter. <p>Note: You set the -lbs parameter on the changedDM command.</p> <p>Note: This parameter is only required if you specify ServerSSL or ClientServerSSL on the -auth parameter.</p>	
-sslpw	The password used to access the keystore you specified on the -sslks parameter, if applicable.	
-wasPorts	The first port in a range of ports that the WebSphere Application Server instance uses. The port you specify should be the first of a contiguous range of 15 open ports. The output of the createDM command displays which ports are assigned to selected WebSphere Application Server functions, such as the EWLM Control Center, for example. You need to remember or record the HTTP ports assigned for the EWLM Control Center and the WebSphere Application Server administration console so that you can connect your Web browser to these applications. You can also use the displayCC command to view the ports currently used by the EWLM Control Center.	
	<p>Note: WebSphere Application Server randomly assigns the port numbers that it requires and does not identify ports that are already in use. You should make certain that all of the ports in the range are not in use or createDM will fail. If a port conflict occurs after you have created the domain manager, the WebSphere Application Server instance and EWLM Control Center will fail to start.</p>	

Related concepts

“Planning for EWLM” on page 79

Use these planning worksheets before you install and configure EWLM.

“Command usage considerations” on page 91

View this topic to learn about the syntax required when issuing an EWLM command. As you complete the EWLM installation and configuration tasks, you may need to run an EWLM command. When you specify an EWLM command, the location from which you run the command and the syntax used to format the command varies by operating system. In addition, the commands that are supported by each operating system vary.

“Configuration wizard usage considerations” on page 106

View this topic to learn about the operating systems that support the EWLM configuration wizards. In addition, learn about the syntax required when you run an EWLM command that launches an EWLM configuration wizard. As you complete the EWLM installation and configuration tasks, you may choose to use the configuration wizards.

“EWLM domain manager” on page 12

The *domain manager* is the central point of control for an EWLM management domain.

Related tasks

“Configuring a domain manager on z/OS” on page 168

View this topic to learn how to configure a domain manager on z/OS.

Configuring a managed server

View this topic to learn how to configure a managed server.

Ensure that you have completed the planning worksheet for the managed server. Use the values from the planning worksheet when you configure the managed server. In addition, ensure that the user responsible for configuring the managed server has the proper user authorities.

After you install the managed servers, you must either run the **EWLM managed server configuration wizard** or the `createMS` configuration command to create and configure each managed server in the management domain.

You can choose to configure the managed server as part of the installation or choose to configure the managed servers after the installation. If you need to configure the managed server outside of the installation process, use one of the following methods.

- “Run the EWLM managed server configuration wizard” on page 112
- “Run `createMS`” on page 112

Run the EWLM managed server configuration wizard:

1. Go to the `$(install)/IBM/VE2/EWLMMS/bin` directory where `$(install)` maps to the platform-specific default directory in which products are typically installed. If you installed the managed server to a different directory, then run the command from the `EWLMMS/bin` subdirectory of that directory.
2. Run the following command:
`configWizardMS`
For more information about starting the wizard on your operating system, if supported, see, “Configuration wizard usage considerations” on page 106.
3. Complete the wizard to configure the managed server. Click **Help** if you need assistance.

Run `createMS`:

Run the following command, replacing the example values shown in *italics* with your own.

```
createMS configID -ma address -mp port -auth [None | ServerSSL | ClientServerSSL]  
-sslks path -sslpw password
```


For more information about issuing the command on your operating system, if supported, see, “Command usage considerations” on page 91.

The following table describes each of the parameters and provides a place for you to enter your specific values if you print this page for reference.

Table 30. Parameters for createMS command

Parameter	Description	Value
-auth	<p>The authority level used to secure communications between the domain manager and the managed servers and, if they exist, the firewall brokers. Possible values follow:</p> <p>None Specify this to provide a default level of security between the managed servers and the domain manager. If you specify <i>None</i>, communication between the managed servers and the domain manager is encrypted and no application other than EWLM can communicate with the domain manager. However, it does not prevent a managed server from connecting to a domain manager to which it should not have access. As a general rule, you should not use this value in a production environment. To provide a greater level of security, you should create and use your own SSL certificates.</p> <p>ServerSSL Specify this if you are using server SSL to secure communications between the managed servers and the domain manager. Server SSL authenticates the domain manager to the managed servers and encrypts the data that flows between the servers.</p> <p>ClientServerSSL Specify this if you are using client/server SSL to secure communications between a managed server and the domain manager. Client/server SSL authenticates the domain manager to the managed server, authenticates the managed server to the domain manager, and encrypts the data that flows between the servers.</p> <p>The -auth value you specify on the createMS command must match the -auth value you specify when you configure the domain manager.</p>	
configID	<p>A unique name for this managed server instance. The configuration ID must be an alphanumeric character string. Valid characters are uppercase letters A-Z, lowercase letters a-z, and numbers 0-9. The length of the ID must not exceed 32 characters.</p>	
-ma	<p>The IP address or host name of the domain manager. If the managed server accesses the domain manager through a SOCKS server, then this value is the IP address or host name of the firewall broker.</p> <p>This value must match the -ma value that you specify when you configure the domain manager or firewall broker, depending on which applies. This allows the managed server to properly identify and communicate with its domain manager either directly or through the firewall broker.</p>	
-mp	<p>The port that the domain manager or firewall broker uses to listen for managed server connection requests. This value must match the -mp value that you specify when you configure the domain manager or firewall broker, depending on which applies.</p>	

Table 30. Parameters for createMS command (continued)

Parameter	Description	Value
-sslks	The path to the certificate keystore that contains the SSL keys that are used for securing the communications between the managed servers and the domain manager or firewall broker, if applicable, when -auth is ServerSSL or ClientServerSSL. Note: This parameter is only required if you specify ServerSSL or ClientServerSSL on the -auth parameter.	
-sslpw	The password used to access the keystore that you specified on the -sslks parameter.	

Related concepts

“Planning for EWLM” on page 79

Use these planning worksheets before you install and configure EWLM.

“Command usage considerations” on page 91

View this topic to learn about the syntax required when issuing an EWLM command. As you complete the EWLM installation and configuration tasks, you may need to run an EWLM command.

When you specify an EWLM command, the location from which you run the command and the syntax used to format the command varies by operating system. In addition, the commands that are supported by each operating system vary.

“Configuration wizard usage considerations” on page 106

View this topic to learn about the operating systems that support the EWLM configuration wizards. In addition, learn about the syntax required when you run an EWLM command that launches an EWLM configuration wizard. As you complete the EWLM installation and configuration tasks, you may choose to use the configuration wizards.

“EWLM managed server” on page 12

A *managed server* is a server or server instance that EWLM monitors.

Related tasks

“Configuring a managed server on z/OS” on page 179

Configuring an EWLM managed server on z/OS is different from configuring a managed server on other platforms. View this topic to learn how to configure the EWLM managed server on z/OS.

Creating a firewall broker

View this topic to learn how to create a firewall broker. You need to create a firewall broker to allow EWLM communication between the domain manager and managed server, if the managed server accesses the domain manager by using a SOCKS server.

Ensure that you have completed the firewall broker planning worksheet. Use the values from the worksheet when you create the firewall broker. In addition, ensure the user responsible for configuring the firewall broker has the proper user authorities.

If a managed server must access its domain manager through a SOCKS server, you must either run the **EWLM managed server configuration wizard**, which provides a path for firewall broker setup, or the createFB configuration command to create and configure a firewall broker. You can create the firewall broker on one of the managed servers or some other server in the same trusted zone as the managed servers.

Use one of the following methods to create a firewall broker:

- Run the EWLM managed server configuration wizard
- Run createFB

Run the EWLM managed server configuration wizard:

1. Go to the `$(install)/IBM/VE2/EWLMMS/bin` directory where `$(install)` maps to the platform-specific default directory in which products are typically installed. If you installed the firewall broker to a different directory, then run the command from the `EWLMMS/bin` subdirectory of that directory.
2. Run the following command:

```
configWizardMS
```

For more information about starting the wizard on your operating system, if supported, see, "Configuration wizard usage considerations" on page 106.
3. Complete the wizard to configure the firewall broker. Click **Help** if you need assistance.

Run createFB:

Run this command, replacing the example parameters shown in italics with your own.

```
createFB configID -ma address -mp port
          -da address -dp port -fp port
          -auth [None | ServerSSL | ClientServerSSL]
          -sslks path -sslpw password
```

For more information about issuing the command on your operating system, if supported, see, "Command usage considerations" on page 91.

The following table describes each of the parameters and provides a place for you to enter your specific values when you print this page for reference.

Table 31. Parameters for createFB command

Parameter	Description	Value
-auth	<p>The authority level used to secure communications between the firewall broker and the managed servers as well as between the firewall broker and the domain manager. Possible values follow:</p> <p>None Specify this to provide a default level of security between the firewall broker and the managed servers. If you specify <i>None</i>, communication between the firewall brokers and the managed servers is encrypted and no application other than EWLM can communicate with the firewall broker. However, it does not prevent a managed server from connecting to a firewall broker to which it should not have access. As a general rule, you should not use this value in a production environment. To provide a greater level of security, you should create and use your own SSL certificates.</p> <p>ServerSSL Specify this if you are using server SSL to secure communications between the managed servers and the firewall broker. Server SSL authenticates the firewall broker to the managed servers and encrypts the data that flows between the servers.</p> <p>ClientServerSSL Specify this if you are using client/server SSL to secure communications between a managed server and the firewall broker. Client/server SSL authenticates the firewall broker to the managed server, authenticates the managed server to the firewall broker, and encrypts the data that flows between the servers.</p> <p>This value must match the -auth value you specify when you configure each managed server that will use this firewall broker.</p>	

Table 31. Parameters for createFB command (continued)

Parameter	Description	Value
configID	A unique name for the firewall broker. The configuration ID must be an alphanumeric character string. Valid characters are uppercase letters A-Z, lowercase letters a-z, and numbers 0-9. The length of the ID must not exceed 32 characters.	
-da	The IP address or host name of the domain manager. This value must match the -ma parameter you specify on the createDM command.	
-dp	The port that the domain manager uses to listen for connection requests from the firewall broker. This value must match the -fp parameter you specify on the changeDM command.	
-fp	The port that the firewall broker uses to listen for connection requests from the domain manager. This value must match the port you specify when you list this firewall broker on the -fb parameter of the changeDM command.	
-ma	The IP address or the host name that the firewall broker uses for connections requests from managed servers in the same trusted zone. The -ma value that you specify here must match the -ma value that you will specify when you configure each managed server that will use this firewall broker to access the domain manager through a SOCKS server.	
-mp	The port that the firewall broker uses to listen for connection requests from the managed servers in the same trusted zone. The -mp value that you specify here must match the -mp value that you will specify when you configure each managed server that will use this firewall broker to access the domain manager through a SOCKS server.	
-sslks	The path to the certificate keystore that contains the SSL keys that are used for securing the following communications: <ul style="list-style-type: none"> Between the firewall brokers and the managed server when -auth is ServerSSL or ClientServerSSL. Between the firewall brokers and domain manager when -auth is ServerSSL or ClientServerSSL. <p>Note: This parameter is only required if you specify ServerSSL or ClientServerSSL on the -auth parameter.</p>	
-sslpw	The password used to access the keystore that you specified on the -sslks parameter.	

Related concepts

“Planning for EWLM” on page 79

Use these planning worksheets before you install and configure EWLM.

“Command usage considerations” on page 91

View this topic to learn about the syntax required when issuing an EWLM command. As you complete the EWLM installation and configuration tasks, you may need to run an EWLM command. When you specify an EWLM command, the location from which you run the command and the syntax used to format the command varies by operating system. In addition, the commands that are supported by each operating system vary.

“Configuration wizard usage considerations” on page 106

View this topic to learn about the operating systems that support the EWLM configuration wizards. In addition, learn about the syntax required when you run an EWLM command that launches an EWLM configuration wizard. As you complete the EWLM installation and configuration tasks, you may choose to use the configuration wizards.

“Firewall broker” on page 15

A *firewall broker* provides the ability to allow communication through a firewall.

Assigning users to EWLM Control Center roles

View this topic to learn how to assign user IDs that exist on the domain manager’s local user registry to access the EWLM Control Center.

To administer EWLM with the EWLM Control Center, users must have a valid user ID on the server where the domain manager and EWLM Control Center functions run.

Then, you must assign each user ID to an EWLM role based on the function the owner of the user ID will perform when using the EWLM Control Center. The user ID must be granted access to the Control Center by assigning it to an EWLM role. Otherwise, the user cannot log in to the EWLM Control Center. A user can also belong to a group, which can then, in turn, be assigned an EWLM role. All users in a group are assigned the role of the group. The EWLM roles follow:

Administrator

This role has access to all functions. A user assigned to this role can perform setup, manage, and monitor tasks

Operator

This role provides access to the managing and monitoring tasks listed in the EWLM Control Center. A user assigned to this role can manage service policies, view the status of the managed servers in an EWLM domain, and view reports.

Monitor

This role provides access to the monitoring tasks. A user assigned to this role can view reports that contain performance data for the EWLM domain. This role has access to only the monitoring functions of the EWLM Control Center.

To assign user IDs to EWLM Control Center roles, use one of the following methods:

- Run the EWLM domain manager configuration wizard
- Run `changeCC`

If the WebSphere Application Server instance is active when you run the command, then the EWLM Control Center is restarted so that your changes take effect immediately. If the WebSphere Application Server is stopped when you run the command, then the server is left in the stopped state when the command finishes running.

• Run the EWLM domain manager configuration wizard

1. Go to the `$(install)/IBM/VE2/EWLM/bin` directory where `$(install)` maps to the platform-specific default directory in which products are typically installed. If you installed the domain manager to a different directory, then run the command from the `EWLM/bin` subdirectory of that directory.
2. Run the following command:
`configWizardDM`
For more information about starting the wizard on your operating system, if supported, see, “Configuration wizard usage considerations” on page 106.
3. Select the **Add or remove users to EWLM Control Center roles** task. Complete the wizard as indicated. Click **Help** if you need assistance.

• Run `changeCC`

To assign users to EWLM Control Center roles, run the following command, replacing the example parameters shown in *italics* with your own and adding the appropriate command extension.

```
changeCC -addUser configID -adminUser userid -adminPW password  
-roleUser user -role [Administrator | Monitor | Operator]
```

To assign groups to EWLM Control Center roles, run this command, replacing the example parameters shown in *italics* with your own and adding the appropriate command extension.

```
changeCC -addGroup configID -adminUser userid -adminPW password
        -roleGroup group -role [Administrator | Monitor | Operator]
```

For more information about issuing the command on your operating system, if supported, see, “Command usage considerations” on page 91.

The following table describes each of the parameters and provides a place for you to enter your specific values if you print this page for reference.

Table 32. Parameters for changeCC -addUser and changeCC -addGroup scripts

Parameter	Description	Value
-adminPW	The password associated with the -adminUser parameter.	
-adminUser	The WebSphere Application Server administrator user ID. This must be the same user ID that you specified when you ran the createDM command or EWLM domain manager configuration wizard.	
-role	The role you want to assign to the users or groups you specified in the -roleUser or -roleGroup parameter. Possible values are: Administrator, Operator, or Monitor. These values are case-sensitive.	
-roleGroup	The group or groups that you want to access the EWLM Control Center. The same syntactical rules apply to the -roleGroup parameter as for the -roleUser parameter.	
-roleUser	The user or users that you want to access the EWLM Control Center. Specify multiple user IDs by separating each user ID with a colon (:). For example, user1:user2:user3. When a user ID is used to log into the EWLM Control Center, you must specify the same user ID as provided for this parameter. Note that the EWLM Control Center log in field is case sensitive.	
configID	The configuration identifier that EWLM uses to identify the EWLM domain manager. You defined this value when you create the domain manager. You cannot change this value.	

Related concepts

“EWLM Control Center” on page 13

The EWLM Control Center is a Web-based user interface that allows you to manage and monitor the performance of servers in an EWLM management domain. It is installed with and hosted by the domain manager.

Related tasks

“Starting EWLM on the systems in your domain” on page 189

View this topic to learn how to start EWLM on the components of EWLM that you have installed and configured in your EWLM domain. The components of EWLM that you can install and configure include the domain manager, managed servers, and firewall brokers.

“Starting the EWLM Control Center” on page 192

View this topic to learn how to start the EWLM Control Center.

“Removing EWLM Control Center users or groups” on page 281

Use this topic to learn how to remove EWLM Control Center access for a user ID.

Authorizing Windows users to ARM services

View this topic to learn how to authorize users to ARM services. If you have an ARM-instrumented application that you want EWLM to monitor, the user ID under which the application runs must be able to access ARM services.

The ARM-instrumented application must run under the **Local System Account**, or under a user that belongs to either the **Administrator** group or the **EWLMArm4Users** group. To minimize security exposures, IBM recommends to add users to the **EWLMArm4Users** group.

| **Note:** You should not change the logon account for any of the EWLM Windows services. In addition, do
| not change the user ID, the password, or the privileges of the user IDs that are created by and for
| EWLM such as eWLMDomainManager and eWLMFirewallBroker. Changing any of the properties
| may cause problems for the domain manager, the firewall broker or the managed server.

| **Enabling ARM services on Windows**

| View this topic to learn how to enable ARM services. ARM services must be enabled for the EWLM
| managed server to function properly.

| ARM services is enabled by default. However, you must also ensure the following conditions are adhered
| to:

- | • The ARM-instrumented application must dynamically bind to the ARM library routine, libarm4.dll, to
| take advantage of the functions provided by ARM 4.0 and EWLM. During the EWLM managed server
| installation, EWLM installs the libarm4.dll to %SystemRoot%\system32 directory. The application must
| bind to this library because it is specific to EWLM.
- | • The user ID that starts the application must be a member of the **eWLMArm4Users** group as described
| in “Authorizing Windows users to ARM services” on page 166.

| **Configuring EWLM on z/OS**

| View this topic to learn how to configure the components of EWLM that you installed. In addition, learn
| how to enable functions of EWLM that are optional.

| The following topics provide step-by-step instructions on how to configure and enable various functions
| of EWLM. Note that not all of these tasks are required. You need to complete only the configuration tasks
| that are applicable to your environment.

| **Updating configuration files on z/OS**

| View this topic to learn how to update the configuration files on z/OS.

| After installing a z/OS domain manager, managed server, or firewall broker and before configuring and
| starting it, you need to update the appropriate configuration file. To update the configuration file,
| complete the following.

| • **Domain manager**

- | 1. Ensure that IBM Software Development Kit (SDK) for z/OS, Java 2 Technology Edition, Version 1.4
| is installed. If you installed the domain manager by using the program directory for IBM
| Virtualization Engine Enterprise Workload Manager for z/OS V2R1, then the SDK is already
| installed.
- | 2. Log on as a user with superuser authority.
- | 3. Go to the /usr/lpp/VE_R2/EWLM/samples directory and run this command to copy the
| configuration file to the etc directory:
|
| `cp ewlmdm_environment.conf /etc`
- | 4. Go to the etc directory and update the configuration file, ewlmdm_environment.conf. Specify
| values as described below.

| **EWLM_DATA_ROOT**

| names a path that must already exist. The user ID under which the domain manager process
| runs must have read/write access to this directory.

| **JREBIN_ROOT**

| is the SDK path. If you installed the SDK as specified in the program directory for IBM
| Virtualization Engine Enterprise Workload Manager for z/OS, or with a ServerPac, then the
| path where the SDK resides is /usr/lpp/java/J1.4/bin. If you installed to a different directory,
| then specify that directory.

VE_BASE

is the path where the Virtualization Engine configuration file will be saved when you run the `installEWLM` command as part of configuring a domain manager.

VE_LIB

is the path of the JAR files needed by the `installEWLM` command. This should be set to `/usr/lpp/VE_R2/VE_LIB/classes`. This setting activates the GCR support in the `createDM` command.

WEBSHERE_ROOT

is the Virtualization Engine WebSphere for z/OS HFS directory.

• Managed server and firewall broker

1. Ensure that IBM SDK for z/OS, Java 2 Technology Edition, Version 1.4 is installed. If you installed the managed server by using the program directory for IBM Virtualization Engine Enterprise Workload Manager for z/OS V2R1, then the SDK is already installed.
2. Log on as a user with superuser authority.
3. Go to the `/usr/lpp/VE_R2/EWLMMS/samples` directory and run this command to copy the configuration file to the `etc` directory:

```
cp ewlmms_environment.conf /etc
```

Note that the managed server and the firewall broker use the same configuration file.

4. Go to the `etc` directory and update the configuration file, `ewlmms_environment.conf`. Update the SDK path variable used by EWLM, `JRE_ROOT`, for example:

```
JRE_ROOT=/usr/lpp/java/J1.4/bin
```

If you installed the SDK as specified in the program directory for IBM Virtualization Engine Enterprise Workload Manager for z/OS V2R1, or with a ServerPac, then the path where the SDK resides is `/usr/lpp/java/J1.4/bin`. If you installed to a different directory, then specify that directory.


Updating the BPXPRMxx member in PARMLIB

View this topic to learn how to update the BPXPRMxx member in PARMLIB on z/OS.

EWLM on z/OS requires z/OS UNIX domain (AF_UNIX) sockets. You should evaluate the values in your BPXPRMxx member of PARMLIB to be sure that they identify this type of file system. You may want to use values similar to those in the following example.


```
FILESYSTYPE TYPE(UDS) ENTRYPPOINT(BPXTUJNT)
NETWORK DOMAINNAME(AF_UNIX)
          DOMAINNUMBER(1)
          MAXSOCKETS(10000)
          TYPE(UDS)
```

For more information on defining file systems, see *z/OS UNIX System Services Planning*

(<http://publibz.boulder.ibm.com/epubs/pdf/bpxzb260.pdf>). 

For more information on the BPXPRMxx member, see *z/OS MVS Initialization and Tuning Reference*

(<http://publibz.boulder.ibm.com/epubs/pdf/iea2e260.pdf>). 

Note: You need Adobe Reader installed on your system to view or print these PDFs. You can download a free copy from the Adobe Web site (www.adobe.com/products/acrobat/readstep.html) .

Configuring a domain manager on z/OS

View this topic to learn how to configure a domain manager on z/OS.

Ensure that you have completed the domain manager worksheets found in the topic “Planning for EWLM” on page 79. Use the values from the planning worksheet when you configure the domain manager. In addition, ensure that the user responsible for configuring the domain manager has the proper user authorities.

The following Unix shell commands are necessary to configure your system after you install the domain manager.

installEWLM

Run this optional but highly recommended command to add this system to the Virtualization Engine Global Configuration Repository (GCR) and Identity Management Repository (IMR). Participation in the GCR and IMR is vital to future Virtualization Engine components on z/OS. If you choose to not run installEWLM, any domain managers created without the GCR will need to be stopped, deleted, and then recreated before the domain manager can use the GCR and IMR.

To save registration entries for this system in the GCR and IMR, the domain manager uses an IBM Tivoli® Directory Server (ITDS) LDAP server. ITDS does not run on z/OS, but may be installed on other systems. It can be installed with the Virtualization Engine installation wizard.

The installEWLM command also creates a Virtualization Engine configuration file that the domain manager uses.

configWAS and createWAS

Run these commands to create and configure the Virtualization Engine WebSphere Application Server profile.

createDM

Run this command to configure the domain manager.

updateWAS

Run this command to accept a configuration file name parameter that has the name of a previously built Virtualization Engine WebSphere Application Server instance and apply any needed updates that are associated with the latest service level.

Follow the steps for each command to run it on your system.

1. Run installEWLM

Run the installEWLM command, replacing the example values shown in italics with your own:

```
installEWLM -GCRUser adminuser -GCRPW adminpw -GCRHost gcrLdapHostname
-GCRPort gcrLdapPort -GCREnv gcrEnvironmentname
-IMRUser adminuser -IMRPW adminpw -IMRHost gcrLdapHostname
-IMRPort imrLdapPort -IMRDN imrDomainname
```

For details on running commands on z/OS, see “Command usage considerations” on page 91.

The following table describes each of the parameters and provides a place for you to enter your specific values when you print this page for reference.

Table 33. Parameters for the installEWLM command.

Parameter	Description	Value
-GCRUser	The Global Configuration Repository (GCR) LDAP server administer ID.	
-GCRPW	The GCR LDAP server administer password.	
-GCRHost	The GCR LDAP server host name.	
-GCRPort	The GCR LDAP server port number.	
-GCREnv	The GCR environment name to assign to this Virtualization Engine configuration. The name you specify here must match the GCR environment name specified for the other systems that are to share the GCR.	

Table 33. Parameters for the `installEWLM` command. (continued)

Parameter	Description	Value
<code>-IMRUser</code>	The Identity Management Repository (IMR) LDAP server administrator ID. If omitted, this parameter defaults to the value specified for the <code>-GCRUser</code> parameter.	
<code>-IMRPW</code>	The IMR LDAP server administrator password. If omitted, this parameter defaults to the value specified for the <code>-GCRPW</code> parameter.	
<code>-IMRHost</code>	The IMR LDAP server host name. If omitted, this parameter defaults to the value specified for the <code>-GCRHost</code> parameter.	
<code>-IMRPort</code>	The IMR LDAP server port number. If omitted, this parameter defaults to the value specified for the <code>-GCRPort</code> parameter.	
<code>-IMRDN</code>	The IMR EIM domain name. If omitted, this parameter defaults to IBM identity mapping repository. The name you specify here must match the EIM domain name specified for the other systems that are to share the IMR.	
<code>-IMRParentDN</code>	The IMR parent domain name under which you should create the IMR domain name. Use this when the IMR domain must exist under something other than the default of <code>cn=root</code> . Use the format of <code>cn=xxxxxxx</code> where <code>xxxxxxx</code> is the name of the parent domain name.	

Usage notes:

- The `installEWLM` command creates the configuration file `vecd` in the directory `vebase/.veconfig`, where `vebase` is a path specified with the `VE_BASE` variable in the `ewlmdm_environment.conf` configuration file located in the `/etc` directory.
 - You can run this command more than once, but you must first delete the VE configuration file, `vecd`, before rerunning the command.
 - The log for the `installEWLM` command is in directory `ve_base/VE_DIAGNOSTICS/RegisterVE.date-time-stamp`.
2. Optional. Install the EWLM bridge to Virtualization Engine console, which allows a Virtualization Engine console HealthCenter portlet to view dynamic EWLM domain manager data on various systems that include z/OS. See *Installing and configuring the EWLM bridge to Virtualization Engine console for z/OS for instructions.*

3. **Run configWAS**

Run the following command to create a Virtualization Engine WebSphere Application Server configuration file, replacing the example values shown in italics with your own.

```
configWAS -cfg filename.cfg
```

For details on running an EWLM command on z/OS, see “Command usage considerations” on page 91.

The following table describes each of the parameters and provides a place for you to enter your specific values when you print this page for reference.

Table 34. Parameters for the `configWAS` command.

Parameter	Description	Value
<code>-cfg</code>	The optional name of an existing configuration file to use as input. Specifying an existing configuration file allows the command to display the current values, so that you can review and modify them. If no configuration file is specified, <code>configWAS</code> creates one.	

The configWAS command prompts you for values. The following table provides tips for responding to some of the prompts.

Table 35. Tips for responding to configWAS prompts

Prompt	Tip
Enter the name of the WAS PRODUCT HFS dataset (SBBWSHFS):	Specify the name where the Virtualization Engine WebSphere Application Server HFS is located. This is the dataset that was SMP/E installed with Virtualization Engine WebSphere Application Server.
Enter the path at which this HFS should be mounted:	Specify the name of the directory where the Virtualization Engine WebSphere Application Server product HFS is to be mounted. This directory will be created by the createWAS command if it does not already exist.
Enter the name of the CONFIG HFS dataset to create:	Specify the name to use when creating the Virtualization Engine WebSphere Application Server configuration HFS. The createWAS command creates this HFS and copies the pre-built Virtualization Engine WebSphere Application Server configuration into it.
Enter the path at which this HFS should be mounted:	Specify the directory used when mounting the Virtualization Engine WebSphere Application Server configuration HFS. It is created by the createWAS command.
Enter the userid to create for the WAS instance:	Specify the started task user ID that the Virtualization Engine WebSphere Application Server instance will start and run with. If this user ID does not already exist, it will be defined to RACF by the createWAS command in a later step. The password for this user ID is set to the user ID. (If the user ID already exists, the password is changed to the user ID.) This user ID is also permitted to the Virtualization Engine WebSphere Application Server administrative console as an administrator.
Enter the numerical UNIX UID to assign to this userid:	Specify the numeric z/OS Unix System Services user identifier for the user ID. Commands such as createDM that you run when logged on with this user ID require a UID of 0. If you don't specify a value of 0 now, you will need to modify the user ID later.
Enter the group to create for the WAS instance:	Specify the group ID for the started task security group that the Virtualization Engine WebSphere Application Server user ID is in.

Note: The configWAS command must be run under an OpenEdition MVS (OMVS) or Telnet session. Other methods, such as running the command under ISHELL, are not supported.

4. Run createWAS

Run the following command to create a Virtualization Engine WebSphere Application Server profile, replacing the example values shown in italics with your own.

```
createWAS -cfg filename.cfg -sec path
```

For details on running commands on z/OS, see “Command usage considerations” on page 91.

The following table describes each of the parameters and provides a place for you to enter your specific values when you print this page for reference.

Table 36. Parameters for the createWAS command.

Parameter	Description	Value
-cfg	The name of the configuration file created by configWAS.	

Table 36. Parameters for the createWAS command. (continued)

Parameter	Description	Value
-sec	<p>The optional path of the REXX file that contains commands to define security for the Virtualization Engine WebSphere Application Server. The possible values for -sec are:</p> <p>None Specify this if the security definition file should be created but not run.</p> <p><i>path</i> Specify this to use an existing security definition file indicated by <i>path</i>.</p> <p>If you omit -sec, the security definition file is created and run. The file is created in directory <i>/config_root_dir/misc/RACFMSTR</i>, where <i>config_root_dir</i> is the mountpoint directory for the configuration HFS in the configWAS command.</p> <p>You might want to run createWAS first with None and then review the generated security definition file. If the file is acceptable as is, you can then run createWAS again, specifying the name of the file. If the file must be modified, make a copy of it, modify the file as required, then run createWAS again, specifying the name of the updated file.</p> <p>Note that if you have made changes to the Virtualization Engine WebSphere Application Server configuration outside of the command, running createWAS will override those changes.</p>	

Usage notes:

- a. The createWAS command must be run under an OMVS or Telnet session. Other methods, such as running the command under ISHELL, are not supported.
- b. The prefix for 10 free TCP/IP ports specified in the configWAS command is used to reserve 10 ports in the Virtualization Engine WebSphere Application Server. The port ending in 8 (for example, 32208 where 3220 is the prefix) is the port where the WebSphere administrative console may be accessed.
- c. The user ID defined in the configWAS command as the ID for the WebSphere Application Server instance is used in the createWAS command is associated with the Virtualization Engine WebSphere Application Server started task procedures, and can be used to log on to the WebSphere administrative console. The password is set to the user ID. The user ID is defined to RACF if it does not already exist. You must be logged on with that ID to run createDM, changeCC, changeCCapp, displayCC, or stopWAS. If the user ID already exists, the createWAS command changes the password to be the same as the user ID, sets the password to non-expiring, changes the ID's z/OS Unix System Services UID, and changes the FILEPROC attributes.
- d. To grant a user ID other than the user ID specified in the configWAS command access to the WebSphere Administrative console, run the following RACF commands for that user ID, replacing the values shown in italics with your own:


```
PERMIT cellname.ADMINISTRATOR CLASS(EJBROLE) ID(userid) ACCESS(READ)
SETROPTS RACLIST(EJBROLE) REFRESH
```
- e. The Virtualization Engine WebSphere Application Server daemon address space (address space *xxxxxxD*, where *xxxxxx* is the started task name specified with the configWAS command) will not stop in response to a stopServer request. (EWLM requests may run stopServer and startServer.) To stop the daemon address space, you must run the MVS STOP command, P *xxxxxxD*, from a system console.

- f. If you use RACF security labels for multilevel security and receive an error message during Virtualization Engine WebSphere Application Server startup associated with missing security label definitions, you will need to run these RACF commands, replacing the values shown in italics with your own:

```
ALU userid SECLABEL(SYSMULTI)
PERMIT SYSMULTI CLASS(SECLABEL) ID(userid) ACCESS(READ)
SETROPTS RACLIST(SECLABEL) GENERIC(STARTED) REFRESH
```

For *userid*, use the user ID specified with the configWAS command as the user ID to create for the Virtualization Engine WebSphere Application Server instance.

- g. After createWAS completes successfully, you can verify the installation of Virtualization Engine WebSphere Application Server by starting the control and servant regions using the MVS START command:

```
S xxxxxx,JOBNAME=xxxxxx,ENV='cellname.NODE1.xxxxxx'
```

where *xxxxxx* is the name of the started task prefix used with configWAS. After the servant region completes initializing, you should be able to logon to the WebSphere administrative console with a Web browser using the following address: `http://hostname:xxxx8/ibm/console/` where *hostname* is the value specified for host name and *xxxx* is the value specified for TCP/IP port prefix and *cellname* is the name of the cell used with the configWAS command. To stop the Virtualization Engine WebSphere Application Server address spaces, run the following MVS STOP command:

```
P xxxxxxD
```

5. Run createDM

Run the following command, replacing the example values shown in italics with your own.

```
createDM configID -adminUser userid -adminPW password -wasProfile profilePath -jp port
      -ma address -mp port -dn domainName -auth [None | ServerSSL | ClientServerSSL]
      -sslks path -sslpw password
```

For details on running commands on z/OS, see “Command usage considerations” on page 91.

The following table describes each of the parameters and provides a place for you to enter your specific values when you print this page for reference.

Table 37. Parameters for the createDM command.

Parameter	Description	Value
-adminUser	The WebSphere Application Server administrator user ID. This user ID must conform to the WebSphere Application Server requirements for an administrator. Note: If you plan to run the command from an OMVS session under Time Sharing Option (TSO), ensure that the -adminUser also has TSO access and that you logon with this ID when running this command. If you are running this under a Telnet session, then you will not need TSO access, but you will still need to execute createDM while signed on with the -adminUser userid.	
-adminPW	The password associated with the -adminUser parameter.	

Table 37. Parameters for the createDM command. (continued)

Parameter	Description	Value
-auth	<p>The authority level used to secure communications between the domain manager and the managed servers and, if they exist, the firewall brokers. Possible values follow:</p> <p>None Specify this to provide a default level of security between the managed servers and the domain manager. If you specify None, communication between the managed servers and the domain manager is encrypted and no application other than EWLM can communicate with the domain manager. However, it does not prevent a managed server from connecting to a domain manager to which it should not have access. As a general rule, you should not use this value in a production environment. To provide a greater level of security, you should create and use your own SSL certificates.</p> <p>ServerSSL Specify this if you are using server SSL to secure communications between the managed servers and the domain manager. Server SSL authenticates the domain manager to the managed servers and encrypts the data that flows between the servers.</p> <p>ClientServerSSL Specify this if you are using client/server SSL to secure communications between a managed server and the domain manager. Client/server SSL authenticates the domain manager to the managed server, authenticates the managed server to the domain manager, and encrypts the data that flows between the servers.</p> <p>The -auth value you specify on the createDM command must match the -auth value that you specify when you configure each managed server and firewall broker, if applicable.</p>	
configID	A unique name for this domain manager instance. The configuration ID must be an alphanumeric character string. Valid characters are uppercase letters A-Z, lowercase letters a-z, and numbers 0-9. The length of the ID must not exceed 32 characters.	
-dn	The name of the EWLM domain. The EWLM domain name must be a character string. All characters are valid except the colon (:), question mark (?) symbol, double quote (") symbol, greater than (>) symbol, less than (<) symbol, or bar () symbol. The length of the ID must not exceed 64 characters.	
-jp	The port that the EWLM Control Center uses to communicate with the domain manager. Ensure that this port is not currently in use and is not the same port that you specify for -mp or one of the 15 contiguous ports that -wasPorts requires.	
-ma	The IP address or host name of the domain manager server. The -ma value you specify on the createDM command must match the -ma value that you specify when you configure each managed server. This allows the managed server to properly identify and communicate with its domain manager.	
-mp	The port that the domain manager uses to listen for managed server connection requests. The -mp value that you specify on the createDM command must match the -mp value you specify when you configure each managed server.	

Table 37. Parameters for the createDM command. (continued)

Parameter	Description	Value
-sslks	<p>The path of the certificate keystore that contains the SSL keys that are used to secure the following communications:</p> <ul style="list-style-type: none"> Between the domain manager and the managed servers when <code>-auth</code> is <code>ServerSSL</code> or <code>ClientServerSSL</code>. Between the domain manager and firewall brokers, if they exist in the EWLM domain, when <code>-auth</code> is <code>ServerSSL</code> or <code>ClientServerSSL</code>. Between the domain manager and load balancers if you provide a port value for the <code>-lbs</code> parameter. <p>Note: You set the <code>-lbs</code> parameter on the <code>changeDM</code> command.</p> <p>Note: This parameter is only required if you specify <code>ServerSSL</code> or <code>ClientServerSSL</code> on the <code>-auth</code> parameter.</p>	
-sslpw	The password used to access the keystore you specified on the <code>-sslks</code> parameter, if applicable.	
-wasProfile	The directory of the WebSphere Application Server profile created by <code>createWAS</code> , to be used by the EWLM domain manager. For example, if the HFS you want to use for this domain manager instance is at <code>/config</code> , then you would specify <code>/config/AppServer/profiles/default</code> for this parameter.	

If `createDM` fails, before you run it again, you must create a new Virtualization Engine WebSphere Application Server profile to pass to `createDM`. To do this, complete the following:

- a. Run `deleteWAS`.
- b. Run `createWAS`.

For more information about `deleteWAS`, see “Deleting a Virtualization Engine WebSphere Application Server profile on z/OS” on page 324

6. Run `updateWAS`

Run the following command to update a Virtualization Engine WebSphere Application Server instance that existed prior to the latest Virtualization Engine WebSphere Application Server fix level, replacing the example values shown in italics with your own.

```
updateWAS -cfgdir was_config_hfs_root
          -prddir was_product_hfs_root
```

For details on running commands on z/OS, see “Command usage considerations” on page 91.

The following table describes each of the parameters and provides a place for you to enter your specific values when you print this page for reference.

Table 38. Parameters for the updateWAS command.

Parameter	Description	Value
-cfgdir	The required parameter that contains the path name for the WebSphere Application Server profile’s configuration root for the Virtualization Engine Console Bridge instance. An example of a possible value is <code>/veconfig</code> . Note that this is the value that was used for the configuration directory during <code>configWAS</code> processing.	
-prddir	The optional parameter that contains the path name for the WebSphere Application Server product HFS root. An example of a possible value is <code>/VEWASProductDir</code> . The default value is <code>/usr/lpp/VE_R2/VEWAS</code> .	

Usage note: This command is designed so that you can safely run and rerun it against any level of a Virtualization Engine WebSphere Application Server configuration file that was built at the same or an older level. When a service update is applied, it may or may not specify that you must run this command before you start Virtualization Engine WebSphere Application Server with the associated new service level.

Related concepts

“Planning for EWLM” on page 79

Use these planning worksheets before you install and configure EWLM.

“Command usage considerations” on page 91

View this topic to learn about the syntax required when issuing an EWLM command. As you complete the EWLM installation and configuration tasks, you may need to run an EWLM command.

When you specify an EWLM command, the location from which you run the command and the syntax used to format the command varies by operating system. In addition, the commands that are supported by each operating system vary.

Configuring z/OS for Virtualization Engine WebSphere for z/OS

View this topic to learn how to configure z/OS for Virtualization Engine WebSphere for z/OS. This is required if you have a domain manager on z/OS.

After configuring a z/OS domain manager, you need to configure z/OS for Virtualization Engine WebSphere for z/OS. To configure z/OS, complete the following.

- **Update BLSCUSER.** To use the IPCS support provided by Virtualization Engine WebSphere for z/OS, you must append the following to the contents of the BLSCUSER member in your IPCSPARM or PARMLIB.

```

/* ----- */
/* Sample BLSCUSER member. */
/* ----- */
/*
/*****
/* ----- */
/* IPCS Verb Exits */
/* ----- */

EXIT EP(BBORDATA) VERB(CBDATA)
/* ----- */
/* IPCS Data Struct */
/* ----- */
DATA STRUCTURE(ACRW) MODEL(BBORACRW) ENVIRONMENT(ALL); /* acrw */
DATA STRUCTURE(ASR) MODEL(BBORASR) ENVIRONMENT(ALL); /* asr */
DATA STRUCTURE(ASRE) MODEL(BBORASRE) ENVIRONMENT(ALL); /* asre */
DATA STRUCTURE(BACB) MODEL(BBORBACB) ENVIRONMENT(ALL); /* bacb */
DATA STRUCTURE(BGVT) MODEL(BBORBGVT) ENVIRONMENT(ALL); /* bgvt */
DATA STRUCTURE(BOAM) MODEL(BBORBOAM) ENVIRONMENT(ALL); /* boam */
DATA STRUCTURE(BOAX) MODEL(BBORBOAX) ENVIRONMENT(ALL); /* boax */
DATA STRUCTURE(BOBK) MODEL(BBORBOBK) ENVIRONMENT(ALL); /* bobk */
DATA STRUCTURE(BTCB) MODEL(BBORBTCB) ENVIRONMENT(ALL); /* btcb */
DATA STRUCTURE(DAUE) MODEL(BBORDAUE) ENVIRONMENT(ALL); /* daue */
DATA STRUCTURE(LSCB) MODEL(BBORLSCB) ENVIRONMENT(ALL); /* lscb */
DATA STRUCTURE(LSCP) MODEL(BBORLSCP) ENVIRONMENT(ALL); /* lscp */
DATA STRUCTURE(LSPC) MODEL(BBORLSPC) ENVIRONMENT(ALL); /* lspc */
DATA STRUCTURE(LSVT) MODEL(BBORLSVT) ENVIRONMENT(ALL); /* lsvt */
DATA STRUCTURE(TBUFSET) MODEL(BBORRTBS) ENVIRONMENT(ALL);
DATA STRUCTURE(TBUF) MODEL(BBORRTBF) ENVIRONMENT(ALL); /* tbuf */
DATA STRUCTURE(TRC) MODEL(BBORTRC) ENVIRONMENT(ALL); /* trc */
DATA STRUCTURE(SCOX) MODEL(BBORSCOX) ENVIRONMENT(ALL); /* scox */
DATA STRUCTURE(TMP) MODEL(BBORTMP) ENVIRONMENT(ALL); /* tmp */

```

- **Update CTIBBOxx.** To configure Virtualization Engine WebSphere for z/OS’s use of component trace (CTRACE), you must ensure the CTIBBOxx member of PARMLIB is set up properly, as shown in the following example:


```

| /* */
| /* FUNCTION: This is a sample parmlib member used to establish */
| /* defaults for WebSphere's use of Component Trace. */
| /* */
| /* TO USE: Copy this member into a data set in the system parmlib */
| /* concatenation and rename the copy to CTIBB0xx, where */
| /* "xx" is a value of your choice. This value must */
| /* match the component trace suffix you specify in the */
| /* WebSphere Application Server Customization Dialog. */
| /* */
| /* Change "procname" to the name of the external writer */
| /* cataloged procedure that will be used to write trace */
| /* data for WebSphere Application Server. */
| /* */
| /* If you want the external writer to start whenever */
| /* WebSphere Application Server does, remove the comments */
| /* around the WTRSTART parameter. */
| /* */
| /* DEFAULT CTIBB0xx MEMBER */
| /* ===== */
| TRACEOPTS
| /* ----- */
| /* -Start a ctrace writer. Remove comments to start the PROC */
| /* during WebSphere Application Server initialization. */
| /* ----- */
| /* WTRSTART(procname) */
| /* ----- */
| /* -Indicate that tracing is active for CB Series: */
| /* ----- */
| ON
| /* ----- */
| /* -Connect to ctrace external writer: */
| /* (Note that the WTR value must match the value for the started */
| /* ctrace external writer, wtrstart.) */
| /* ----- */
| WTR(procname)

```

- **Ensure that BBORTSS5 is in LPA.** The BBORTSS5 module is used by Virtualization Engine WebSphere for z/OS for component trace support. For CTRACE to work correctly, a copy of this module must be in the system link pack area (LPA). If a copy of BBORTSS5 is not currently loaded into LPA, you must either:

- Run the following MVS command to load BBORTSS5 into dynamic LPA:

```
SETPROG LPA,ADD,MODNAME(BBORTSS5),DSNAME(BBW.SBBWCOMB)
```
- Place the following statement in a PROGxx member of PARMLIB

```
LPA ADD MODNAME(BBORTSS5) DSNAME(BBW.SBBWCOMB)
```

and activate the member with the SET PROG= command after the IPL of the system is complete.

Make sure that the BBORTSS5 module is loaded into LPA after each IPL of the system.

- **Update SCHEDxx.** To set the correct program properties in Virtualization Engine WebSphere for z/OS runtime executables, you must:

1. Append the following to the contents of the SCHEDxx member in PARMLIB:

```

| PPT PGMNAME(BBODAEMN) /* PROGRAM NAME BBODAEMN */
| KEY(2) /* PROTECTION KEY 2 */
| NOSWAP /* NON-SWAPPABLE */
| PRIV /* PRIVILEGED */
| SYST /* SYSTEM TASK, NOT TIMED */
| CANCEL /* CANCEL ALLOWED */
| PPT PGMNAME(BBOCTL) /* PROGRAM NAME BBOCTL */
| KEY(2) /* PROTECTION KEY 2 */

```

```

|     NOSWAP                /* NON-SWAPPABLE      */
|     PRIV                  /* PRIVILEGED      */
|     CANCEL                /* CANCEL ALLOWED  */

```


2. Run the command SET SCH=(xx,xx) to activate SCHEDxx and load a new program properties table.

- **Prepare to collect SMF records.** If you want to collect the SMF type 120 records created by the runtime servers, update the SMFPRMxx member of PARMLIB as follows:
 - Update the SYS or SUBSYS(STC,...) statement for started tasks to include the type 120 record.
 - Optionally, specify designated subtypes 1 through 6 as shown in the following example:


```

|     SUBSYS(STC,EXITS(IEFU29,IEFACTRT),INTERVAL(SMF,SYNC),
|           TYPE(0,30,70:79,88,89,120,245))


```


For more information about the SMF records, see WebSphere Application Server for z/OS Information Center (<http://publib.boulder.ibm.com/infocenter/wasinfo/v6r0/index.jsp>). 


- **Review APF authorization.** Ensure that the following data sets are APF-authorized:
 - BBW.SBBWCOMB
 - CEE.SCEERUN
 - CEE.SCEERUN2
- **Reserve TCP/IP ports.** Update TCP/IP by reserving the following ports for Virtualization Engine WebSphere for z/OS.

Table 39. Reserved Ports. xxxx is the first four digits of the port definitions, for example, 3210.

Description	Port
SOAP JMX connector port	xxxx0
ORB port	xxxx1
ORB SSL port	xxxx2
HTTP port	xxxx8
HTTP SSL port	xxxx9
High availability manager communication port	xxxx5
Service integration port	xxxx3
Service integration secure port	xxxx4
Service integration MQ interoperability port	xxxx3
Service integration MQ interoperability secure port	xxxx4
Daemon IP port	xxxx6
Daemon SSL port	xxxx7

- **Ensure that required LE data sets are in the link list.** The following Language Environment® (LE) data sets must be in the system link list:
 - CEE.SCEERUN
 - PCEE.SCEERUN
- **Ensure that couple data sets have been defined and system logger has been set up.** For more information on defining couple data sets and setting up system logger, see z/OS MVS Setting Up a Sysplex (<http://publibz.boulder.ibm.com/zoslib/pdf/iea2f160.pdf>). 


Note: You need Adobe Reader installed on your system to view or print this PDF. You can download a free copy from the Adobe Web site (www.adobe.com/products/acrobat/readstep.html). 

- **Define the error log stream for Virtualization Engine WebSphere for z/OS.** For more information on defining the error log stream for Virtualization Engine WebSphere for z/OS, see z/OS MVS Setting Up a Sysplex (<http://publibz.boulder.ibm.com/zoslib/pdf/iea2f160.pdf>).  The following JCL illustrates how this might be accomplished:

```
//BBORCLGS EXEC PGM=IXCMIAPU
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
    DATA TYPE(LOGR)
    DEFINE LOGSTREAM NAME(XDOMAINX.ERROR.LOG)
        DASDONLY(YES)
        HLQ(LOGGER)
        LS_SIZE(3000)
        STG_SIZE(3000)
        MAXBUFSIZE(4096)
        AUTODELETE(YES)
        RETPD(1)
        LS_DATACLAS(STANDARD)
//
```


- **Ensure that RRS is active.** Look for the following console message to verify that Resource Recovery Services (RRS) was started:

```
ASA2011I RRS INITIALIZATION COMPLETE. COMPONENT ID=SCRRS
```

For more information on RRS, see z/OS MVS Programming: Resource Recovery (<http://publibz.boulder.ibm.com/zoslib/pdf/iea2n150.pdf>). 

- **Define the hosts file.** You must define an HFS hosts file, /SYSTEM/etc/hosts. The following is an example of a host file.

```
127.0.0.1 localhost
9.57.2.230 ALPS1229
9.57.2.230 ALPS1229.pok.ibm.com
```

For more information, see z/OS z/OS Communications Server IP Configuration Guide (<http://publibz.boulder.ibm.com/zoslib/pdf/f1a1b350.pdf>). 

Note: This step is necessary only if DNS is not available.

Configuring a managed server on z/OS

View this topic to learn how to configure a managed server on z/OS.

Ensure that you have completed the planning worksheet for the managed server. Use the values from the planning worksheet when you configure the managed server. In addition, ensure that the user responsible for configuring the managed server has the proper user authorities.

After you install the managed servers, you must create and configure each managed server in the management domain. To create and configure a management server on z/OS, you use the createMS configuration script.

Run createMS:

Run the following command, replacing the example values shown in *italics* with your own.

```
createMS configID -ma address -mp port -auth [None | ServerSSL | ClientServerSSL]
        -sslks path -sslpw password
```

For more information about issuing the command on your operating system, if supported, see, “Command usage considerations” on page 91.

The following table describes each of the parameters and provides a place for you to enter your specific values if you print this page for reference.

Table 40. Parameters for createMS command

Parameter	Description	Value
-auth	<p>The authority level used to secure communications between the domain manager and the managed servers and, if they exist, the firewall brokers. Possible values follow:</p> <p>None Specify this to provide a default level of security between the managed servers and the domain manager. If you specify <i>None</i>, communication between the managed servers and the domain manager is encrypted and no application other than EWLM can communicate with the domain manager. However, it does not prevent a managed server from connecting to a domain manager to which it should not have access. As a general rule, you should not use this value in a production environment. To provide a greater level of security, you should create and use your own SSL certificates.</p> <p>ServerSSL Specify this if you are using server SSL to secure communications between the managed servers and the domain manager. Server SSL authenticates the domain manager to the managed servers and encrypts the data that flows between the servers.</p> <p>ClientServerSSL Specify this if you are using client/server SSL to secure communications between a managed server and the domain manager. Client/server SSL authenticates the domain manager to the managed server, authenticates the managed server to the domain manager, and encrypts the data that flows between the servers.</p> <p>The -auth value you specify on the createMS command must match the -auth value you specify when you configure the domain manager.</p>	
configID	A unique name for this managed server instance. The configuration ID must be an alphanumeric character string. Valid characters are uppercase letters A-Z, lowercase letters a-z, and numbers 0-9. The length of the ID must not exceed 32 characters.	
-ma	<p>The IP address or host name of the domain manager. If the managed server accesses the domain manager through a SOCKS server, then this value is the IP address or host name of the firewall broker.</p> <p>This value must match the -ma value that you specify when you configure the domain manager or firewall broker, depending on which applies. This allows the managed server to properly identify and communicate with its domain manager either directly or through the firewall broker.</p>	
-mp	The port that the domain manager or firewall broker uses to listen for managed server connection requests. This value must match the -mp value that you specify when you configure the domain manager or firewall broker, depending on which applies.	

Table 40. Parameters for createMS command (continued)

Parameter	Description	Value
-sslks	The path to the certificate keystore that contains the SSL keys that are used for securing the communications between the managed servers and the domain manager or firewall broker, if applicable, when -auth is ServerSSL or ClientServerSSL. Note: This parameter is only required if you specify ServerSSL or ClientServerSSL on the -auth parameter.	
-sslpw	The password used to access the keystore that you specified on the -sslks parameter.	

Related concepts

“Planning for EWLM” on page 79

Use these planning worksheets before you install and configure EWLM.

“Command usage considerations” on page 91

View this topic to learn about the syntax required when issuing an EWLM command. As you complete the EWLM installation and configuration tasks, you may need to run an EWLM command.

When you specify an EWLM command, the location from which you run the command and the syntax used to format the command varies by operating system. In addition, the commands that are supported by each operating system vary.

Configuring a firewall broker on z/OS

View this topic to learn how to configure a firewall broker on z/OS. You need to configure a firewall broker to allow EWLM communication between the domain manager and managed server, if a firewall between the domain manager and managed server exists.

Ensure that you have completed the firewall broker planning worksheet. Use the values from the worksheet when you create the firewall broker. In addition, ensure the user responsible for configuring the firewall broker has the proper user authorities.

If a managed server must access its domain manager through a SOCKS server, you must create and configure a firewall broker. You can create the firewall broker on the managed server or some other server in the same trusted zone as the managed servers. To create and configure a firewall broker on z/OS, you run the createFB configuration command .

Run createFB:

Run this command, replacing the example parameters shown in italics with your own.

```
createFB configID -ma address -mp port
      -da address -dp port -fp port
      -auth [None | ServerSSL | ClientServerSSL]
      -sslks path -sslpw password
```

For more information about issuing the command on your operating system, if supported, see, “Command usage considerations” on page 91.

The following table describes each of the parameters and provides a place for you to enter your specific values when you print this page for reference.

Table 41. Parameters for createFB command

Parameter	Description	Value
-auth	<p>The authority level used to secure communications between the firewall broker and the managed servers as well as between the firewall broker and the domain manager. Possible values follow:</p> <p>None Specify this to provide a default level of security between the firewall broker and the managed servers. If you specify <i>None</i>, communication between the firewall brokers and the managed servers is encrypted and no application other than EWLM can communicate with the firewall broker. However, it does not prevent a managed server from connecting to a firewall broker to which it should not have access. As a general rule, you should not use this value in a production environment. To provide a greater level of security, you should create and use your own SSL certificates.</p> <p>ServerSSL Specify this if you are using server SSL to secure communications between the managed servers and the firewall broker. Server SSL authenticates the firewall broker to the managed servers and encrypts the data that flows between the servers.</p> <p>ClientServerSSL Specify this if you are using client/server SSL to secure communications between a managed server and the firewall broker. Client/server SSL authenticates the firewall broker to the managed server, authenticates the managed server to the firewall broker, and encrypts the data that flows between the servers.</p> <p>This value must match the -auth value you specify when you configure each managed server that will use this firewall broker.</p>	
configID	A unique name for the firewall broker. The configuration ID must be an alphanumeric character string. Valid characters are uppercase letters A-Z, lowercase letters a-z, and numbers 0-9. The length of the ID must not exceed 32 characters.	
-da	The IP address or host name of the domain manager. This value must match the -ma parameter you specify on the createDM command.	
-dp	The port that the domain manager uses to listen for connection requests from the firewall broker. This value must match the -fp parameter you specify on the changeDM command.	
-fp	The port that the firewall broker uses to listen for connection requests from the domain manager. This value must match the port you specify when you list this firewall broker on the -fb parameter of the changeDM command.	
-ma	The IP address or the host name that the firewall broker uses for connections requests from managed servers in the same trusted zone. The -ma value that you specify here must match the -ma value that you will specify when you configure each managed server that will use this firewall broker to access the domain manager through a SOCKS server.	

Table 41. Parameters for createFB command (continued)

Parameter	Description	Value
-mp	The port that the firewall broker uses to listen for connection requests from the managed servers in the same trusted zone. The -mp value that you specify here must match the -mp value that you will specify when you configure each managed server that will use this firewall broker to access the domain manager through a SOCKS server.	
-sslks	The path to the certificate keystore that contains the SSL keys that are used for securing the following communications: <ul style="list-style-type: none"> Between the firewall brokers and the managed server when -auth is ServerSSL or ClientServerSSL. Between the firewall brokers and domain manager when -auth is ServerSSL or ClientServerSSL. <p>Note: This parameter is only required if you specify ServerSSL or ClientServerSSL on the -auth parameter.</p>	
-sslpw	The password used to access the keystore that you specified on the -sslks parameter.	

Related concepts

“Planning for EWLM” on page 79

Use these planning worksheets before you install and configure EWLM.

“Command usage considerations” on page 91

View this topic to learn about the syntax required when issuing an EWLM command. As you complete the EWLM installation and configuration tasks, you may need to run an EWLM command. When you specify an EWLM command, the location from which you run the command and the syntax used to format the command varies by operating system. In addition, the commands that are supported by each operating system vary.

Assigning users to EWLM Control Center roles

View this topic to learn how to assign user IDs that exist on the domain manager’s local user registry to access the EWLM Control Center.

To administer EWLM with the EWLM Control Center, users must have a valid user ID on the server where the domain manager and EWLM Control Center functions run.

Then, you must assign each user ID to an EWLM role based on the function the owner of the user ID will perform when using the EWLM Control Center. The user ID must be granted access to the Control Center by assigning it to an EWLM role. Otherwise, the user cannot log in to the EWLM Control Center. A user can also belong to a group, which can then, in turn, be assigned an EWLM role. All users in a group are assigned the role of the group. The EWLM roles follow:

Administrator

This role has access to all functions. A user assigned to this role can perform setup, manage, and monitor tasks

Operator

This role provides access to the managing and monitoring tasks listed in the EWLM Control Center. A user assigned to this role can manage service policies, view the status of the managed servers in an EWLM domain, and view reports.

Monitor

This role provides access to the monitoring tasks. A user assigned to this role can view reports that contain performance data for the EWLM domain. This role has access to only the monitoring functions of the EWLM Control Center.

To assign user IDs to EWLM Control Center roles, use one of the following methods:

- Run the EWLM domain manager configuration wizard
- Run `changeCC`

If the WebSphere Application Server instance is active when you run the command, then the EWLM Control Center is restarted so that your changes take effect immediately. If the WebSphere Application Server is stopped when you run the command, then the server is left in the stopped state when the command finishes running.

• **Run the EWLM domain manager configuration wizard**

1. Go to the `$(install)/IBM/VE2/EWLM/bin` directory where `$(install)` maps to the platform-specific default directory in which products are typically installed. If you installed the domain manager to a different directory, then run the command from the `EWLM/bin` subdirectory of that directory.

2. Run the following command:

```
configWizardDM
```

For more information about starting the wizard on your operating system, if supported, see, “Configuration wizard usage considerations” on page 106.

3. Select the **Add or remove users to EWLM Control Center roles** task. Complete the wizard as indicated. Click **Help** if you need assistance.

• **Run `changeCC`**

To assign users to EWLM Control Center roles, run the following command, replacing the example parameters shown in italics with your own and adding the appropriate command extension.

```
changeCC -addUser configID -adminUser userid -adminPW password
           -roleUser user -role [Administrator | Monitor | Operator]
```

To assign groups to EWLM Control Center roles, run this command, replacing the example parameters shown in italics with your own and adding the appropriate command extension.

```
changeCC -addGroup configID -adminUser userid -adminPW password
           -roleGroup group -role [Administrator | Monitor | Operator]
```

For more information about issuing the command on your operating system, if supported, see, “Command usage considerations” on page 91.

The following table describes each of the parameters and provides a place for you to enter your specific values if you print this page for reference.

Table 42. Parameters for `changeCC -addUser` and `changeCC -addGroup` scripts

Parameter	Description	Value
-adminPW	The password associated with the -adminUser parameter.	
-adminUser	The WebSphere Application Server administrator user ID. This must be the same user ID that you specified when you ran the <code>createDM</code> command or EWLM domain manager configuration wizard.	
-role	The role you want to assign to the users or groups you specified in the -roleUser or -roleGroup parameter. Possible values are: Administrator, Operator, or Monitor. These values are case-sensitive.	
-roleGroup	The group or groups that you want to access the EWLM Control Center. The same syntactical rules apply to the -roleGroup parameter as for the -roleUser parameter.	
-roleUser	The user or users that you want to access the EWLM Control Center. Specify multiple user IDs by separating each user ID with a colon (:). For example, <code>user1:user2:user3</code> . When a user ID is used to log into the EWLM Control Center, you must specify the same user ID as provided for this parameter. Note that the EWLM Control Center log in field is case sensitive.	

Table 42. Parameters for `changeCC -addUser` and `changeCC -addGroup` scripts (continued)

Parameter	Description	Value
configID	The configuration identifier that EWLM uses to identify the EWLM domain manager. You defined this value when you create the domain manager. You cannot change this value.	

Related concepts

“EWLM Control Center” on page 13

The EWLM Control Center is a Web-based user interface that allows you to manage and monitor the performance of servers in an EWLM management domain. It is installed with and hosted by the domain manager.

Related tasks

“Starting EWLM on the systems in your domain” on page 189

View this topic to learn how to start EWLM on the components of EWLM that you have installed and configured in your EWLM domain. The components of EWLM that you can install and configure include the domain manager, managed servers, and firewall brokers.

“Starting the EWLM Control Center” on page 192

View this topic to learn how to start the EWLM Control Center.

“Removing EWLM Control Center users or groups” on page 281

Use this topic to learn how to remove EWLM Control Center access for a user ID.

Authorizing z/OS users to start EWLM

View this topic to learn how to authorize z/OS users to start EWLM.

After you install and create the domain manager, managed server or firewall broker on z/OS, you must add a user to start it. That user ID, under which the domain manager, managed server or firewall broker process will run, must have the proper authority and access to the necessary directories and files. If you are running the domain manager on z/OS, you must also add a superuser associated with the startup of the WebSphere Application Server.

Complete the following to add a single authorized user to start EWLM on z/OS.

• Domain manager


1. Log on as the user ID defined in the `configWAS` command as the ID for the WebSphere Application Server. This ID must be a superuser (UID(0)).
2. Define a group and user that will be used to run EWLM. The domain manager process will run using the user ID defined here. Use the user ID and group name that you specify in this step throughout rest of these instructions.

To create a group and add a user to the group using RACF, run the following commands:

```
ADDGROUP groupname SUPGROUP(SYS1) OWNER(SYS1) OMVS(GID(group-identifier))
ADDUSER userid DFLTGRP(groupname) NOPASSWORD OMVS(UID(0))
        PROGRAM(/bin/sh) HOME(/usr/lpp/VE_R2/EWLM)
```

Replace the example values shown in italics with your own, as follows:

- *groupname* is the name of the group that contains EWLM user IDs.
- *userid* is the user ID that the domain manager process will run as. IBM recommends using EWLMDM.
- *group-identifier* is the numeric group identifier.

For more information, see z/OS UNIX System Services Planning (<http://publibz.boulder.ibm.com/epubs/pdf/bpxzb260.pdf>). 

Note: You need Adobe Reader installed on your system to view or print this PDF. You can download a free copy from the Adobe Web site

(www.adobe.com/products/acrobat/readstep.html) .

- Grant the group that you created in the previous step access to the EWLM directories. To do this, run the following commands:

```
chgrp -R groupname ewlm_data_root/servers/configID
chgrp groupname /etc/ewlmdm_environment.conf
chgrp -R groupname /var/adm/ewlm
```

Replace the example values shown in italics with your own, as follows:

- *groupname* is the group name that was created in step 2 on page 185.
- *ewlm_data_root* is the value specified for EWLM_DATA_ROOT in the configuration file, ewlmdm_environment.conf.
- *configID* is the EWLM configuration ID for this domain manager. You cannot change this parameter.

- Grant the group access to the configuration file and remove access for all other users. To do this, run the following commands:

```
chmod -R g+rxw /var/adm/ewlm
chmod g+w /etc/ewlmdm_environment.conf
chmod o-rx /etc/ewlmdm_environment.conf
```

- Grant the user the ability to start the EWLM domain manager. To do this with RACF, run the following commands:

```
RDEFINE STARTED EWLM.** STDATA(USER(userid) GROUP(groupname) TRUSTED)
SETROPTS RACLIST(STARTED) REFRESH
```

Replace the example values shown in italics with your own, as follows:

- *userid* is the user ID that was created in step 2 on page 185.
- *groupname* is the group name that was created in step 2 on page 185.

• WebSphere Application Server

- If you haven't already done so, complete the steps for a domain manager.
- Grant the user ID that you created in the steps for a domain manager the ability to start the process. To do this with RACF, run the following commands:

```
RDEFINE STARTED EWLSW.** STDATA(USER(userid) GROUP(groupname) TRUSTED)
SETROPTS RACLIST(STARTED) REFRESH
```

Replace the example values shown in italics with your own, as follows:

- *userid* is the user ID that was created in step 2 on page 185.
- *groupname* is the group name that was created in step 2 on page 185.

• Managed server

- If you haven't already done so, log on to the system on which you want to run the managed server as a superuser.
- If you haven't already done so, define a group and user that will be used to run EWLM. The managed server process will run using the user ID defined here.


To create a group and add a user to the group with RACF, run the following commands:

```
ADDGROUP groupname SUPGROUP(SYS1) OWNER(SYS1) OMVS(GID(group-identifier))
ADDUSER userid DFLTGRP(groupname) NOPASSWORD OMVS(UID(user-identifier))
PROGRAM(/bin/sh) HOME(/usr/lpp/VE_R2/EWLM)
```

Replace the example values shown in italics with your own, as follows:

- *groupname* is the name of the group that contains EWLM user IDs.

- | – *userid* is the user ID that the managed server process will run as. IBM recommends using EWLMS.
- | – *group-identifier* is the numeric group identifier.
- | – *user-identifier* is the numeric user identifier.

| For more information, see z/OS UNIX System Services Planning (<http://publibz.boulder.ibm.com/epubs/pdf/bpxzb250.pdf>). 

- | 3. If you haven't already done so, grant the group that you created in the previous step access to the EWLM directories. To do this, run the following commands:

```
| chgrp -R groupname ewlm_data_root/servers/configID
| chgrp groupname /etc/ewlms_environment.conf
| chgrp -R groupname /var/adm/ewlm
```

| Replace the example values shown in italics with your own, as follows:

- | – *groupname* is the group name that was created in step 2 on page 185.
- | – *ewlm_data_root* is the value specified for EWLM_DATA_ROOT in the configuration file, *ewlms_environment.conf*.
- | – *configID* is the EWLM configuration ID for this managed server. You cannot change this parameter.

- | 4. Grant the group access to the configuration file and remove access for all other users. To do this, run the following commands:

```
| chmod g+w /etc/ewlms_environment.conf
| chmod o-rx /etc/ewlms_environment.conf
```

- | 5. Grant the user ID access to call z/OS platform services. To do this with RACF, run the following commands:

```
| RDEFINE FACILITY MVSADMIN.EWLM.AGENT UACC(NONE)
| PERMIT MVSADMIN.EWLM.AGENT CLASS(FACILITY) ID(userid) ACCESS(READ)
| SETROPTS RACLIST(FACILITY) REFRESH
```

| where *userid* is the user ID that the managed server process will use, which you created in step 2 on page 185

- | 6. Grant the user the ability to start the EWLM managed server. To do this with RACF, run the following commands:

```
| RDEFINE STARTED EWLMS.** STDATA(USER(userid) GROUP(groupname) TRUSTED)
| SETROPTS RACLIST(STARTED) REFRESH
```

| Replace the example values shown in italics with your own, as follows:

- | – *userid* is the user ID that was created in step 2 on page 185.
- | – *groupname* is the group name that was created in step 2 on page 185.

| • **Firewall broker**

- | 1. Log on to the system on which you want to run the firewall broker as a superuser.
- | 2. Define a group and user that will be used to start the firewall broker. The firewall broker process will run using the user ID defined here.


| To create a group and add a user to the group with RACF, run the following commands:

```
| ADDGROUP groupname SUPGROUP(SYS1) OWNER(SYS1) OMVS(GID(group-identifier))
| ADDUSER userid DFLTGRP(groupname) NOPASSWORD OMVS(UID(user-identifier))
| PROGRAM(/bin/sh) HOME(/usr/lpp/VE_R2/EWLM)
```

| Replace the example values shown in italics with your own, as follows:

- | – *groupname* is the name of the group that contains EWLM user IDs.
- | – *userid* is the user ID that the firewall broker process will run as. IBM recommends using EWLMBF.
- | – *group-identifier* is the numeric group identifier.

– *user-identifier* is the numeric user identifier.

For more information, see z/OS UNIX System Services Planning (<http://publibz.boulder.ibm.com/epubs/pdf/bpxzb250.pdf>). 

3. Grant the group that you created in the previous step access to the EWLM directories. To do this, run the following commands:

```
chgrp -R groupname ewlm_data_root/servers/configID
chgrp groupname /etc/ewlmms_environment.conf
chgrp -R groupname /var/adm/ewlm
```

Replace the example values shown in italics with your own, as follows:

- *groupname* is the group name that was created in step 2 on page 185.
- *ewlm_data_root* is the value specified for EWLM_DATA_ROOT in the configuration file, *ewlmms_environment.conf*.
- *configID* is the EWLM configuration ID for this firewall broker. You cannot change this parameter.

4. Grant the group access to the configuration file and remove access for all other users. To do this, run the following commands:

```
chmod g+w /etc/ewlmms_environment.conf
chmod o-rx /etc/ewlmms_environment.conf
```

5. Grant the user ID access to call z/OS platform services. To do this with RACF, run the following commands:

```
RDEFINE FACILITY MVSADMIN.EWLM.AGENT UACC(NONE)
PERMIT MVSADMIN.EWLM.AGENT CLASS(FACILITY) ID(userid) ACCESS(READ)
SETROPTS RACLIST(FACILITY) REFRESH
```

where *userid* is the user ID that the firewall broker process will use, which you created in step 2 on page 185

6. Grant the user the ability to start the EWLM firewall broker. To do this with RACF, run the following commands:

```
RDEFINE STARTED EWLMFB.** STDATA(USER(userid) GROUP(groupname) TRUSTED)
SETROPTS RACLIST(STARTED) REFRESH
```

Replace the example values shown in italics with your own, as follows:

- *userid* is the user ID that was created in step 2 on page 185.
- *groupname* is the group name that was created in step 2 on page 185.

Authorizing z/OS users to ARM services

View this topic to learn how to authorize z/OS users to ARM services. If you have an ARM-instrumented application that you want EWLM to monitor, the user ID under which the application runs must be able to access ARM services.

This step is not required for applications that are instrumented using enclave services, such as DB2 DDF.

For an application that is instrumented using native C or Java services, the user ID the application is running under must have READ access to the System Authorization Facility (SAF) resource BPX.WLMSEVER in the FACILITY class, or the user ID must be defined as a superuser. You can define the user ID as a superuser either with UID(0) or with READ access to the SAF resource BPX.SUPERUSER in the FACILITY class.

Example: To define the BPX.WLMSEVER profile and grant access to it, use the RDEFINE and PERMIT commands, as follows:

```
| RDEFINE FACILITY BPX.WLMSEVER UACC(NONE)
| PERMIT BPX.WLMSEVER CLASS(FACILITY) ID(userid) ACCESS(READ)
```

| **Enabling ARM services on z/OS**

| View this topic to learn how to enable ARM services on z/OS. ARM services must be enabled for the EWLM managed server to function properly.

| The EWLM ARM support in z/OS is automatically enabled.

| If EWLM ARM processing becomes disabled, you can enable ARM services by issuing this command from an operator console:

```
| MODIFY WLM,AM=ENABLE
```

| **Disabling ARM services on z/OS**

| View this topic to learn how to disable ARM services on z/OS.

| To disable EWLM ARM services, run this command from an operator console:

```
| MODIFY WLM,AM=DISABLE
```

| **Verifying the status of ARM services on z/OS**

| View this topic to learn how to verify the status of ARM services on z/OS.

| To verify the status of ARM services on z/OS, run the following command from an operator console:

```
| DISPLAY WLM,AM
```

| The following is an example of a response to the command, showing that ARM services are enabled.

```
| IWM075I 17.54.21 WLM DISPLAY 512
| EWLM ARM SERVICES ARE ENABLED
| EWLM MANAGED SERVER JOBNAME=EWMMS ASID=0032
| EWLM POLICY NAME=BOSCH DAYTIME POLICY
| NUMBER OF REGISTERED PROCESSES=3, APPLICATIONS=1
```

| **Starting EWLM**

| After you install EWLM and configure the components of EWLM that exist in your environment, you are ready to start EWLM.

| You must start EWLM on the system that is the EWLM domain manager. In addition, you must start EWLM on any system that is defined as an EWLM managed server. Similarly, if you have configured a firewall broker, you must also start EWLM on the firewall broker.

| View the following subtopics to learn how to start EWLM on each of the EWLM components that exist in your EWLM domain.

| **Starting EWLM on the systems in your domain**

| View this topic to learn how to start EWLM on the components of EWLM that you have installed and configured in your EWLM domain. The components of EWLM that you can install and configure include the domain manager, managed servers, and firewall brokers.

| Before starting EWLM, ensure that you have configured the component such as the domain manager, managed server, or firewall broker on which you want to start EWLM. As part of the configuration, you define a configuration ID for each component. You will need this configuration ID for each component on which you want to start EWLM.

| The EWLM commands that you use to start EWLM depend on which component you want to start, as follows:

- | • Start EWLM on the domain manager
- | • Start EWLM on the managed server
- | • Start EWLM on the firewall broker

| **Starting EWLM on the domain manager**

- | • **AIX, i5/OS, Linux, and Windows domain managers**

| `startDM configID [-initHeap number] [-maxHeap number] [-autostart [yes | no]]`

| where each parameter is defined as follows:

| **-initHeap** is the initial heap size, in megabytes, allocated to the EWLM domain manager. The default value is 128 MB. This parameter is optional.

| **-maxHeap** is the maximum heap size, in megabytes, allocated to the EWLM domain manager. The default value is 512 MB. This parameter is optional.

| *configID* is the EWLM configuration ID that you specified when you ran `createDM`. You cannot change this parameter.

| **-autostart** indicates whether the domain manager service should start automatically after a restart. If this parameter is not specified, EWLM uses the current value for the Windows service to determine if the service is started automatically after a restart. This parameter is available only on Windows.

- | • **z/OS domain managers**

| You can use either a z/OS Unix System Services command or an MVS command. Log on with the user ID you created for this process. See “Authorizing z/OS users to start EWLM” on page 185.

- | – z/OS Unix System Services command:

| `startDM configID [-initHeap number] [-maxHeap number]`

| where each parameter is defined as follows:

| **-initHeap** is the initial heap size, in megabytes, allocated to the EWLM domain manager. The default value is 128 MB. This parameter is optional.

| **-maxHeap** is the maximum heap size, in megabytes, allocated to the EWLM domain manager. The default value is 512 MB. This parameter is optional.

| *configID* is the EWLM configuration ID that you specified when you ran `createDM`. You cannot change this parameter.

| The `startDM` command runs in the foreground.

- | – MVS command:

| `START EWLMDM,CONFIGID='configID'`

| where *configID* is the configuration ID that is created when you ran `createDM`.

| Note that if the configuration ID includes lowercase characters, you will need to enter the `START` command in a way that preserves the case of parameters (for example, entering the command from an MVS console and enclosing the *configID* in single quotation marks, as shown above).

| The `EWLMDM` procedure runs in the background using the user ID you created for it.

| **Starting EWLM on the managed server**

- | • **AIX, HP-UX, i5/OS, Linux, Solaris, and Windows managed servers**

| `startMS configID [-autostart [yes | no]]`

| where each parameter is defined as follows:

| *configID* is the EWLM configuration ID that you specified when you ran `createMS`. You cannot change this parameter.

| **-autostart** indicates whether the domain manager service should start automatically after a restart. If this parameter is not specified, EWLM uses the current value for the Windows service to determine if the service is started automatically after a restart. This parameter is available only on Windows.

You can use either a z/OS Unix System Services command or an MVS command. Log on with the user ID you created for this process. See “Authorizing z/OS users to start EWLM” on page 185

- z/OS Unix System Services command:

```
startMS configID
```

where the parameter is defined as follows:

configID is the EWLM configuration ID that you specified when you ran createMS. You cannot change this parameter.

The startMS command runs in the foreground.

- MVS command:

```
START EWLMS,CONFIGID='configid'
```

where *configID* is the configuration ID that is created with createMS.

If the configuration ID includes lowercase characters, you need to enter the START command in a way that preserves the case of parameters. For example, entering the command from an MVS console and enclosing the working directory name in single quotation marks, as shown above.

The EWLMS procedure runs in the background using the user ID you created for it.

- **z/OS managed server**

You can use either a z/OS Unix System Services command or an MVS command. Log on with the user ID you created for this process. See “Authorizing z/OS users to start EWLM” on page 185

- z/OS Unix System Services command:

```
startMS configID
```

where the parameter is defined as follows:

configID is the EWLM configuration ID that you specified when you ran createMS. You cannot change this parameter.

The startMS command runs in the foreground.

- MVS command:

```
START EWLMS,CONFIGID='configid'
```

where *configID* is the configuration ID that is created with createMS.

If the configuration ID includes lowercase characters, you need to enter the START command in a way that preserves the case of parameters. For example, entering the command from an MVS console and enclosing the working directory name in single quotation marks, as shown above.

The EWLMS procedure runs in the background using the user ID you created for it.

Note: You must start each managed server in the domain separately.

Starting EWLM on the firewall broker

If you use a firewall broker to allow the managed servers to access the domain manager through a SOCKS server, you must run the following command found in the managed server installation path.

- **AIX, HP-UX, i5/OS, Linux, Solaris, and Windows firewall brokers**

```
startFB configID [-autostart [yes | no]]
```

where each parameter is defined as follows:

configID is the EWLM configuration ID that is created when you ran createFB. This parameter is required and you cannot change it with this command.

-autostart indicates whether the firewall broker service should start automatically after a restart. If this parameter is not specified, EWLM uses the current value for the Windows service to determine if the service is started automatically after a restart. This parameter is available only on Windows.

- **z/OS firewall brokers**

You can use either a z/OS Unix System Services command or an MVS command. Log on with the user ID you created for this process. See “Authorizing z/OS users to start EWLM” on page 185

- z/OS Unix System Services command:

```
startFB configID
```

where the parameter is defined as follows:

configID is the EWLM configuration ID that is created when you ran createFB. This parameter is required and you cannot change it with this command.

The startFB command runs in the foreground.

- MVS command:

```
START EWLMFB,CONFIGID='configID'
```

where *configID* is the configuration ID that is created when you ran createFB.

If the configuration ID includes lowercase characters, you need to enter the START command in a way that preserves the case of parameters. For example, entering the command from an MVS console and enclosing the working directory name in single quotation marks, as shown above.

The EWLMFB procedure runs in the background using the user ID you created for it.

Related concepts

“EWLM domain manager” on page 12

The *domain manager* is the central point of control for an EWLM management domain.

“EWLM managed server” on page 12

A *managed server* is a server or server instance that EWLM monitors.

“Firewall broker” on page 15

A *firewall broker* provides the ability to allow communication through a firewall.

Related reference

“Start EWLM (STRWLM) i5/OS CL command” on page 193

View this topic to learn about the i5/OS STRWLM CL command. You can use this command as an alternate to the EWLM commands to start EWLM on an i5/OS system.

Starting the EWLM Control Center

View this topic to learn how to start the EWLM Control Center.

Ensure the user account under which the Web Sphere Application Server instance will run, has the proper authorities.

To start the EWLM Control Center, complete the following:

1. Start the WebSphere Application Server instance, which also starts the EWLM Control Center. To do this, run the following command on the server that you have configured as the domain manager:

```
startWAS configID [-autostart [YES | NO]
-adminUser userID
-adminPW password]
```

where each of the possible parameters is described as follows:

- **configID** is the configuration identifier that uniquely identifies the domain manager or single-system EWLM.
- **-autostart** indicates whether the Control Center service should start automatically after a restart of the system. The default value is yes. A value of yes indicates that the service is started automatically when the server is restarted. If this parameter is not specified, the default value is used. If no is specified, you must manually start the Control Center service when Windows is restarted. If this parameter is not specified, the current autostart value associated with this Windows service is used. This is an optional parameter that is only supported on Windows.
- **-adminUser** is the user ID that administers the WebSphere Application Server. You must specify this parameter if you specified **-autostart**. This is an optional parameter that is only supported on Windows.

- **-adminPW** is the password for the **-adminUser**. You must specify this parameter if you specified **-autostart**. This is an optional parameter that is only supported on Windows.

For more information about issuing the command on your operating system, if supported, see, “Command usage considerations” on page 91.

2. Access the EWLM Control Center from a supported Web browser using the following URL:

https://domain_manager_ipaddress:control_center_port/webui

where each of the values shown in italics are described as follows:

- *domain_manager_ipaddress* is the host name or IP address of the domain manager. This value maps to the value you specified for the **-ma** on the **createDM** command.
- *control_center_port* is EWLM Control Center port. This is the value that you specified when you ran the **createDM** command.

Note: Ensure that the Web browser can access the Java Runtime Environment (JRE) version that EWLM requires. For more information about the supported JRE, see Web browser requirements

3. Log in to the EWLM Control Center as one of the users you assigned to an EWLM role.

Related concepts

“EWLM Control Center” on page 13

The EWLM Control Center is a Web-based user interface that allows you to manage and monitor the performance of servers in an EWLM management domain. It is installed with and hosted by the domain manager.

Start EWLM (STRWLM) i5/OS CL command

View this topic to learn about the i5/OS STRWLM CL command. You can use this command as an alternate to the EWLM commands to start EWLM on an i5/OS system.

Use this CL command to start EWLM on the domain manager, managed server or firewall broker.

If you do not want to use the i5/OS CL command to start EWLM, you can use the EWLM commands. For more information on how to start EWLM using the EWLM commands, see “Starting EWLM on the systems in your domain” on page 189.

After the EWLM installation is complete, the i5/OS CL commands for EWLM reside in the QSYS library, if 2924 is the primary language. If 2924 is the secondary language, the CL commands reside in the QWLM22924 library.

This command also starts a new EWLM server job under the QWLM2 subsystem, running under the QWLM2 profile, with job name, QWLM2SVR. You must have *JOBCTL special authority to use the STRWLM CL command.

Syntax

STRWLM *configID*

Parameters

Parameter	Description
configID	The configuration ID that you specified when you created the domain manager, managed server, or firewall broker. Specify the configuration ID for the component of EWLM that you want to start. You created the configuration ID when you created the domain manager, managed server, or firewall broker.

Related tasks

“Add EWLM library to your system library list” on page 127

Learn how to add the EWLM library to your system library list. You must complete this step if English is not the primary language installed on the system.

“Configuring a domain manager” on page 108

View this topic to learn how to configure a domain manager.

“Configuring a managed server” on page 111

View this topic to learn how to configure a managed server.

“Creating a firewall broker” on page 114

View this topic to learn how to create a firewall broker. You need to create a firewall broker to allow EWLM communication between the domain manager and managed server, if the managed server accesses the domain manager by using a SOCKS server.

“Starting EWLM on the systems in your domain” on page 189

View this topic to learn how to start EWLM on the components of EWLM that you have installed and configured in your EWLM domain. The components of EWLM that you can install and configure include the domain manager, managed servers, and firewall brokers.

Stopping EWLM

View this topic to learn how to stop the EWLM domain manager, managed server, and firewall broker. This task describes how to stop the EWLM function that runs on the system that contains the EWLM domain manager, managed server, or firewall broker. This task does not shutdown or stop the system.

The command you use to stop EWLM depends on which component you want to stop, as follows:

- Stop EWLM on the domain manager
- Stop EWLM on the managed server
- Stop the EWLM firewall broker

For more information on issuing the command described in the following subtopics on your operating system, if supported, see, Command usage considerations.

Stopping EWLM on the domain manager

To stop EWLM on the domain manager, run the following command, replacing the example values shown in italics with your own.

```
stopDM configID
```

where the parameters are defined as follows:

configID is the EWLM configuration ID that you specified when you ran the `createDM` command. You cannot change this parameter.

Stopping EWLM on the managed server

To stop EWLM on the managed server, you can use one of the following methods:

- **EWLM Control Center**
 1. Log on to the EWLM Control Center.
 2. Select **Set up** → **Manage** → **Managed servers**.
 3. Select the managed server on which you want to stop EWLM so that it is highlighted.
 4. Select **Disable** in the Select Action field.

5. Click **Go**.

- **EWLM command**

Run the following command, replacing the example values shown in italics with your own.

```
stopMS configID
```

where the parameters are defined as follows:

configID is the EWLM configuration ID that you specified when you ran the createMS command.
You cannot change this parameter.

Stopping the firewall broker

To stop the EWLM firewall broker, run the following command, replacing the example values shown in italics with your own.

```
stopFB configID
```

where the parameters are defined as follows:

configID is the EWLM configuration ID that you specified when you ran the createFB command. You cannot change this parameter.

Related tasks

“Uninstalling EWLM” on page 320

View this topic to learn how to uninstall EWLM.

End EWLM (ENDWLM) i5/OS CL command

View this topic to learn about the i5/OS ENDWLM CL command. You can use this command as an alternate to the EWLM commands to stop EWLM on an i5/OS system.

Use this CL command to stop EWLM on the domain manager, managed server, or firewall broker.

If you do not want to use the i5/OS CL command to stop EWLM, you can use the EWLM commands. For more information on how to stop EWLM using the EWLM commands, see “Stopping EWLM” on page 194.

After the EWLM installation is complete, the i5/OS CL commands for EWLM reside in the QSYS library, if 2924 is the primary language. If 2924 is the secondary language, the CL commands reside in the QWLM2924 library.

You must have *JOBCTL special authority to use the ENDWLM CL command.

Syntax

```
ENDWLM configID
```

Parameters

Parameter	Description
<i>configID</i>	The configuration ID that you specified when you created the domain manager, managed server, or firewall broker. Specify the configuration ID for the component of EWLM that you want to stop. You created the configuration ID when you created the domain manager, managed server, or firewall broker.

| “Configuring a domain manager” on page 108

| View this topic to learn how to configure a domain manager.

| “Configuring a managed server” on page 111

| View this topic to learn how to configure a managed server.

| “Creating a firewall broker” on page 114

| View this topic to learn how to create a firewall broker. You need to create a firewall broker to allow EWLM communication between the domain manager and managed server, if the managed server accesses the domain manager by using a SOCKS server.

| “Stopping EWLM” on page 194

| View this topic to learn how to stop the EWLM domain manager, managed server, and firewall broker. This task describes how to stop the EWLM function that runs on the system that contains the EWLM domain manager, managed server, or firewall broker. This task does not shutdown or stop the system.

| **Stopping EWLM on z/OS**

| View this topic to learn how to stop the EWLM domain manager, managed server, and firewall broker on z/OS.

| You use an MVS CANCEL command to stop EWLM on z/OS. The procedure name varies with the component you want to stop, as follows:

- | • Stopping EWLM on the domain manager
- | • Stopping EWLM on the managed server
- | • Stop the EWLM firewall broker

| **Stopping EWLM on the domain manager on z/OS**

| To stop EWLM on the domain manager, run the MVS CANCEL command as follows:

| CANCEL EWLMMDM

| **Stopping EWLM on the managed server on z/OS**

| To stop EWLM the managed server, run the MVS CANCEL command as follows:

| CANCEL EWLMMS

| **Stopping the EWLM firewall broker on z/OS**

| To stop the firewall broker, run the MVS CANCEL command as follows:

| CANCEL EWLMFB

| **Stopping the EWLM Control Center**

| Use this topic to learn how to stop the EWLM Control Center. Even if you close the Web session for the EWLM Control Center, the EWLM Control processes are still running. To stop it completely, you must complete the task outlined in this topic.

| Stopping the EWLM Control Center requires that you stop the WebSphere Application Server instance, which also stops the EWLM Control Center.

| Select one of the following methods to stop the EWLM Control Center, depending on which method that you used to start it, unless noted otherwise.

- | • Stop the EWLM Control Center using the stopWAS command

- Stop the EWLM Control Center Windows service

Stop the EWLM Control Center using the stopWAS command

To stop the EWLM Control Center, run this command, replacing the example parameters shown in italics with your own and adding the appropriate command extension.

```
stopWAS configID -adminUser userid -adminPW userpw
```

The following table describes each of the parameters and provides a place for you to enter your specific values if you print this page for reference.

Table 43. Parameters for the *changeCC -removeGroup* and *changeCC -removeUser* commands.

Parameter	Description	Your value
configID	The configuration identifier that was created when you ran the createDM configuration command.	
userid	The WebSphere Application Server administrator user ID.	
userpw	The password associated with userid .	

Stop the EWLM Control Center Windows service

Windows users can stop the EWLM Control Center by using either the stopWAS command method or by stopping the Windows service. The method that you use does not depend on which method you used to start the Control Center. To stop the EWLM Control Center Windows service, complete the following steps:

1. From the **Control Panel**, open **Administrative Tools** → **Services**.
2. From the right pane of the **Services** window, right-click **IBM WebSphere Application Server V6 - EWLM Control Center - *configID***, where *configID* is the configID associated with this domain manager.
3. Select **Stop**.

Related tasks

“Managing service policies” on page 272

View this topic to learn about the tasks you can perform with service policy in the Control Center. Also, you will learn how to use service policies within your business environment.

“Managing managed servers” on page 273

View this topic to learn about the tasks that you can perform with your managed servers. You can also learn how to use the Control Center to monitor the performance of your managed servers.

“Managing applications in EWLM” on page 274

View this topic to learn how to manage applications in your EWLM domain. EWLM can monitor application-level transactions for applications that use the Application Response Measurement (ARM) 4.0 standard APIs. The EWLM Control Center contains a predefined list of applications that use the ARM 4.0 standard.

“Managing platforms in EWLM” on page 278

View this topic to learn how to manage a platform in EWLM.

“Managing partitions in EWLM” on page 279

View this task to learn how you can manage a partition in EWLM. You will also learn how you can use the EWLM Control Center to manage the partitions that EWLM is to monitor.

“Enabling EWLM Control Center tracing” on page 280

View this topic to learn how to define the EWLM Control Center trace level.

“Removing EWLM Control Center users or groups” on page 281

Use this topic to learn how to remove EWLM Control Center access for a user ID.

Managing EWLM

View this topic collection to work with additional tasks that you can perform when using EWLM. In addition, this topic collection describes how to enable optional functions of EWLM such as partition management and load balancing capabilities.

Securing EWLM

EWLM provides a default level of security. However, you can take additional steps to enhance the security level for your business environment.

Related concepts

“Firewall broker” on page 15

A *firewall broker* provides the ability to allow communication through a firewall.

Accessing the domain manager through a firewall

Use this topic to choose the proper firewall method for your environment.

A typical e-business environment includes the use of firewalls to control the access and flow of information between secure or trusted systems and unsecure or untrusted systems across a network. EWLM supports connecting managed servers to a domain manager through two industry standard firewall methods. They are:

HTTP proxy server

If your managed servers must connect to the domain manager through an HTTP proxy server, you must configure the managed server so that it can identify the HTTP proxy server. The domain manager requires no additional configuration. In this scenario, the domain manager receives data from the managed server as though the managed server were connecting directly to it. See “Example: Accessing the domain manager through an HTTP proxy server” for more details about the configuration parameters required when you connect your managed servers to the domain manager through an HTTP proxy server. If the HTTP proxy server is used to limit connections from the domain manager to the managed server, no additional configuration is required on either the managed servers or the domain manager.

SOCKS server

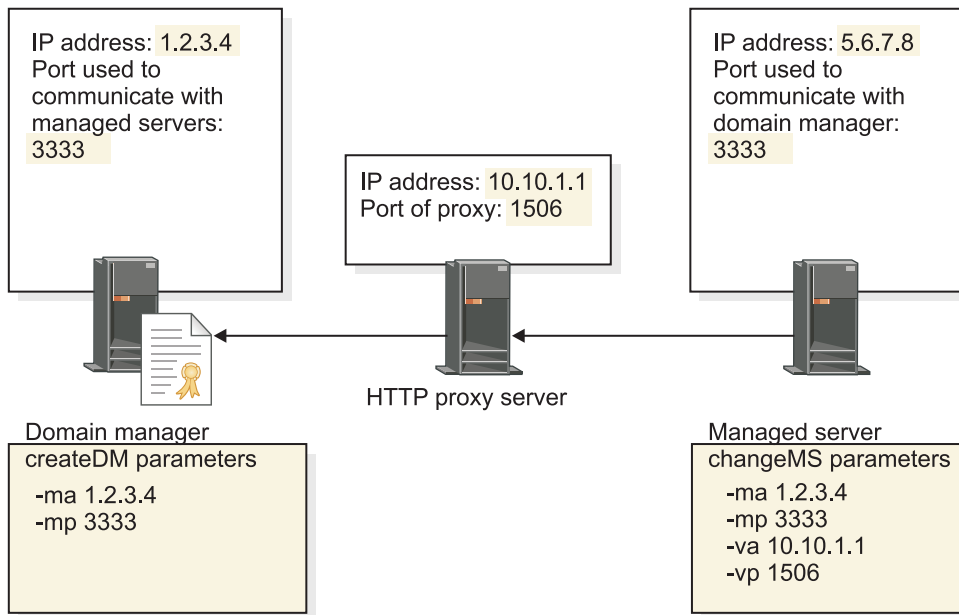
If a SOCKS server limits connections from the managed servers to the domain manager, you must configure a firewall broker. If the SOCKS server only limits connections from the domain manager to the managed server, then no firewall broker is required. The firewall broker does not have to run on a server that is acting as a managed server. However, it does have to be in the same trusted zone as the managed servers.

There are several configuration parameters that must match when you create and configure the domain manager, the managed servers, and the firewall broker. If any of the required parameters are incorrect, the entire communication stream will fail. “Example: Accessing the domain manager through a SOCKS server” on page 199 illustrates the configuration requirements when your managed servers access the domain manager through a SOCKS server. In addition, “Creating a firewall broker” on page 114 describes how to create a firewall broker.

Example: Accessing the domain manager through an HTTP proxy server:

If your managed servers must connect to the domain manager through an HTTP proxy server, you must configure the managed server so that it can identify the HTTP proxy server.

The following figure illustrates the configuration required when you connect your managed servers to the domain manager through an HTTP proxy server.



EICAA518-1

In this scenario, after you run createMS to create the managed server instance, you must use the changeMS command to set the **-va** and **-vp** parameters. These parameters point the managed server to the IP address and port of the HTTP proxy server, respectively. changeMS and all of its parameters are described in “Changing managed server configuration properties” on page 257.

If you use the HTTP proxy server to connect to the managed servers from the domain manager, no additional configuration is required on either the managed servers or the domain manager. In other words, you do not need to run changeMS to set the **-va** and **-vp** parameters.

Example: Accessing the domain manager through a SOCKS server:

If a SOCKS server limits connections from the managed servers to the domain manager, you must configure a firewall broker. When you installed the managed server images by using the Virtualization Engine, a firewall broker image was also installed to the following directory:

```
$D(install)/IBM/VE2/EWLMMS/bin
```

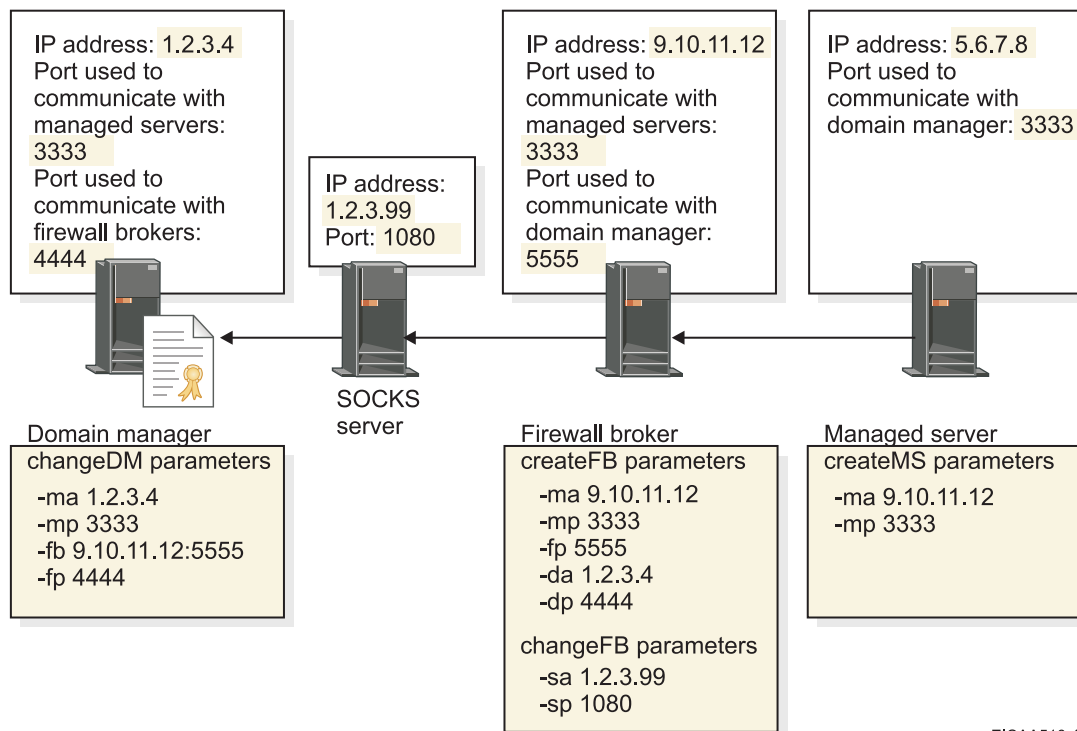
where `$D(install)` maps to the operating system-specific default directory in which products are typically installed. If you installed the firewall broker to a different directory, then run the command from the `EWLMMS/bin` subdirectory of that directory.

See “Command usage considerations” on page 91 for information on how to format this command for your operating system.

Just as with the managed server images, you must also distribute the firewall broker image to the appropriate server platform and create and configure it from that server. The firewall broker does not have to run on a server that is acting as a managed server. However, it does have to be in the same trusted zone as the managed servers. There are several configuration parameters that need to match when you create and configure the domain manager, the managed servers, and the firewall broker. If any of the required parameters are incorrect, the entire communication stream will fail. The following figures illustrate the configuration requirements when your managed servers access the domain manager through a SOCKS server.

The first figure shows a managed server connecting to a firewall broker, which in turn connects to the domain manager through the SOCKS server. In this scenario, the SOCKS server is protecting the zone

where the domain manager is located.



Managed server set up

The managed server requires no additional parameters, however its **-ma** and **-mp** parameters must point to the IP address (or host name) and port of the firewall broker.

Firewall broker set up

The firewall broker requires that you set the following parameters on the createFB command:

- **-ma** which is the IP address or hostname of the firewall broker. The **-ma** parameter you will specify on the createMS command must match this value.
- **-fp** which is the port used exclusively for domain manager to firewall broker communications. This parameter maps to the port value passed on the **-fb** parameter of the changeDM command.
- **-da** which is the IP address of the domain manager. This parameter maps to the **-ma** parameter on the changeDM command.
- **-dp** which is the port used by the firewall broker to connect to the domain manager. This parameter maps to the **-fp** parameter on the changeDM command.

In this example, after you run createFB, you must also run changeFB to set the following properties:

- **-sa** which is the IP address of the SOCKS server. On which server you specify this parameter depends on how the SOCKS server is setup. If the SOCKS server is protecting the zone where the domain manager is located, then you set the **-sa** parameter on the firewall broker, as in this example. In a scenario where there are SOCKS servers protecting both zones, as in a DMZ, you would set the **-sa** parameter on both the firewall broker and the domain manager to point to their respective SOCKS server. An example of this setup is described in the next figure.
- **-sp** which is the port of the SOCKS server. As with the **-sa** parameter, where you set **-sp** depends on how the firewall protection is set up.

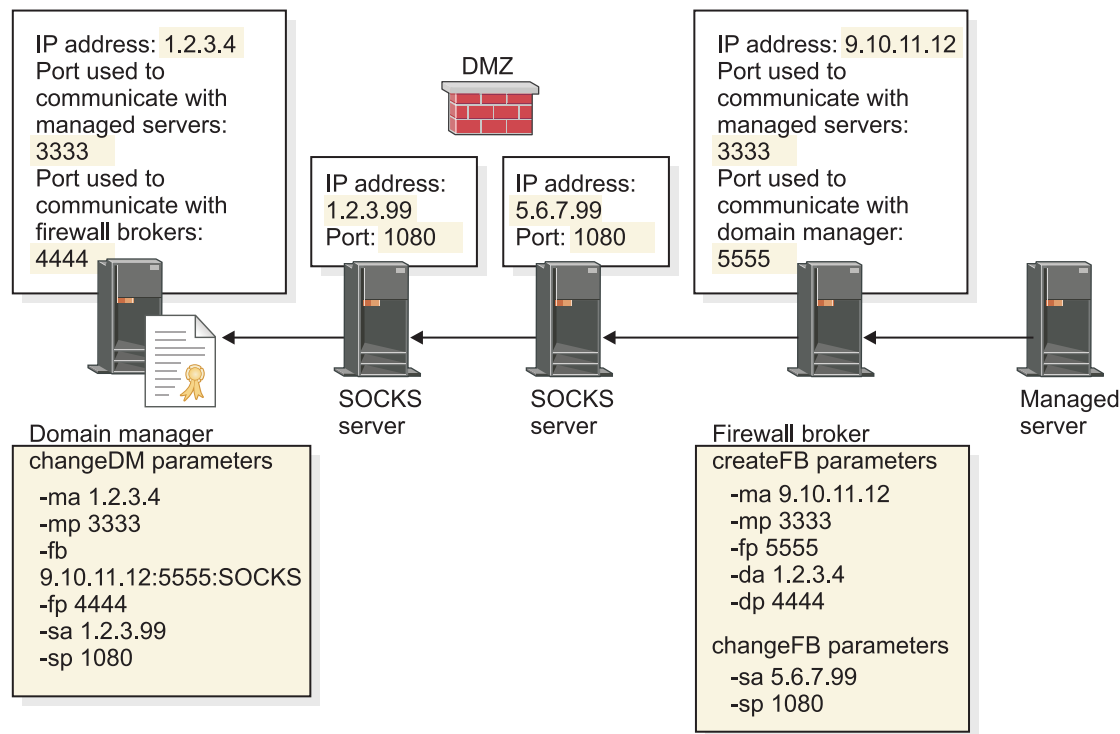
Domain manager set up

The domain manager requires that you set the following parameters on the changeDM command:

- **-fb** which identifies a list of firewall brokers with which the domain manager will communicate. Each firewall broker is identified by:
 - IP address, which maps to the **-ma** parameter on the createFB command.
 - Port, which maps to the **-fp** parameter on the createFB command.
- **-fp** which is the port the firewall broker uses to communicate with the domain manager. This parameter maps to the **-dp** parameter on the createFB command.

The next figure illustrates the configuration requirements when a SOCKS server protects the zone where the firewall broker and managed servers are located and another SOCKS server protects the zone where the domain manager is located.

! **Note:** A DMZ is a configuration of firewalls in which connections in both directions are restricted.



EICAA517-1

The configuration is essentially the same as described in the first figure. The only addition is that you must also set the **-sa** and **-sp** parameters to identify the IP address and port of the SOCKS server that is protecting the zone where the domain manager is located and add the "SOCKS" tag to the **-fb** parameter to indicate that a SOCKS server is required to connect to the firewall broker. You specify each of these parameters on the changeDM command

Related information

"Creating a firewall broker" on page 114 describes the createFB command and all of its parameters.

"Changing domain manager configuration properties" on page 253 describes the changeDM command and all of its parameters.

"Configuring a managed server" on page 111 describes the createMS command and all of its parameters.

“Changing managed server configuration properties” on page 257 describes the changeMS command and all of its parameters.

Specifying a secure SSL certificate for Web browser and EWLM Control Center communications

Use this topic to learn the two methods for obtaining and deploying your own trusted SSL keys in a keystore to ensure the highest level of security in your production environment.

The WebSphere Application Server instance, under which the EWLM Control Center runs, provides a default level of SSL security for protecting communications between a Web browser and the EWLM Control Center. However, you should not use the default SSL security in a production environment. To ensure the highest level of security, you should obtain and deploy your own trusted SSL keys in a keystore. There are two methods for doing this, depending on whether or not you have the EWLM domain manager configuration wizard available on your system:

- If the EWLM domain manager configuration wizard is available on your system, specify a keystore when you create a domain manager.
- If the EWLM domain manager configuration wizard is **not** available on your system, specify a keystore using the changeCC command with the -sslSecurity option.
- Specifying '-level adminDefined' on the changeCC -sslSecurity configuration command enables you to specify the path to the keystore for the SSL keys and the password for accessing it.

For more information on issuing the command on your operating system, if supported, see “Command usage considerations” on page 91.

To ensure secure communications between a Web browser and the EWLM Control Center, run this command from the server on which the domain manager is running:

```
changeCC -sslSecurity configID
          -adminUser userid
          -adminPW password
          -level [adminDefined | ewlmDefined]
          -keystore keystore
          -keystorePW keystorepassword
```

where each parameter is defined as follows.

Table 44. Parameter definitions


Parameter	Description	Your value
-configID	The configuration identifier used to identify the EWLM domain manager server job.	
-adminPW	The password associated with the administrator user ID.	
-adminUser	The user ID of the administrator of this WebSphere Application Server instance.	
-keystore	The path to the certificate keystore. Note: You must specify "-level adminDefined" in order to use this parameter.	
-keystorePW	The password used to access the keystore you specified on the -keystore parameter. Note: You must specify "-level adminDefined" in order to use this parameter.	

Table 44. Parameter definitions (continued)

Parameter	Description	Your value
-level	<p>The source of the SSL certificate that is presented to a Web browser when connecting to the EWLM Control Center. Possible values are:</p> <p>adminDefined You have obtained your own SSL, so the keystore is provided on the -keystore and -keystorePW parameters. Note: You must set the -level parameter to "adminDefined" if you wish to use the -keystore and -keystorePW parameters.</p> <p>ewlmDefined You want to revert to the default keys. (This is not recommended.) Note: If you are not ready to obtain keys, there is no need to run this command.</p>	

Setting up secure communications

Use this topic to learn how to create and use your own SSL certificates to ensure a greater level of security across your EWLM domain.

EWLM provides a default level of security for protecting the various communication streams in an EWLM domain. However, as a general rule, you should not use the EWLM default security in a production environment. To provide a greater level of security across the EWLM domain, you should create and use your own SSL certificates. With EWLM, you can use either server SSL or client/server SSL to secure communications. Currently, EWLM supports Sun JKS keystores. You can create keystores using a number of methods, one of which is to use the **keytool** utility available with the Java 2 Platform, Enterprise Edition (J2EE) SDK. The high-level process for creating SSL certificates follows. Refer to the J2EE security  documentation for detailed setup instructions.

Using server SSL between the domain manager and managed servers:

The following procedure shows you how to set up server SSL between a domain manager and the managed servers. The domain manager and the managed server will each have their own keystore. To use server SSL in your EWLM environment on AIX, Linux, Windows and z/OS, you must set the following parameters on createDM or changeDM, and createMS or changeMS. To use server SSL in your EWLM environment on HP-UX, i5/OS and Solaris, you must set the following parameters on createMS or changeMS.

- **-auth** ServerSSL
- **-sslks**
- **-sslpw**

Note: Server SSL between a firewall broker and managed servers uses the same set up procedures, simply replace references to the domain manager with firewall broker.

1. Create a keystore for the domain manager in a directory on the system acting as the domain manager. This is the directory you will specify on the **-sslks** parameter for createDM or changeDM.
2. Export the public key into a certificate.
3. Import the public key back into the keystore of the domain manager. This enables the domain manager to perform internal communications.
4. Send the certificate to the managed server as binary data using a secure transport mechanism.
5. Create a keystore for the managed server in a directory on the system acting as a managed server. This is the directory you will specify on the **-sslks** parameter for createMS or changeMS.
6. Import the certificate into the keystore on the managed server.

7. Specify that you want to store the public key (in the certificate) in the keystore on the managed server.
8. When prompted, indicate that, yes, this is a trusted key, and then specify the password for the keystore.

Using client/server SSL between the domain manager and managed servers:

Client/server SSL provides an additional level of security by requiring mutual authentication of the servers.

- | To use client/server SSL in your EWLM environment on AIX, Linux, Windows and z/OS, you must set the following parameters on createDM or changeDM, and createMS or changeMS. To use server SSL in your EWLM environment on HP-UX, i5/OS and Solaris, you must set the following parameters on createMS or changeMS.

- **-auth** ClientServerSSL
- **-sslks**
- | • **-sslpw**

Follow the same procedures for setting up client/server SSL as those described for setting up server SSL. After you have set up server SSL on the domain manager and managed servers, you must complete the following additional steps on each managed server:

1. Create a public/private keypair for the managed server and place it in the keystore you created earlier.
2. Export the public key into a certificate.
3. Send the certificate to the domain manager as binary data using a secure transport mechanism.
4. Import the certificate into the keystore of the domain manager that you created earlier.
5. Specify that you want to store the public key (in the certificate) in the keystore on the domain manager.
6. When prompted, indicate that, yes, this is a trusted key, and then specify the password for the keystore.

Using client/server SSL between the domain manager and firewall brokers:

If you plan to use your own SSL certificates to protect communications between the domain manager and firewall brokers, you must use client/server SSL. The following procedure provides an example of how to set up client/server SSL between a domain manager and the firewall brokers:

1. Create a keystore for the domain manager in a directory on the system acting as the domain manager. This is the directory you will specify on the **-sslks** parameter for createDM or changeDM.
2. Export the public key into a certificate.
3. Import the public key back into the keystore of the domain manager. This step is necessary because it enables the domain manager to perform internal communications.
4. Send the certificate to the firewall broker as binary data using a secure transport mechanism.
5. Create a keystore for the firewall broker in a directory on the system acting as a firewall broker. This is the directory you will specify on the **-sslks** parameter for createFB or changeFB.
6. Import the certificate into the keystore on the firewall broker.
7. Specify that you want to store the public key (in the certificate) in the keystore on the firewall broker.
8. When prompted, indicate that, yes, this is a trusted key, and then specify the password for the keystore.
9. Create a public/private key pair for the firewall broker and place it in the keystore you created earlier.

10. Export the public key into a certificate.
11. Send the certificate to the domain manager as binary data using a secure transport mechanism.
12. Import the certificate into the keystore of the domain manager.
13. Specify that you want to store the public key (in the certificate) in the keystore on the domain manager.
14. When prompted, indicate that, yes, this is a trusted key, and then specify the password for the keystore.

Enabling ARM on IBM-instrumented applications

Use this topic to learn how to enable ARM on supported middleware applications.

Applications for which Application Response Measurement (ARM) instrumentation is necessary are those acting as application servers, that is, applications that service requests from multiple clients, typically in parallel. Middleware applications, such as DB2 Universal Database and WebSphere Application Server, are normally deployed in multi-tier customer solutions and have the types of problems that EWLM monitors. By installing applications and middleware that are instrumented using ARM 4.0 on your managed servers, EWLM can assemble an end-to-end view of transaction flows for the entire EWLM domain.



IBM currently provides ARM instrumentation support on the following middleware applications:

- DB2 Universal Database, Version 8.2 or later
- WebSphere Application Server, Version 5.1.1 or later
- IBM Web serving plug-ins

Note: The WebSphere Application Server includes the following predefined applications that provide ARM instrumentation support by default because they use the ARM instrumentation framework provided by WebSphere Application Server. They include, but are not limited to, the following applications:

- Apache HTTP Server
- IBM HTTP Server
- Microsoft Internet Information Services (IIS)

For more information about Web serving plug-ins and required server levels, refer to the appropriate WebSphere Application Server information center:

- WebSphere Application Server 5.1 
- WebSphere Application Server 6.0 

In addition, there exist the following non-WebSphere Application Server Web serving plug-ins which use the same filter types as the WebSphere Application Server Web serving plug-ins listed above:

- Independent plug-in for Apache HTTP Server 2.0.47 or later.
- Independent plug-in for IBM HTTP Server 2.0.47 or later.
- IBM HTTP Server for iSeries
- EWLM-supplied plug-in for Internet Information Services (IIS) 5.0 and 6.0

Use this plug-in for static Web page serving only. Deployment of this IIS plug-in for other than static Web page serving can result in inaccurate ARM transaction event reporting, and subsequently inaccurate EWLM interpretation of transaction data.

Note: Only one plug-in is allowed on a Web server at any time.

| **Note:** Linux: The EWLM managed server for Linux on SUSE Linux Enterprise Server (SLES) 9 does not
| support applications, middleware, daemons or other deployed program solutions that run in a
| LinuxThreads environment. Furthermore, the EWLM managed server for Linux process monitoring
| and performance metrics collection is unreliable for programs that run in this LinuxThreads
| environment and can lead to unexpected results in EWLM reporting and management. The EWLM
| managed server for Linux on SLES 9 does fully support applications, middleware, daemons or
| other deployed program solutions that execute in a Native POSIX Threading Library (NPTL)
| environment.

While any application may instrument the ARM 4.0 standard on the supported operating systems, IBM has not tested every ARM instrumentation with EWLM. IBM does not provide support or documentation for all the environments and makes no warranties of any kind that EWLM will or can be made to work on all applications and operating systems.

Use the following subtopics as a reference when enabling ARM for the applications specified. Note that the applications and steps are subject to change. Please refer to documentation for your application as a master source of reference.

| **Note:** In the following topics, variables are used to represent the default file system install locations to
| which you installed the middleware applications:
| • *IHS_HOME* and *IHS2_HOME* represent the default install paths for IBM HTTP Server.
| • *WAS_HOME* represents the default install path for WebSphere Application Server.

Related concepts

“Application instrumentation” on page 20

Instrumenting an application with the appropriate APIs allows EWLM to properly monitor the application-level transactions.

Related references

EWLM APIs

Enabling ARM on DB2 Universal Database for use by WebSphere Application Server

Use this topic to enable ARM on DB2 Universal Database for use by WebSphere Application Server.

The specific tasks you must complete to enable ARM vary depending on the target operating system. Select from the list below a topic collection that is specific to the operating system on which you want to enable ARM on DB2 Universal Database for use by WebSphere Application Server.

Related information

DB2 UDB preparing to work with IBM Enterprise Workload Manager (EWLM)

(http://www.ibm.com/support/docview.wss?rs=71&context=SSEPGG&dc=DB520&dc=D600&dc=DB530&dc=D700&dc=DB500&dc=DB540&dc=DB510&dc=DB550&q1=EWLM&uid=swg21213956&loc=en_US&cs=utf-8&lang=en)

Enabling ARM on DB2 Universal Database for use by WebSphere Application Server on AIX:



Follow the instructions in this topic to enable ARM on DB2 Universal Database for use by WebSphere Application Server on AIX.

ARM is enabled on DB2 Universal Database version 8 by default. However, there are several additional steps you must complete to deploy DB2 for use by WebSphere Application Server in an EWLM environment on your managed servers.

1. Ensure that the EWLM managed server is configured and started, and that it is communicating properly with its EWLM domain manager.

2. Ensure that the middleware application is installed and configured. Use the specific product documentation if you need assistance.
3. Ensure that the user account under which the application will run is properly authorized to the ARM services. For more information, see “Authorizing AIX users to ARM services” on page 120.
4. Set the following variable in the 32-bit or 64-bit DB2 instance that you want to monitor by EWLM:
`db2set DB2LIBPATH=/usr/lib`
5. Restart your DB2 instance.
6. Verify that ARM is enabled by using the lsarm command.

Note: The lsarm command displays information about applications instrumented with ARM and registered with the operating system. For more information on using this command, see “lsarm” on page 327.

7. Create and configure a Java Database Connectivity (JDBC) driver and data source in WebSphere Application Server. The JDBC driver allows an application to pass the ARM correlator from WebSphere Application Server to DB2. You do not need to modify the application if it is written to the J2EE specification level. To create the JDBC driver and data source, choose from the following topics the one that corresponds to your version of WebSphere Application Server and follow the instructions.
 - For WebSphere Application Server v5.1, see [Creating and configuring a JDBC provider and data source](#) .
 - For WebSphere Application Server v6.0, see [Configuring a JDBC provider and data source](#) .

Follow the instructions to create and configure your JDBC driver and data source, and set the custom properties and related items for your data source. Throughout the process, choose the options that best suit your environment and ensure you set the following properties as specified:

Table 45. General properties

Property	Value/Description
Database type	DB2
Provider type	DB2 Universal JDBC Driver Provider
Implementation type	Both options are supported. Choose the best one based on your application.
Class path	Input the path to the db2jcc.jar file that contains the JDBC provider code. In addition, the Universal JDBC driver requires that you add the license JAR file, db2jcc_license_cu.jar, to the classpath. Note: You should use variables for the path rather than hard-coding it.

Important: If transactions are received from the WebSphere Application Server, you must set the com.ibm.websphere.pmi.reqmetrics.PassCorrelatorToDB custom property to true when you configure your application in order for the ARM correlator to properly flow from WebSphere Application Server to DB2. For information on how to set this property, see the appropriate topic under “Enabling ARM on WebSphere Application Server 5.1.1 or 6.0” on page 215.

8. Test the connection by selecting the data source and clicking Test Connection. If successful, save the configuration. If unsuccessful, repeat the above steps and ensure that all fields are specified correctly.
9. Restart WebSphere Application Server.

As part of its instrumentation, DB2 has defined filters you can use to classify transactions into service classes. DB2 provides its instrumented filters to the domain manager in an XML document so the EWLM Control Center is aware of which filters apply specifically to DB2. The DB2 filters are preloaded into the

EWLM Control Center; all you must do is set up the application in the EWLM Control Center and specify which filters you want to use. The filters you specify during the set up are then available in the domain policy when you create transaction class rules. See “EWLM-supplied filter types” on page 35 for a complete list of filters available to all applications and “Application-specific filter types” on page 40 for additional filters for DB2.

Enabling ARM on DB2 Universal Database for use by WebSphere Application Server on i5/OS:

Follow the instructions in this topic to enable ARM on DB2 Universal Database for use by WebSphere Application Server on i5/OS.

ARM is enabled on DB2 Universal Database version 8 by default. However, there are several additional steps you must complete to deploy DB2 for use by WebSphere Application Server in an EWLM environment on your managed servers.

1. Ensure that the EWLM managed server is configured and started, and that it is communicating properly with its EWLM domain manager.
2. Ensure that the middleware application is installed and configured. Use the specific product documentation if you need assistance.
3. Ensure that the user account under which the application will run is properly authorized to the ARM services. For more information, see “Authorizing i5/OS users to ARM services” on page 138.
4. Use the WRKARMJOB CL command to verify that ARM support on your system is enabled and that an application is currently calling ARM APIs.

Note: This command will show DB2 as an ARM-enabled application only after WebSphere Application Server is restarted and is driving a workload to DB2.

5. Create and configure a Java Database Connectivity (JDBC) driver and data source in WebSphere Application Server. The JDBC driver allows an application to pass the ARM correlator from WebSphere Application Server to DB2. You do not need to modify the application if it is written to the J2EE specification level. To create the JDBC driver and data source, choose from the following topics the one that corresponds to your version of WebSphere Application Server and follow the instructions.

Note: For i5/OS V5R3, the WebSphere Application Server and DB2 applications must reside on the same system. Otherwise, EWLM will not collect the end-to-end transaction flow from WebSphere Application Server to DB2.

- For WebSphere Application Server v5.1, see [Creating and configuring a JDBC provider and data source](#).

- For WebSphere Application Server v6.0, see [Configuring a JDBC provider and data source](#).

Follow the instructions to create and configure your JDBC driver and data source, and set the custom properties and related items for your data source. Throughout the process, choose the options that best suit your environment and ensure you set the following properties as specified:

Table 46. General properties

Property	Value/Description
Database type	DB2
Provider type	Select any of the DB2 UDB for iSeries drivers. Note: ARM is supported on DB2/400 version 5 release 3 (V5R3) of the operating system. If your backend database is DB2/400 V5R2 or earlier, ARM transactions will not be supported in the database layer.

Table 46. General properties (continued)

Property	Value/Description
Name	Specifies one of the following values: <ul style="list-style-type: none"> DB2 UDB for iSeries (Toolbox XA) DB2 UDB for iSeries (Native XA) DB2 UDB for iSeries (Toolbox) DB2 UDB for iSeries (Native)
Class path	Specifies one of the following values (list is relative to list of values for Name): <ul style="list-style-type: none"> <code>\${OS400_TOOLBOX_JDBC_DRIVER_PATH}/jt400.jar</code> <code>\${OS400_NATIVE_JDBC_DRIVER_PATH}/db2_classes.jar</code> <code>\${OS400_TOOLBOX_JDBC_DRIVER_PATH}/jt400.jar</code> <code>\${OS400_NATIVE_JDBC_DRIVER_PATH}/db2_classes.jar</code>
Implementation class name	Specifies one of the following values (list is relative to list of values for Name): <ul style="list-style-type: none"> <code>com.ibm.db2.jdbc.app.UDBXADDataSource</code> <code>com.ibm.as400.access.AS400JDBCXADDataSource</code> <code>com.ibm.as400.access.AS400JDBCConnectionPoolDataSource</code> <code>com.ibm.db2.jdbc.app.UDBConnectionPoolDataSource</code>

Important: If transactions are received from the WebSphere Application Server, you must set the `com.ibm.websphere.pmi.reqmetrics.PassCorrelatorToDB` custom property to true when you configure your application in order for the ARM correlator to properly flow from WebSphere Application Server to DB2. For information on how to set this property, see the appropriate topic under “Enabling ARM on WebSphere Application Server 5.1.1 or 6.0” on page 215.

6. Restart WebSphere Application Server.

Enabling ARM on DB2 Universal Database for use by WebSphere Application Server on Linux:

Follow the instructions in this topic to enable ARM on DB2 Universal Database for use by WebSphere Application Server on Linux on xSeries.

ARM is enabled on DB2 Universal Database version 8 by default. However, there are several additional steps you must complete to deploy DB2 for use by WebSphere Application Server in an EWLM environment on your managed servers.

Note: The same pathing instructions exist for all middleware on Linux.

1. Ensure that the EWLM managed server is configured and started, and that it is communicating properly with its EWLM domain manager.
2. Ensure that the middleware application is installed and configured. Use the specific product documentation if you need assistance.
3. Ensure that the user account under which the application will run is properly authorized to the ARM services. For more information, see “Authorizing Linux users to ARM services” on page 148.
4. Set the following variable in the 32-bit DB2 instance that you want to monitor by EWLM:

```
db2set DB2LIBPATH=/usr/lib
```
5. Restart your DB2 instance.
6. Verify that ARM is enabled by using the `lsarm` command.

Note: The lsarm command displays information about applications instrumented with ARM and registered with the operating system. For more information on using this command, see “lsarm” on page 327.

7. Create and configure a Java Database Connectivity (JDBC) driver and data source in WebSphere Application Server. The JDBC driver allows an application to pass the ARM correlator from WebSphere Application Server to DB2. You do not need to modify the application if it is written to the J2EE specification level. To create the JDBC driver and data source, choose from the following topics the one that corresponds to your version of WebSphere Application Server and follow the instructions.
 - For WebSphere Application Server v5.1, see [Creating and configuring a JDBC provider and data source](#).
 - For WebSphere Application Server v6.0, see [Configuring a JDBC provider and data source](#).

Follow the instructions to create and configure your JDBC driver and data source, and set the custom properties and related items for your data source. Throughout the process, choose the options that best suit your environment and ensure you set the following properties as specified:

Table 47. General properties

Property	Value/Description
Database type	DB2
Provider type	DB2 Universal JDBC Driver Provider
Implementation type	Both options are supported. Choose the best one based on your application.
Class path	Input the path to the db2jcc.jar file that contains the JDBC provider code. In addition, the Universal JDBC driver requires that you add the license JAR file, db2jcc_license_cu.jar, to the classpath. Note: You should use variables for the path rather than hard-coding it.

Important: If transactions are received from the WebSphere Application Server, you must set the com.ibm.websphere.pmi.reqmetrics.PassCorrelatorToDB custom property to true when you configure your application in order for the ARM correlator to properly flow from WebSphere Application Server to DB2. For information on how to set this property, see the appropriate topic under “Enabling ARM on WebSphere Application Server 5.1.1 or 6.0” on page 215.

8. Test the connection by selecting the data source and clicking Test Connection. If successful, save the configuration. If unsuccessful, repeat the above steps and ensure that all fields are specified correctly.
9. Restart WebSphere Application Server.

As part of its instrumentation, DB2 has defined filters you can use to classify transactions into service classes. DB2 provides its instrumented filters to the domain manager in an XML document so the EWLM Control Center is aware of which filters apply specifically to DB2. The DB2 filters are preloaded into the EWLM Control Center; all you must do is set up the application in the EWLM Control Center and specify which filters you want to use. The filters you specify during the set up are then available in the domain policy when you create transaction class rules. See “EWLM-supplied filter types” on page 35 for a complete list of filters available to all applications and “Application-specific filter types” on page 40 for additional filters for DB2.



Enabling ARM on DB2 Universal Database for use by WebSphere Application Server on Solaris:

Follow the instructions in this topic to enable ARM on DB2 Universal Database for use by WebSphere Application Server on Solaris.

ARM is enabled on DB2 Universal Database version 8 by default. However, there are several additional steps you must complete to deploy DB2 for use by WebSphere Application Server in an EWLM environment on your managed servers.

1. Ensure that the EWLM managed server is configured and started, and that it is communicating properly with its EWLM domain manager.
2. Ensure that the middleware application is installed and configured. Use the specific product documentation if you need assistance.
3. Ensure that the user account under which the application will run is properly authorized to the ARM services. For more information, see “Authorizing Solaris users to ARM services” on page 156.
4. Set the following variable in the DB2 instance that you want to monitor by EWLM:
 - For 32-bit DB2 instances, use:
db2set DB2LIBPATH=/usr/lib
 - For 64-bit DB2 instances, use:
db2set DB2LIBPATH=/usr/lib/sparcv9
5. Restart your DB2 instance.
6. Verify that ARM is enabled by using the lsarm command.

Note: The lsarm command displays information about applications instrumented with ARM and registered with the operating system. For more information on using this command, see “lsarm” on page 327.

7. Create and configure a Java Database Connectivity (JDBC) driver and data source in WebSphere Application Server. The JDBC driver allows an application to pass the ARM correlator from WebSphere Application Server to DB2. You do not need to modify the application if it is written to the J2EE specification level. To create the JDBC driver and data source, choose from the following topics the one that corresponds to your version of WebSphere Application Server and follow the instructions.
 - For WebSphere Application Server v5.1, see [Creating and configuring a JDBC provider and data source](#) .
 - For WebSphere Application Server v6.0, see [Configuring a JDBC provider and data source](#) .

Follow the instructions to create and configure your JDBC driver and data source, and set the custom properties and related items for your data source. Throughout the process, choose the options that best suit your environment and ensure you set the following properties as specified:

Table 48. General properties

Property	Value/Description
Database type	DB2
Provider type	DB2 Universal JDBC Driver Provider
Implementation type	Both options are supported. Choose the best one based on your application.
Class path	Input the path to the db2jcc.jar file that contains the JDBC provider code. In addition, the Universal JDBC driver requires that you add the license JAR file, db2jcc_license_cu.jar, to the classpath. Note: You should use variables for the path rather than hard-coding it.

Important: If transactions are received from the WebSphere Application Server, you must set the com.ibm.websphere.pmi.reqmetrics.PassCorrelatorToDB custom property to true when you configure your application in order for the ARM correlator to properly flow from

WebSphere Application Server to DB2. For information on how to set this property, see the appropriate topic under “Enabling ARM on WebSphere Application Server 5.1.1 or 6.0” on page 215.

8. Test the connection by selecting the data source and clicking Test Connection. If successful, save the configuration. If unsuccessful, repeat the above steps and ensure that all fields are specified correctly.
9. Restart WebSphere Application Server.

As part of its instrumentation, DB2 has defined filters you can use to classify transactions into service classes. DB2 provides its instrumented filters to the domain manager in an XML document so the EWLM Control Center is aware of which filters apply specifically to DB2. The DB2 filters are preloaded into the EWLM Control Center; all you must do is set up the application in the EWLM Control Center and specify which filters you want to use. The filters you specify during the set up are then available in the domain policy when you create transaction class rules. See “EWLM-supplied filter types” on page 35 for a complete list of filters available to all applications and “Application-specific filter types” on page 40 for additional filters for DB2.

Enabling ARM on DB2 Universal Database for use by WebSphere Application Server on Windows:

Follow the instructions in this topic to enable ARM on DB2 Universal Database for use by WebSphere Application Server on Windows.

ARM is enabled on DB2 Universal Database version 8 by default. However, there are several additional steps you must complete to deploy DB2 for use by WebSphere Application Server in an EWLM environment on your managed servers.

1. Ensure that the EWLM managed server is configured and started, and that it is communicating properly with its EWLM domain manager.
2. Ensure that the middleware application is installed and configured. Use the specific product documentation if you need assistance.
3. Ensure that the user account under which the application will run is properly authorized to the ARM services. For more information, see “Authorizing Windows users to ARM services” on page 166.
4. Verify that ARM is enabled by using the `ewlmWinADTool.exe` program.

Note: `ewlmWinADTool` displays information about applications instrumented with ARM and registered with the operating system. For more information on using this command, see “`ewlmWinADTool.exe`” on page 328.

5. Create and configure a Java Database Connectivity (JDBC) driver and data source in WebSphere Application Server. The JDBC driver allows an application to pass the ARM correlator from WebSphere Application Server to DB2. You do not need to modify the application if it is written to the J2EE specification level. To create the JDBC driver and data source, choose from the following topics the one that corresponds to your version of WebSphere Application Server and follow the instructions.

- For WebSphere Application Server v5.1, see [Creating and configuring a JDBC provider and data source](#) .

- For WebSphere Application Server v6.0, see [Configuring a JDBC provider and data source](#) .

Follow the instructions to create and configure your JDBC driver and data source, and set the custom properties and related items for your data source. Throughout the process, choose the options that best suit your environment and ensure you set the following properties as specified:

Table 49. General properties

Property	Value/Description
Database type	DB2
Provider type	DB2 Universal JDBC Driver Provider

Table 49. General properties (continued)

Property	Value/Description
Implementation type	Both options are supported. Choose the best one based on your application.
Class path	Input the path to the db2jcc.jar file that contains the JDBC provider code. In addition, the Universal JDBC driver requires that you add the license JAR file, db2jcc_license_cu.jar, to the classpath. Note: You should use variables for the path rather than hard-coding it.

Important: If transactions are received from the WebSphere Application Server, you must set the `com.ibm.websphere.pmi.reqmetrics.PassCorrelatorToDB` custom property to true when you configure your application in order for the ARM correlator to properly flow from WebSphere Application Server to DB2. For information on how to set this property, see the appropriate topic under “Enabling ARM on WebSphere Application Server 5.1.1 or 6.0” on page 215.

6. Test the connection by selecting the data source and clicking Test Connection. If successful, save the configuration. If unsuccessful, repeat the above steps and ensure that all fields are specified correctly.
7. Restart WebSphere Application Server.

As part of its instrumentation, DB2 has defined filters you can use to classify transactions into service classes. DB2 provides its instrumented filters to the domain manager in an XML document so the EWLM Control Center is aware of which filters apply specifically to DB2. The DB2 filters are preloaded into the EWLM Control Center; all you must do is set up the application in the EWLM Control Center and specify which filters you want to use. The filters you specify during the set up are then available in the domain policy when you create transaction class rules. See “EWLM-supplied filter types” on page 35 for a complete list of filters available to all applications and “Application-specific filter types” on page 40 for additional filters for DB2.


Enabling ARM on DB2 Universal Database for use by WebSphere Application Server on z/OS:

Follow the instructions in this topic to enable ARM on DB2 Universal Database for use by WebSphere Application Server on z/OS.

ARM is enabled on DB2 Universal Database version 8 by default. However, there are several additional steps you must complete to deploy DB2 for use by WebSphere Application Server in an EWLM environment on your managed servers.

1. Ensure that the EWLM managed server is configured and started, and that it is communicating properly with its EWLM domain manager.
2. Ensure that the middleware application is installed and configured. Use the specific product documentation if you need assistance.
 - a. Ensure the DB2 ZPARM `CMTSTAT`, which is a parameter of the `DSN6FAC` macro, is set to `INACTIVE`. This parameter is similar to setting `DDF THREADS = INACTIVE` using the installation panel `DSNTIPR`. See the section “Distributed data facility panel 1: `DSNTIPR`” in the *DB2 UDB for z/OS V8 Installation Guide* (GC18-7418-03) for more information.
3. Ensure that the user account under which the application will run is properly authorized to the ARM services. For more information, see “Authorizing z/OS users to ARM services” on page 188.
4. Verify that ARM is enabled. For more information on how to do this, see “Verifying the status of ARM services on z/OS” on page 189.
5. Create and configure a Java Database Connectivity (JDBC) driver and data source in WebSphere Application Server. The JDBC driver allows an application to pass the ARM correlator from WebSphere Application Server to DB2. You do not need to modify the application if it is written to the

J2EE specification level. To create the JDBC driver and data source, choose from the following topics the one that corresponds to your version of WebSphere Application Server and follow the instructions.

- For WebSphere Application Server v5.1, see [Creating and configuring a JDBC provider and data source](#) .

- For WebSphere Application Server v6.0, see [Configuring a JDBC provider and data source](#) .

Follow the instructions to create and configure your JDBC driver and data source, and set the custom properties and related items for your data source. Throughout the process, choose the options that best suit your environment and ensure you set the following properties as specified:

Table 50. General properties

Property	Value/Description
Database type	DB2
Provider type	DB2 Universal JDBC Driver Provider
Implementation type	Both options are supported. Choose the best one based on your application.
Class path	Input the path to the db2jcc.jar file that contains the JDBC provider code. In addition, the Universal JDBC driver requires that you add the license JAR file, db2jcc_license_cu.jar, to the classpath. Note: You should use variables for the path rather than hard-coding it.

Important: If transactions are received from the WebSphere Application Server, you must set the `com.ibm.websphere.pmi.reqmetrics.PassCorrelatorToDB` custom property to true when you configure your application in order for the ARM correlator to properly flow from WebSphere Application Server to DB2. For information on how to set this property, see the appropriate topic under “Enabling ARM on WebSphere Application Server 5.1.1 or 6.0” on page 215.

6. Test the connection by selecting the data source and clicking Test Connection. If successful, save the configuration. If unsuccessful, repeat the above steps and ensure that all fields are specified correctly.
7. Restart WebSphere Application Server.

Note: If you find that a particular DB2 transaction is undesirably classified to the system default transaction class, it is most likely because its associated inbound correlator is unknown. To qualify as known, a correlator’s upstream hop to DB2 for z/OS must come from an EWLM ARM-enabled application that resides in the same EWLM domain. With such a correlator, the DB2 transaction retains the upstream classification result. However, if you are running a configuration where the correlator’s previous hop comes from an application that is not EWLM ARM-enabled, or if it comes from another domain, then EWLM cannot interpret the inbound correlator to DB2 and the transaction is automatically classified to the system default transaction class. This is generally undesirable because you need EWLM to interpret the inbound correlators in order to produce usable correlated DB2 for z/OS data. Keep in mind:

- In the event there is no correlator in the first place, DB2 for z/OS will not initiate EWLM ARM processing for the transaction at all. Note, however, that you might still see DB2 as hop 0.
- If your transaction is classified to the system default transaction class, that indicates that your inbound correlators are not useful for establishing the transaction class for the DB2 transaction.

Related information

DB2 UDB for z/OS V8 Installation Guide (GC18-7418-03)

Enabling ARM on WebSphere Application Server 5.1.1 or 6.0

Use this topic to select the version of WebSphere Application server you are running.

Select from below the version of WebSphere Application server you are running.

Enabling ARM on WebSphere Application Server 5.1.1:

Use this topic to enable ARM on WebSphere Application Server 5.1.1.

The specific tasks you must complete to enable ARM vary depending on the target operating system. Select from the list below a topic collection that is specific to the operating system on which you want to enable ARM on WebSphere Application Server 5.1.1.

Enabling ARM on WebSphere Application Server 5.1.1 on AIX:

Follow the instructions in this topic to enable ARM on WebSphere Application Server 5.1.1 on AIX.

Note: For more information about WebSphere Application Server and the administrative console, refer to the administrative console online help in the WebSphere Application Server 5.1 information center



(<http://publib.boulder.ibm.com/infocenter/ws51help/>) .

1. Ensure that the EWLM managed server is configured and started, and that it is communicating properly with its EWLM domain manager.
2. Ensure that the middleware application is installed and configured. Use the specific product documentation if you need assistance.
3. Ensure that the user account under which the application will run is properly authorized to the ARM services. For more information, see “Authorizing AIX users to ARM services” on page 120.
4. Start the WebSphere Application Server.
5. From a Web browser, go to and log in to the administrative console.



Note: The default port is 9090 for HTTP and 9043 for HTTPS.

- <http://hostname:9090/admin>
- <https://hostname:9043/admin>

where *hostname* is the hostname or IP address for the node. The administrative console will load into the browser and you will be asked to login. Log in to the administrative console under the appropriate user account.

6. Enable request metrics. To do this, follow the instructions in Enabling Performance Monitoring Infrastructure Request Metrics (http://publib.boulder.ibm.com/infocenter/wasinfo/v5r1//topic/com.ibm.websphere.base.doc/info/aes/ae/tprf_rqenable.html) .
7. Enable ARM. To do this, follow the instructions in Enabling Application Response Measurement (http://publib.boulder.ibm.com/infocenter/wasinfo/v5r1//topic/com.ibm.websphere.base.doc/info/aes/ae/tprf_arm1.html)  and its subtopics.

Note: If you use the GenPluginCfg.sh utility included by WebSphere Application Server to generate the configuration for WebSphere Application Server plug-ins, the plug-ins that the utility generates will be ARM enabled. If you do not want the utility to enable ARM, you must edit the plug-in configuration file, plug-in-cfg.xml. The default location for this file is `WAS_HOME/config/cells/plugin-cfg.xml`. Open the file in an editor and change `<RequestMetrics armEnabled="true" newBehavior="false" rmEnabled="true" tracelevel="HOPS">` to `<RequestMetrics armEnabled="false" newBehavior="false" rmEnabled="true" tracelevel="HOPS">`.

8. Set the **Trace level** to Hops. To do this, follow the instructions in Setting the trace level in Performance Monitoring Infrastructure Request Metrics (http://publib.boulder.ibm.com/infocenter/wasinfo/v5r1//topic/com.ibm.websphere.base.doc/info/aes/ae/uprf_settrace.html) .
9. Set the custom properties listed in the following table. To do this, use the instructions in Setting Custom JVM properties to choose an ARM agent (http://publib.boulder.ibm.com/infocenter/wasinfo/v5r1//topic/com.ibm.websphere.base.doc/info/aes/ae/xprf_jvmarm.html) .

Note: "32-bit JNI library" in the table below refers to the bit level of the JVM, not that of the operating system.

Table 51. Custom properties

Property	Value
ArmTransactionFactory	(All on one line): com.ibm.wlm.arm40SDK.transaction. Arm40TransactionFactory
com.ibm.websphere.pmi.reqmetrics.ARMIMPL	arm4
java.library.path	For 32-bit JNI library, use: /usr/lib
ws.ext.dirs	/opt/IBM/VE2/EWLMMS/classes/ARM
Optional. com.ibm.websphere.pmi.reqmetrics.loggingEnabled	false
com.ibm.websphere.pmi.reqmetrics.PassCorrelatorToDB	Choose the proper setting based on your configuration: <ul style="list-style-type: none"> • If the next hop database is ARM instrumented, set this property to true. • If the next hop database is not ARM instrumented, either set this property to false or do not set it at all. This avoids any unnecessary overhead in the JDBC calls.


10. Restart WebSphere Application Server.
11. Verify that ARM is enabled by using the lsarm command.

Note: The lsarm command displays information about applications instrumented with ARM and registered with the operating system. For more information on using this command, see "lsarm" on page 327.

As part of its instrumentation, WebSphere Application Server has defined filters that you can use to classify transactions into service classes. WebSphere Application Server provides its instrumented filters to the domain manager in an XML document, so the EWLM Control Center is aware of which filters apply specifically to WebSphere Application Server. The WebSphere Application Server filters are preloaded into the EWLM Control Center; all you must do is set up the application in the EWLM Control Center and specify what filters you want to use. The filters you specify during the setup are available in the domain policy when you create transaction class rules. See "EWLM-supplied filter types" on page 35 for a complete list of filters available to all applications and "Application-specific filter types" on page 40 for additional filters for WebSphere Application Server.

Enabling ARM on WebSphere Application Server 5.1.1 on HP-UX:

Follow the instructions in this topic to enable ARM on WebSphere Application Server 5.1.1 on HP-UX.

Note: For more information about WebSphere Application Server and the administrative console, refer to the administrative console online help in the WebSphere Application Server 5.1 information center (<http://publib.boulder.ibm.com/infocenter/ws51help/>) .

1. Ensure that the EWLM managed server is configured and started, and that it is communicating properly with its EWLM domain manager.
2. Ensure that the middleware application is installed and configured. Use the specific product documentation if you need assistance.
3. Ensure that the user account under which the application will run is properly authorized to the ARM services. For more information, see “Authorizing HP-UX users to ARM services” on page 126.
4. Ensure that the instrumented application can locate the EWLM ARM libraries. To do this, set the `SHLIB_PATH` environment variable for the application instrumented with ARM in either the user’s login environment or in a script file that is used to launch the application.

For 32-bit applications instrumented with ARM, set the following environment variable:

```
export SHLIB_PATH="/opt/IBM/VE2/EWLM/usr/lib"
```

For 64-bit applications instrumented with ARM, set the following environment variable:

```
export SHLIB_PATH="/opt/IBM/VE2/EWLM/usr/lib/pa20_64"
```


Note: If you start the application from a Telnet session, you can update the `.profile` for the user to reflect the `SHLIB_PATH` setting as shown above. If you start the application from a script, the script should export the `SHLIB_PATH` environment variable before starting the application.


5. Start the WebSphere Application Server.
6. From a Web browser, go to and log in to the administrative console.

Note: The default port is 9090 for HTTP and 9043 for HTTPS.


- `http://hostname:9090/admin`
- `https://hostname:9043/admin`


where *hostname* is the hostname or IP address for the node. The administrative console will load into the browser and you will be asked to login. Log in to the administrative console under the appropriate user account.

7. Enable request metrics. To do this, follow the instructions in Enabling Performance Monitoring Infrastructure Request Metrics (http://publib.boulder.ibm.com/infocenter/wasinfo/v5r1//topic/com.ibm.websphere.base.doc/info/aes/ae/tprf_rqenable.html) .

8. Enable ARM. To do this, follow the instructions in Enabling Application Response Measurement (http://publib.boulder.ibm.com/infocenter/wasinfo/v5r1//topic/com.ibm.websphere.base.doc/info/aes/ae/tprf_arm1.html)  and its subtopics.

Note: If you use the `GenPluginCfg.sh` utility included by WebSphere Application Server to generate the configuration for WebSphere Application Server plug-ins, the plug-ins that the utility generates will be ARM enabled. If you do not want the utility to enable ARM, you must edit the plug-in configuration file, `plug-in-cfg.xml`. The default location for this file is `WAS_HOME/config/cells/plugin-cfg.xml`. Open the file in an editor and change `<RequestMetrics armEnabled="true" newBehavior="false" rmEnabled="true" traceLevel="HOPS">` to `<RequestMetrics armEnabled="false" newBehavior="false" rmEnabled="true" traceLevel="HOPS">`.

9. Set the **Trace level** to Hops. To do this, follow the instructions in Setting the trace level in Performance Monitoring Infrastructure Request Metrics (http://publib.boulder.ibm.com/infocenter/wasinfo/v5r1//topic/com.ibm.websphere.base.doc/info/aes/ae/uprf_settrace.html) .

10. Set the custom properties listed in the following table. To do this, use the instructions in Setting Custom JVM properties to choose an ARM agent (http://publib.boulder.ibm.com/infocenter/wasinfo/v5r1//topic/com.ibm.websphere.base.doc/info/aes/ae/xprf_jvmarm.html) .

Note: "32-bit JNI library" in the table below refers to the bit level of the JVM, not that of the operating system.

Table 52. Custom properties

Property	Value
ArmTransactionFactory	(All on one line): com.ibm.wlm.arm40SDK.transaction. Arm40TransactionFactory
com.ibm.websphere.pmi.reqmetrics.ARMIMPL	arm4
java.library.path	For 32-bit JNI library, use: /usr/lib
ws.ext.dirs	/opt/IBM/VE2/EWLMMS/classes/ARM
Optional. com.ibm.websphere.pmi.reqmetrics.loggingEnabled	false
com.ibm.websphere.pmi.reqmetrics.PassCorrelatorToDB	Choose the proper setting based on your configuration: <ul style="list-style-type: none"> • If the next hop database is ARM instrumented, set this property to true. • If the next hop database is not ARM instrumented, either set this property to false or do not set it at all. This avoids any unnecessary overhead in the JDBC calls.

- Restart WebSphere Application Server.
- Verify that ARM is enabled by using the lsarm command.

Note: The lsarm command displays information about applications instrumented with ARM and registered with the operating system. For more information on using this command, see "lsarm" on page 327.

As part of its instrumentation, WebSphere Application Server has defined filters that you can use to classify transactions into service classes. WebSphere Application Server provides its instrumented filters to the domain manager in an XML document, so the EWLM Control Center is aware of which filters apply specifically to WebSphere Application Server. The WebSphere Application Server filters are preloaded into the EWLM Control Center; all you must do is set up the application in the EWLM Control Center and specify what filters you want to use. The filters you specify during the setup are available in the domain policy when you create transaction class rules. See "EWLM-supplied filter types" on page 35 for a complete list of filters available to all applications and "Application-specific filter types" on page 40 for additional filters for WebSphere Application Server.

Enabling ARM on WebSphere Application Server 5.1.1 on i5/OS:

Follow the instructions in this topic to enable ARM on WebSphere Application Server 5.1.1 on i5/OS.



Note: For more information about WebSphere Application Server and the administrative console, refer to the administrative console online help in the WebSphere Application Server 5.1 information center (<http://publib.boulder.ibm.com/infocenter/ws51help/>) .

1. Ensure that the EWLM managed server is configured and started, and that it is communicating properly with its EWLM domain manager.
2. Ensure that the middleware application is installed and configured. Use the specific product documentation if you need assistance.
3. Ensure that the user account under which the application will run is properly authorized to the ARM services. For more information, see “Authorizing i5/OS users to ARM services” on page 138.
4. Start the WebSphere Application Server.
5. From a Web browser, go to and log in to the administrative console.

Note: The default port is 9090 for HTTP and 9043 for HTTPS.

- `http://hostname:9090/admin`
- `https://hostname:9043/admin`

where *hostname* is the hostname or IP address for the node. The administrative console will load into the browser and you will be asked to login. Log in to the administrative console under the appropriate user account.

6. Enable request metrics. To do this, follow the instructions in Enabling Performance Monitoring Infrastructure Request Metrics (http://publib.boulder.ibm.com/infocenter/wasinfo/v5r1//topic/com.ibm.websphere.base.doc/info/aes/ae/tprf_rqenable.html)  .
7. Enable ARM. To do this, follow the instructions in Enabling Application Response Measurement (http://publib.boulder.ibm.com/infocenter/wasinfo/v5r1//topic/com.ibm.websphere.base.doc/info/aes/ae/tprf_arm1.html)  and its subtopics.

Note: If you use the `GenPluginCfg.sh` utility included by WebSphere Application Server to generate the configuration for WebSphere Application Server plug-ins, the plug-ins that the utility generates will be ARM enabled. If you do not want the utility to enable ARM, you must edit the plug-in configuration file, `plug-in-cfg.xml`. The default location for this file is `WAS_HOME/config/cells/plugin-cfg.xml`. Open the file in an editor and change `<RequestMetrics armEnabled="true" newBehavior="false" rmEnabled="true" traceLevel="HOPS">` to `<RequestMetrics armEnabled="false" newBehavior="false" rmEnabled="true" traceLevel="HOPS">`.



8. Set the **Trace level** to Hops. To do this, follow the instructions in Setting the trace level in Performance Monitoring Infrastructure Request Metrics (http://publib.boulder.ibm.com/infocenter/wasinfo/v5r1//topic/com.ibm.websphere.base.doc/info/aes/ae/uprf_settrace.html)  .
9. Set the custom properties listed in the following table. To do this, use the instructions in Setting Custom JVM properties to choose an ARM agent (http://publib.boulder.ibm.com/infocenter/wasinfo/v5r1//topic/com.ibm.websphere.base.doc/info/aes/ae/xprf_jvmarm.html)  .

Table 53. Custom properties

Property	Value
ArmTransactionFactory	(All on one line): com.ibm.wlm.arm40SDK.transaction. Arm40TransactionFactory
com.ibm.websphere.pmi.reqmetrics.ARMIMPL	arm4
java.library.path	/QSYS.LIB/QWLM2.LIB
ws.ext.dirs	/QIBM/ProdData/VE2/EWLMMS/classes/ARM
Optional. com.ibm.websphere.pmi.reqmetrics.loggingEnabled	false

Table 53. Custom properties (continued)

Property	Value
com.ibm.websphere.pmi.reqmetrics.PassCorrelatorToDB	Choose the proper setting based on your configuration: <ul style="list-style-type: none"> • If the next hop database is ARM instrumented, set this property to true. • If the next hop database is not ARM instrumented, either set this property to false or do not set it at all. This avoids any unnecessary overhead in the JDBC calls.

- Restart WebSphere Application Server.
- Use the WRKARMJOB CL command to verify that ARM support on your system is enabled and that an application is currently calling ARM APIs.

As part of its instrumentation, WebSphere Application Server has defined filters that you can use to classify transactions into service classes. WebSphere Application Server provides its instrumented filters to the domain manager in an XML document, so the EWLM Control Center is aware of which filters apply specifically to WebSphere Application Server. The WebSphere Application Server filters are preloaded into the EWLM Control Center; all you must do is set up the application in the EWLM Control Center and specify what filters you want to use. The filters you specify during the setup are available in the domain policy when you create transaction class rules. See “EWLM-supplied filter types” on page 35 for a complete list of filters available to all applications and “Application-specific filter types” on page 40 for additional filters for WebSphere Application Server.

Enabling ARM on WebSphere Application Server 5.1.1 on Solaris:

Follow the instructions in this topic to enable ARM on WebSphere Application Server 5.1.1 on Solaris.

Note: For more information about WebSphere Application Server and the administrative console, refer to the administrative console online help in the WebSphere Application Server 5.1 information center


(<http://publib.boulder.ibm.com/infocenter/ws51help/>) .


- Ensure that the EWLM managed server is configured and started, and that it is communicating properly with its EWLM domain manager.
- Ensure that the middleware application is installed and configured. Use the specific product documentation if you need assistance.
- Ensure that the user account under which the application will run is properly authorized to the ARM services. For more information, see “Authorizing Solaris users to ARM services” on page 156.
- Start the WebSphere Application Server.
- From a Web browser, go to and log in to the administrative console.

Note: The default port is 9090 for HTTP and 9043 for HTTPS.



- <http://hostname:9090/admin>
- <https://hostname:9043/admin>

where *hostname* is the hostname or IP address for the node. The administrative console will load into the browser and you will be asked to login. Log in to the administrative console under the appropriate user account.

- Enable request metrics. To do this, follow the instructions in Enabling Performance Monitoring Infrastructure Request Metrics (http://publib.boulder.ibm.com/infocenter/wasinfo/v5r1//topic/com.ibm.websphere.base.doc/info/aes/ae/tprf_rqenable.html) .

7. Enable ARM. To do this, follow the instructions in Enabling Application Response Measurement (http://publib.boulder.ibm.com/infocenter/wasinfo/v5r1//topic/com.ibm.websphere.base.doc/info/aes/ae/tprf_arm1.html)  and its subtopics.

Note: If you use the GenPluginCfg.sh utility included by WebSphere Application Server to generate the configuration for WebSphere Application Server plug-ins, the plug-ins that the utility generates will be ARM enabled. If you do not want the utility to enable ARM, you must edit the plug-in configuration file, plug-in-cfg.xml. The default location for this file is `WAS_HOME/config/cells/plugin-cfg.xml`. Open the file in an editor and change `<RequestMetrics armEnabled="true" newBehavior="false" rmEnabled="true" traceLevel="HOPS">` to `<RequestMetrics armEnabled="false" newBehavior="false" rmEnabled="true" traceLevel="HOPS">`.

8. Set the **Trace level** to Hops. To do this, follow the instructions in Setting the trace level in Performance Monitoring Infrastructure Request Metrics (http://publib.boulder.ibm.com/infocenter/wasinfo/v5r1//topic/com.ibm.websphere.base.doc/info/aes/ae/uprf_settrace.html) .
9. Set the custom properties listed in the following table. To do this, use the instructions in Setting Custom JVM properties to choose an ARM agent (http://publib.boulder.ibm.com/infocenter/wasinfo/v5r1//topic/com.ibm.websphere.base.doc/info/aes/ae/xprf_jvmarm.html) .

Note: "32-bit JNI library" in the table below refers to the bit level of the JVM, not that of the operating system.

Table 54. Custom properties

Property	Value
ArmTransactionFactory	(All on one line): com.ibm.wlm.arm40SDK.transaction. Arm40TransactionFactory
com.ibm.websphere.pmi.reqmetrics.ARMIMPL	arm4
java.library.path	For 32-bit JNI library, use: /usr/lib
ws.ext.dirs	/opt/IBM/VE2/EWLMMS/classes/ARM
Optional. com.ibm.websphere.pmi.reqmetrics.loggingEnabled	false
com.ibm.websphere.pmi.reqmetrics.PassCorrelatorToDB	Choose the proper setting based on your configuration: <ul style="list-style-type: none"> • If the next hop database is ARM instrumented, set this property to true. • If the next hop database is not ARM instrumented, either set this property to false or do not set it at all. This avoids any unnecessary overhead in the JDBC calls.

10. Restart WebSphere Application Server.
11. Verify that ARM is enabled by using the lsarm command.

Note: The lsarm command displays information about applications instrumented with ARM and registered with the operating system. For more information on using this command, see "lsarm" on page 327.

As part of its instrumentation, WebSphere Application Server has defined filters that you can use to classify transactions into service classes. WebSphere Application Server provides its instrumented filters

to the domain manager in an XML document, so the EWLM Control Center is aware of which filters apply specifically to WebSphere Application Server. The WebSphere Application Server filters are preloaded into the EWLM Control Center; all you must do is set up the application in the EWLM Control Center and specify what filters you want to use. The filters you specify during the setup are available in the domain policy when you create transaction class rules. See “EWLM-supplied filter types” on page 35 for a complete list of filters available to all applications and “Application-specific filter types” on page 40 for additional filters for WebSphere Application Server.

Enabling ARM on WebSphere Application Server 5.1.1 on Windows:

Follow the instructions in this topic to enable ARM on WebSphere Application Server 5.1.1 on Windows.

Note: For more information about WebSphere Application Server and the administrative console, refer to the administrative console online help in the WebSphere Application Server 5.1 information center



(<http://publib.boulder.ibm.com/infocenter/ws51help/>)  .

1. Ensure that the EWLM managed server is configured and started, and that it is communicating properly with its EWLM domain manager.
2. Ensure that the middleware application is installed and configured. Use the specific product documentation if you need assistance.
3. Ensure that the user account under which the application will run is properly authorized to the ARM services. For more information, see “Authorizing Windows users to ARM services” on page 166.
4. Start the WebSphere Application Server.
5. From a Web browser, go to and log in to the administrative console.


Note: The default port is 9090 for HTTP and 9043 for HTTPS.


- <http://hostname:9090/admin>
- <https://hostname:9043/admin>

where *hostname* is the hostname or IP address for the node. The administrative console will load into the browser and you will be asked to login. Log in to the administrative console under the appropriate user account.

6. Enable request metrics. To do this, follow the instructions in Enabling Performance Monitoring Infrastructure Request Metrics (http://publib.boulder.ibm.com/infocenter/wasinfo/v5r1//topic/com.ibm.websphere.base.doc/info/aes/ae/tprf_rqenable.html)  .
7. Enable ARM. To do this, follow the instructions in Enabling Application Response Measurement (http://publib.boulder.ibm.com/infocenter/wasinfo/v5r1//topic/com.ibm.websphere.base.doc/info/aes/ae/tprf_arm1.html)  and its subtopics.

Note: If you use the GenPluginCfg.sh utility included by WebSphere Application Server to generate the configuration for WebSphere Application Server plug-ins, the plug-ins that the utility generates will be ARM enabled. If you do not want the utility to enable ARM, you must edit the plug-in configuration file, `plug-in-cfg.xml`. The default location for this file is `WAS_HOME/config/cells/plugin-cfg.xml`. Open the file in an editor and change `<RequestMetrics armEnabled="true" newBehavior="false" rmEnabled="true" tracelevel="HOPS">` to `<RequestMetrics armEnabled="false" newBehavior="false" rmEnabled="true" tracelevel="HOPS">`.

8. Set the **Trace level** to Hops. To do this, follow the instructions in Setting the trace level in Performance Monitoring Infrastructure Request Metrics (http://publib.boulder.ibm.com/infocenter/wasinfo/v5r1//topic/com.ibm.websphere.base.doc/info/aes/ae/uprf_settrace.html)  .

9. Set the custom properties listed in the following table. To do this, use the instructions in Setting Custom JVM properties to choose an ARM agent (http://publib.boulder.ibm.com/infocenter/wasinfo/v5r1//topic/com.ibm.websphere.base.doc/info/aes/ae/xprf_jvmarm.html) .

Note: "32-bit JNI library" in the table below refers to the bit level of the JVM, not that of the operating system.

Table 55. Custom properties

Property	Value
ArmTransactionFactory	(All on one line): com.ibm.wlm.arm40SDK.transaction. Arm40TransactionFactory
com.ibm.websphere.pmi.reqmetrics.ARMIMPL	arm4
java.library.path	For 32-bit JNI library, use: C:\Program Files\IBM\VE2\EWLMMS\classes\ms
ws.ext.dirs	C:\Program Files\IBM\VE2\EWLMMS\classes\ARM
Optional. com.ibm.websphere.pmi.reqmetrics.loggingEnabled	false
com.ibm.websphere.pmi.reqmetrics.PassCorrelatorToDB	Choose the proper setting based on your configuration: <ul style="list-style-type: none"> • If the next hop database is ARM instrumented, set this property to true. • If the next hop database is not ARM instrumented, either set this property to false or do not set it at all. This avoids any unnecessary overhead in the JDBC calls.

10. Restart WebSphere Application Server.
11. Verify that ARM is enabled by using the `ewlmWinADTool.exe` program.

Note: `ewlmWinADTool` displays information about applications instrumented with ARM and registered with the operating system. For more information on using this command, see "ewlmWinADTool.exe" on page 328.

As part of its instrumentation, WebSphere Application Server has defined filters that you can use to classify transactions into service classes. WebSphere Application Server provides its instrumented filters to the domain manager in an XML document, so the EWLM Control Center is aware of which filters apply specifically to WebSphere Application Server. The WebSphere Application Server filters are preloaded into the EWLM Control Center; all you must do is set up the application in the EWLM Control Center and specify what filters you want to use. The filters you specify during the setup are available in the domain policy when you create transaction class rules. See "EWLM-supplied filter types" on page 35 for a complete list of filters available to all applications and "Application-specific filter types" on page 40 for additional filters for WebSphere Application Server.

Enabling ARM on WebSphere Application Server 6.0:

Use this topic to enable ARM on WebSphere Application Server 6.0.

The specific tasks you must complete to enable ARM vary depending on the target operating system. Select from the list below a topic collection that is specific to the operating system on which you want to enable ARM on WebSphere Application Server 6.0.

Enabling ARM on WebSphere Application Server 6.0 on AIX:

Follow the instructions in this topic to enable ARM on WebSphere Application Server 6.0 on AIX.

Note: For more information about WebSphere Application Server and the administrative console, refer to the administrative console online help in the WebSphere Application Server 6.0 information center

(<http://publib.boulder.ibm.com/infocenter/ws60help/>)  .

1. Ensure that the EWLM managed server is configured and started, and that it is communicating properly with its EWLM domain manager.
2. Ensure that the middleware application is installed and configured. Use the specific product documentation if you need assistance.
3. Ensure that the user account under which the application will run is properly authorized to the ARM services. For more information, see “Authorizing AIX users to ARM services” on page 120.
4. Start the WebSphere Application Server.


5. From a Web browser, go to and log in to the administrative console. Go to

`http://hostname:9060/ibm/console`

where *hostname* is the hostname or IP address for the node. The administrative console will load into the browser and you will be asked to login. Log in to the administrative console under the appropriate user account.


Note: The default port is 9060.

6. Enable request metrics. Use the instructions in Getting performance data from request metrics (<http://publib.boulder.ibm.com/infocenter/wasinfo/v6r0//topic/com.ibm.websphere.base.doc/info/>

[aes/ae/tprf_rqenable.html](http://publib.boulder.ibm.com/infocenter/wasinfo/v6r0//topic/com.ibm.websphere.base.doc/info/aes/ae/tprf_rqenable.html))  and its subtopics to enable request metrics and ARM. During enablement, use the following settings, which are recommended or required for use with EWLM:

- Select ALL in the **Components to be instrumented** list.
- Set the WebSphere Application Server request metrics **Trace level** to Hops.
- Set your **Agent Type** to ARM4 because this is the only WebSphere Application Server request metrics agent type supported for use with EWLM.
- Specify `com.ibm.wlm.arm40SDK.transaction.Arm40TransactionFactory` for the **Arm transaction factory implementation class name**.

7. Set the custom properties listed in the following table. To do this, use the instructions in Setting custom JVM properties ([http://publib.boulder.ibm.com/infocenter/wasinfo/v6r0/topic/](http://publib.boulder.ibm.com/infocenter/wasinfo/v6r0/topic/com.ibm.websphere.base.doc/info/aes/ae/xrun_jvm.html)

[com.ibm.websphere.base.doc/info/aes/ae/xrun_jvm.html](http://publib.boulder.ibm.com/infocenter/wasinfo/v6r0/topic/com.ibm.websphere.base.doc/info/aes/ae/xrun_jvm.html))  and its subtopics.

Note: “32-bit JNI library” and “64-bit JNI library” in the table below refer to the bit level of the JVM, not that of the operating system.

Table 56. Custom properties

Property	Value
java.library.path	For 32-bit JNI library, use: /usr/lib For 64-bit JNI library, use: /usr/lpp/ewlm/java/lib_64
ws.ext.dirs	/opt/IBM/VE2/EWLMMS/classes/ARM

Table 56. Custom properties (continued)

Property	Value
com.ibm.websphere.pmi.reqmetrics.PassCorrelatorToDB	Choose the proper setting based on your configuration: <ul style="list-style-type: none"> • If the next hop database is ARM instrumented, set this property to true. • If the next hop database is not ARM instrumented, either set this property to false or do not set it at all. This avoids any unnecessary overhead in the JDBC calls.


8. Restart WebSphere Application Server.
9. Verify that ARM is enabled by using the lsarm command.

Note: The lsarm command displays information about applications instrumented with ARM and registered with the operating system. For more information on using this command, see “lsarm” on page 327.

As part of its instrumentation, WebSphere Application Server has defined filters that you can use to classify transactions into service classes. WebSphere Application Server provides its instrumented filters to the domain manager in an XML document, so the EWLM Control Center is aware of which filters apply specifically to WebSphere Application Server. The WebSphere Application Server filters are preloaded into the EWLM Control Center; all you must do is set up the application in the EWLM Control Center and specify what filters you want to use. The filters you specify during the setup are available in the domain policy when you create transaction class rules. See “EWLM-supplied filter types” on page 35 for a complete list of filters available to all applications and “Application-specific filter types” on page 40 for additional filters for WebSphere Application Server.

Enabling ARM on WebSphere Application Server 6.0 on HP-UX:

Follow the instructions in this topic to enable ARM on WebSphere Application Server 6.0 on HP-UX.

Note: For more information about WebSphere Application Server and the administrative console, refer to the administrative console online help in the WebSphere Application Server 6.0 information center (<http://publib.boulder.ibm.com/infocenter/ws60help/>) .

1. Ensure that the EWLM managed server is configured and started, and that it is communicating properly with its EWLM domain manager.
2. Ensure that the middleware application is installed and configured. Use the specific product documentation if you need assistance.
3. Ensure that the user account under which the application will run is properly authorized to the ARM services. For more information, see “Authorizing HP-UX users to ARM services” on page 126.
4. Ensure that the instrumented application can locate the EWLM ARM libraries. To do this, set the `SHLIB_PATH` environment variable for the application instrumented with ARM in either the user’s login environment or in a script file that is used to launch the application.

For 32-bit applications instrumented with ARM, set the following environment variable:

```
export SHLIB_PATH="/opt/IBM/VE2/EWLM/usr/lib"
```

For 64-bit applications instrumented with ARM, set the following environment variable:

```
export SHLIB_PATH="/opt/IBM/VE2/EWLM/usr/lib/pa20_64"
```

Note: If you start the application from a Telnet session, you can update the .profile for the user to reflect the `SHLIB_PATH` setting as shown above. If you start the application from a script, the script should export the `SHLIB_PATH` environment variable before starting the application.


5. Start the WebSphere Application Server.


6. From a Web browser, go to and log in to the administrative console. Go to

`http://hostname:9060/ibm/console`


where *hostname* is the hostname or IP address for the node. The administrative console will load into the browser and you will be asked to login. Log in to the administrative console under the appropriate user account.


Note: The default port is 9060.

7. Enable request metrics. Use the instructions in Getting performance data from request metrics (http://publib.boulder.ibm.com/infocenter/wasinfo/v6r0/topic/com.ibm.websphere.base.doc/info/aes/ae/tprf_rqenable.html)  and its subtopics to enable request metrics and ARM. During enablement, use the following settings, which are recommended or required for use with EWLM:

info/aes/ae/tprf_rqenable.html)  and its subtopics to enable request metrics and ARM. During enablement, use the following settings, which are recommended or required for use with EWLM:

- Select ALL in the **Components to be instrumented** list.
- Set the WebSphere Application Server request metrics **Trace level** to Hops.
- Set your **Agent Type** to ARM4 because this is the only WebSphere Application Server request metrics agent type supported for use with EWLM.
- Specify `com.ibm.wlm.arm40SDK.transaction.Arm40TransactionFactory` for the **Arm transaction factory implementation class name**.

8. Set the custom properties listed in the following table. To do this, use the instructions in Setting custom JVM properties (http://publib.boulder.ibm.com/infocenter/wasinfo/v6r0/topic/com.ibm.websphere.base.doc/info/aes/ae/xrun_jvm.html)  and its subtopics.

com.ibm.websphere.base.doc/info/aes/ae/xrun_jvm.html)  and its subtopics.

Note: "32-bit JNI library" and "64-bit JNI library" in the table below refer to the bit level of the JVM, not that of the operating system.

Table 57. Custom properties

Property	Value
java.library.path	For 32-bit JNI library, use: /usr/lib For 64-bit JNI library, use: /usr/lib/pa20_64
ws.ext.dirs	/opt/IBM/VE2/EWLMS/classes/ARM
com.ibm.websphere.pmi.reqmetrics.PassCorrelatorToDB	Choose the proper setting based on your configuration: <ul style="list-style-type: none">• If the next hop database is ARM instrumented, set this property to true.• If the next hop database is not ARM instrumented, either set this property to false or do not set it at all. This avoids any unnecessary overhead in the JDBC calls.

9. Restart WebSphere Application Server.

10. Verify that ARM is enabled by using the `lsarm` command.

Note: The `lsarm` command displays information about applications instrumented with ARM and registered with the operating system. For more information on using this command, see "lsarm" on page 327.

As part of its instrumentation, WebSphere Application Server has defined filters that you can use to classify transactions into service classes. WebSphere Application Server provides its instrumented filters to the domain manager in an XML document, so the EWLM Control Center is aware of which filters

apply specifically to WebSphere Application Server. The WebSphere Application Server filters are preloaded into the EWLM Control Center; all you must do is set up the application in the EWLM Control Center and specify what filters you want to use. The filters you specify during the setup are available in the domain policy when you create transaction class rules. See “EWLM-supplied filter types” on page 35 for a complete list of filters available to all applications and “Application-specific filter types” on page 40 for additional filters for WebSphere Application Server.

Enabling ARM on WebSphere Application Server 6.0 on i5/OS:

Follow the instructions in this topic to enable ARM on WebSphere Application Server 6.0 on i5/OS.

Note: For more information about WebSphere Application Server and the administrative console, refer to the administrative console online help in the WebSphere Application Server 6.0 information center

(<http://publib.boulder.ibm.com/infocenter/ws60help/>) .

1. Ensure that the EWLM managed server is configured and started, and that it is communicating properly with its EWLM domain manager.
2. Ensure that the middleware application is installed and configured. Use the specific product documentation if you need assistance.
3. Ensure that the user account under which the application will run is properly authorized to the ARM services. For more information, see “Authorizing i5/OS users to ARM services” on page 138.
4. Start the WebSphere Application Server.
5. From a Web browser, go to and log in to the administrative console. Go to

`http://hostname:9060/ibm/console`

where *hostname* is the hostname or IP address for the node. The administrative console will load into the browser and you will be asked to login. Log in to the administrative console under the appropriate user account.

Note: The default port is 9060.



6. Enable request metrics. Use the instructions in Getting performance data from request metrics (http://publib.boulder.ibm.com/infocenter/wasinfo/v6r0//topic/com.ibm.websphere.base.doc/info/aes/ae/tprf_rqenable.html)  and its subtopics to enable request metrics and ARM. During enablement, use the following settings, which are recommended or required for use with EWLM:
 - Select ALL in the **Components to be instrumented** list.
 - Set the WebSphere Application Server request metrics **Trace level** to Hops.
 - Set your **Agent Type** to ARM4 because this is the only WebSphere Application Server request metrics agent type supported for use with EWLM.
 - Specify `com.ibm.wlm.arm40SDK.transaction.Arm40TransactionFactory` for the **Arm transaction factory implementation class name**.
7. Set the custom properties listed in the following table. To do this, use the instructions in Setting custom JVM properties (http://publib.boulder.ibm.com/infocenter/wasinfo/v6r0/topic/com.ibm.websphere.base.doc/info/aes/ae/xrun_jvm.html)  and its subtopics.
 - a. Expand **Servers > Application Servers > server1** (or the name of the appropriate server) > **Process Definition > Java Virtual Machine > Custom Properties**.
 - b. Add the following custom properties.

Table 58. Custom properties

Property	Value
java.library.path	/QSYS.LIB/QWLM2.LIB

Table 58. Custom properties (continued)

Property	Value
ws.ext.dirs	/QIBM/ProdData/VE2/EWLMMS/classes/ARM
com.ibm.websphere.pmi.reqmetrics.PassCorrelatorToDB	Choose the proper setting based on your configuration: <ul style="list-style-type: none"> • If the next hop database is ARM instrumented, set this property to true. • If the next hop database is not ARM instrumented, either set this property to false or do not set it at all. This avoids any unnecessary overhead in the JDBC calls.

- Restart WebSphere Application Server.
- Use the WRKARMJOB CL command to verify that ARM support on your system is enabled and that an application is currently calling ARM APIs.

As part of its instrumentation, WebSphere Application Server has defined filters that you can use to classify transactions into service classes. WebSphere Application Server provides its instrumented filters to the domain manager in an XML document, so the EWLM Control Center is aware of which filters apply specifically to WebSphere Application Server. The WebSphere Application Server filters are preloaded into the EWLM Control Center; all you must do is set up the application in the EWLM Control Center and specify what filters you want to use. The filters you specify during the setup are available in the domain policy when you create transaction class rules. See “EWLM-supplied filter types” on page 35 for a complete list of filters available to all applications and “Application-specific filter types” on page 40 for additional filters for WebSphere Application Server.

Enabling ARM on WebSphere Application Server 6.0 on Linux:

Follow the instructions in this topic to enable ARM on WebSphere Application Server 6.0 on Linux.

Note: For more information about WebSphere Application Server and the administrative console, refer to the administrative console online help in the WebSphere Application Server 6.0 information center


(<http://publib.boulder.ibm.com/infocenter/ws60help/>) .


- Ensure that the EWLM managed server is configured and started, and that it is communicating properly with its EWLM domain manager.
- Ensure that the middleware application is installed and configured. Use the specific product documentation if you need assistance.
- Ensure that the user account under which the application will run is properly authorized to the ARM services. For more information, see “Authorizing Linux users to ARM services” on page 148.
- Start the WebSphere Application Server.
- From a Web browser, go to and log in to the administrative console. Go to

`http://hostname:9060/ibm/console`

where *hostname* is the hostname or IP address for the node. The administrative console will load into the browser and you will be asked to login. Log in to the administrative console under the appropriate user account.

Note: The default port is 9060.

- Enable request metrics. Use the instructions in Getting performance data from request metrics (http://publib.boulder.ibm.com/infocenter/wasinfo/v6r0/topic/com.ibm.websphere.base.doc/info/aes/ae/tprf_rqenable.html)  and its subtopics to enable request metrics and ARM. During enablement, use the following settings, which are recommended or required for use with EWLM:

- Select ALL in the **Components to be instrumented** list.
 - Set the WebSphere Application Server request metrics **Trace level** to Hops.
 - Set your **Agent Type** to ARM4 because this is the only WebSphere Application Server request metrics agent type supported for use with EWLM.
 - Specify `com.ibm.wlm.arm40SDK.transaction.Arm40TransactionFactory` for the **Arm transaction factory implementation class name**.
7. Set the custom properties listed in the following table. To do this, use the instructions in Setting custom JVM properties (http://publib.boulder.ibm.com/infocenter/wasinfo/v6r0/topic/com.ibm.websphere.base.doc/info/aes/ae/xrun_jvm.html)  and its subtopics.

Note: "32-bit JNI library" and "64-bit JNI library" in the table below refer to the bit level of the JVM, not that of the operating system.

Table 59. Custom properties

Property	Value
java.library.path	For 32-bit JNI library, use: /usr/lib For 64-bit JNI library, use: /usr/lib64
ws.ext.dirs	/opt/IBM/VE2/EWLMMS/classes/ARM
com.ibm.websphere.pmi.reqmetrics.PassCorrelatorToDB	Choose the proper setting based on your configuration: <ul style="list-style-type: none"> • If the next hop database is ARM instrumented, set this property to true. • If the next hop database is not ARM instrumented, either set this property to false or do not set it at all. This avoids any unnecessary overhead in the JDBC calls.


- Restart WebSphere Application Server.
- Verify that ARM is enabled by using the `lsarm` command.

Note: The `lsarm` command displays information about applications instrumented with ARM and registered with the operating system. For more information on using this command, see "lsarm" on page 327.

As part of its instrumentation, WebSphere Application Server has defined filters that you can use to classify transactions into service classes. WebSphere Application Server provides its instrumented filters to the domain manager in an XML document, so the EWLM Control Center is aware of which filters apply specifically to WebSphere Application Server. The WebSphere Application Server filters are preloaded into the EWLM Control Center; all you must do is set up the application in the EWLM Control Center and specify what filters you want to use. The filters you specify during the setup are available in the domain policy when you create transaction class rules. See "EWLM-supplied filter types" on page 35 for a complete list of filters available to all applications and "Application-specific filter types" on page 40 for additional filters for WebSphere Application Server.

Enabling ARM on WebSphere Application Server 6.0 on Solaris:

Follow the instructions in this topic to enable ARM on WebSphere Application Server 6.0 on Solaris.



Note: For more information about WebSphere Application Server and the administrative console, refer to the administrative console online help in the WebSphere Application Server 6.0 information center (<http://publib.boulder.ibm.com/infocenter/ws60help/>)  .

1. Ensure that the EWLM managed server is configured and started, and that it is communicating properly with its EWLM domain manager.
2. Ensure that the middleware application is installed and configured. Use the specific product documentation if you need assistance.
3. Ensure that the user account under which the application will run is properly authorized to the ARM services. For more information, see “Authorizing Solaris users to ARM services” on page 156.
4. Start the WebSphere Application Server.
5. From a Web browser, go to and log in to the administrative console. Go to

`http://hostname:9060/ibm/console`

where *hostname* is the hostname or IP address for the node. The administrative console will load into the browser and you will be asked to login. Log in to the administrative console under the appropriate user account.

Note: The default port is 9060.

6. Enable request metrics. Use the instructions in Getting performance data from request metrics (http://publib.boulder.ibm.com/infocenter/wasinfo/v6r0//topic/com.ibm.websphere.base.doc/info/aes/ae/tprf_rqenable.html)  and its subtopics to enable request metrics and ARM. During enablement, use the following settings, which are recommended or required for use with EWLM:
 - Select ALL in the **Components to be instrumented** list.
 - Set the WebSphere Application Server request metrics **Trace level** to Hops.
 - Set your **Agent Type** to ARM4 because this is the only WebSphere Application Server request metrics agent type supported for use with EWLM.
 - Specify `com.ibm.wlm.arm40SDK.transaction.Arm40TransactionFactory` for the **Arm transaction factory implementation class name**.
7. Set the custom properties listed in the following table. To do this, use the instructions in Setting custom JVM properties (http://publib.boulder.ibm.com/infocenter/wasinfo/v6r0/topic/com.ibm.websphere.base.doc/info/aes/ae/xrun_jvm.html)  and its subtopics.

Note: “32-bit JNI library” and “64-bit JNI library” in the table below refer to the bit level of the JVM, not that of the operating system.

Table 60. Custom properties

Property	Value
java.library.path	For 32-bit JNI library, use: /usr/lib For 64-bit JNI library, use: /usr/lib/sparcv9
ws.ext.dirs	/opt/IBM/VE2/EWLMMS/classes/ARM

Table 60. Custom properties (continued)

Property	Value
com.ibm.websphere.pmi.reqmetrics.PassCorrelatorToDB	Choose the proper setting based on your configuration: <ul style="list-style-type: none"> • If the next hop database is ARM instrumented, set this property to true. • If the next hop database is not ARM instrumented, either set this property to false or do not set it at all. This avoids any unnecessary overhead in the JDBC calls.


- Restart WebSphere Application Server.
- Verify that ARM is enabled by using the lsarm command.

Note: The lsarm command displays information about applications instrumented with ARM and registered with the operating system. For more information on using this command, see “lsarm” on page 327.

As part of its instrumentation, WebSphere Application Server has defined filters that you can use to classify transactions into service classes. WebSphere Application Server provides its instrumented filters to the domain manager in an XML document, so the EWLM Control Center is aware of which filters apply specifically to WebSphere Application Server. The WebSphere Application Server filters are preloaded into the EWLM Control Center; all you must do is set up the application in the EWLM Control Center and specify what filters you want to use. The filters you specify during the setup are available in the domain policy when you create transaction class rules. See “EWLM-supplied filter types” on page 35 for a complete list of filters available to all applications and “Application-specific filter types” on page 40 for additional filters for WebSphere Application Server.

Enabling ARM on WebSphere Application Server 6.0 on Windows:

Follow the instructions in this topic to enable ARM on WebSphere Application Server 6.0 on Windows.

Note: For more information about WebSphere Application Server and the administrative console, refer to the administrative console online help in the WebSphere Application Server 6.0 information center (<http://publib.boulder.ibm.com/infocenter/ws60help/>) .


- Ensure that the EWLM managed server is configured and started, and that it is communicating properly with its EWLM domain manager.
- Ensure that the middleware application is installed and configured. Use the specific product documentation if you need assistance.
- Ensure that the user account under which the application will run is properly authorized to the ARM services. For more information, see “Authorizing Windows users to ARM services” on page 166.
- Start the WebSphere Application Server.
- From a Web browser, go to and log in to the administrative console. Go to

`http://hostname:9060/ibm/console`


where *hostname* is the hostname or IP address for the node. The administrative console will load into the browser and you will be asked to login. Log in to the administrative console under the appropriate user account.

Note: The default port is 9060.

- Enable request metrics. Use the instructions in Getting performance data from request metrics (<http://publib.boulder.ibm.com/infocenter/wasinfo/v6r0//topic/com.ibm.websphere.base.doc/info/>

aes/ae/tprf_rqenable.html)  and its subtopics to enable request metrics and ARM. During enablement, use the following settings, which are recommended or required for use with EWLM:

- Select ALL in the **Components to be instrumented** list.
- Set the WebSphere Application Server request metrics **Trace level** to Hops.
- Set your **Agent Type** to ARM4 because this is the only WebSphere Application Server request metrics agent type supported for use with EWLM.
- Specify `com.ibm.wlm.arm40SDK.transaction.Arm40TransactionFactory` for the **Arm transaction factory implementation class name**.

7. Set the custom properties listed in the following table. To do this, use the instructions in Setting custom JVM properties (http://publib.boulder.ibm.com/infocenter/wasinfo/v6r0/topic/com.ibm.websphere.base.doc/info/aes/ae/xrun_jvm.html)  and its subtopics.

Note: "32-bit JNI library" and "64-bit JNI library" in the table below refer to the bit level of the JVM, not that of the operating system.

Table 61. Custom properties

Property	Value
java.library.path	For 32-bit JNI library, use: C:\Program Files\IBM\VE2\EWLMMS\classes\ms
ws.ext.dirs	C:\Program Files\IBM\VE2\EWLMMS\classes\ARM
com.ibm.websphere.pmi.reqmetrics.PassCorrelatorToDB	Choose the proper setting based on your configuration: <ul style="list-style-type: none"> • If the next hop database is ARM instrumented, set this property to true. • If the next hop database is not ARM instrumented, either set this property to false or do not set it at all. This avoids any unnecessary overhead in the JDBC calls.

8. Restart WebSphere Application Server.
9. Verify that ARM is enabled by using the `ewlmWinADTool.exe` program.

Note: `ewlmWinADTool` displays information about applications instrumented with ARM and registered with the operating system. For more information on using this command, see "ewlmWinADTool.exe" on page 328.

As part of its instrumentation, WebSphere Application Server has defined filters that you can use to classify transactions into service classes. WebSphere Application Server provides its instrumented filters to the domain manager in an XML document, so the EWLM Control Center is aware of which filters apply specifically to WebSphere Application Server. The WebSphere Application Server filters are preloaded into the EWLM Control Center; all you must do is set up the application in the EWLM Control Center and specify what filters you want to use. The filters you specify during the setup are available in the domain policy when you create transaction class rules. See "EWLM-supplied filter types" on page 35 for a complete list of filters available to all applications and "Application-specific filter types" on page 40 for additional filters for WebSphere Application Server.

Enabling ARM on WebSphere Application Server 6.0 on z/OS:

Follow the instructions in this topic to enable ARM on WebSphere Application Server 6.0 on z/OS.

Note: For more information about WebSphere Application Server and the administrative console, refer to the administrative console online help in the WebSphere Application Server 6.0 information center (<http://publib.boulder.ibm.com/infocenter/ws60help/>) .

1. Ensure that the EWLM managed server is configured and started, and that it is communicating properly with its EWLM domain manager.
2. Ensure that the middleware application is installed and configured. Use the specific product documentation if you need assistance.
3. Ensure that the user account under which the application will run is properly authorized to the ARM services. For more information, see “Authorizing z/OS users to ARM services” on page 188.
4. Start the WebSphere Application Server.
5. From a Web browser, go to and log in to the administrative console. Go to

`http://hostname:9060/ibm/console`

where *hostname* is the hostname or IP address for the node. The administrative console will load into the browser and you will be asked to login. Log in to the administrative console under the appropriate user account.

Note: The default port is 9060.



6. Enable request metrics. Use the instructions in Getting performance data from request metrics (http://publib.boulder.ibm.com/infocenter/wasinfo/v6r0//topic/com.ibm.websphere.zseries.doc/info/zseries/ae/tprf_rqenable.html)  and its subtopics to enable request metrics and ARM. During enablement, use the following settings, which are recommended or required for use with EWLM:
 - Select ALL in the **Components to be instrumented** list.
 - Set the WebSphere Application Server request metrics **Trace level** to Hops.
 - Set your **Agent Type** to ARM4 because this is the only WebSphere Application Server request metrics agent type supported for use with EWLM.
 - Specify `com.ibm.wlm.arm40SDK.transaction.Arm40TransactionFactory` for the **Arm transaction factory implementation class name**.
7. Set the custom properties.
 - a. Follow the instructions in Setting custom JVM properties (http://publib.boulder.ibm.com/infocenter/wasinfo/v6r0//topic/com.ibm.websphere.zseries.doc/info/zseries/ae/xrun_jvm.html)  and its subtopics to set the following custom properties.

Table 62. Custom properties

Property	Value
<code>com.ibm.websphere.pmi.reqmetrics.PassCorrelatorToDB</code>	Choose the proper setting based on your configuration: <ul style="list-style-type: none"> • If the next hop database is ARM instrumented, set this property to true. • If the next hop database is not ARM instrumented, either set this property to false or do not set it at all. This avoids any unnecessary overhead in the JDBC calls.
<code>ws.ext.dirs</code>	<code>/usr/lpp/VE_R2/EWLMMS/classes/ARM</code>

- b. Set the *WAS SERVER ONLY server region libpath* variable:
 - 1) Expand **Environments > WebSphere Variables**.
 - 2) Append the *WAS SERVER ONLY server region libpath* variable to include the following text at the end of the value:


```
:/usr/lpp/VE_R2/EWLMMS/lib:/usr/lib
```

8. Restart WebSphere Application Server.

- | 9. Verify that ARM is enabled. For more information on how to do this, see “Verifying the status of ARM services on z/OS” on page 189.

As part of its instrumentation, WebSphere Application Server has defined filters that you can use to classify transactions into service classes. WebSphere Application Server provides its instrumented filters to the domain manager in an XML document, so the EWLM Control Center is aware of which filters apply specifically to WebSphere Application Server. The WebSphere Application Server filters are preloaded into the EWLM Control Center; all you must do is set up the application in the EWLM Control Center and specify what filters you want to use. The filters you specify during the setup are available in the domain policy when you create transaction class rules. See “EWLM-supplied filter types” on page 35 for a complete list of filters available to all applications and “Application-specific filter types” on page 40 for additional filters for WebSphere Application Server.

Enabling ARM on the EWLM plug-in for IIS 5.0 or 6.0

Use this topic to enable ARM on the EWLM plug-in for IIS 5.0 or 6.0.

The specific tasks you must complete to enable ARM vary depending on the target operating system. Select from the list below a topic collection that is specific to the operating system on which you want to enable ARM on the EWLM plug-in for IIS 5.0 or 6.0.

Enabling ARM on EWLM plug-in for IIS 5.0 or 6.0 on Windows:

Follow the instructions in this topic to enable ARM on the EWLM plug-in for IIS 5.0 or 6.0. on Windows.

If you currently use Microsoft Internet Information Services (IIS) 5.0 or 6.0 on any managed server in your EWLM domain, you can use the independent plug-in for IIS included with EWLM to monitor static page serving transactions flowing through the Web server. Do not enable this plug-in for IIS Web servers that handle requests other than static page serving.

The EWLM IIS plug-in is actually an Internet Services Application Programming Interface (ISAPI) filter, which is a DLL file that, in this case, is instrumented to provide ARM services to IIS. ISAPI filters are used only on IIS servers. You can either enable the filter for all Web sites (global filter) or you can enable it for individual Web sites. You must be a member of the **Administrators** group on the local server to install the plug-in, or you must have been delegated the appropriate authority. For more information refer to the Microsoft IIS Server version 5.0 and 6.0 documentation available at <http://www.microsoft.com>.

- | If you wish to enable this plug-in for multiple applications, see the White Paper “Integration of Enterprise Workload Manager (EWLM) with Microsoft Internet Information Services (IIS) Plug-in Through Application Response Measurement (ARM) Instrumentation” (WP100601). It describes how a second tier application acquires the correlator from a first tier application, including information about the coexistence of plug-ins and how different tiers work together.

Follow these steps to install the EWLM IIS plug-in. Instructions for uninstalling the plug-in are also described in this topic.

1. Copy `EWLMIISFilter.dll` to an appropriate folder, such as the `SCRIPTS` or `CGI-BIN` subdirectory. You can also copy the file to your own directory, such as `C:\IISplug-inTST`. The EWLM IIS plug-in, `EWLMIISFilter.dll`, is located in the `%ProgramFiles%\IBM\VE2\EWLMMS\classes\ms\IISFilter` directory by default.
2. Open **IIS Manager** by completing the following steps:
 - a. Click **Start > Run**.
 - b. Type `mmc %systemroot%\system32\inetmgr\iis.msc` and click **OK**.
3. Expand the local computer and right-click the Web server or Web site to which you want to add the plug-in:
 - a. Select the Web server or Server Name to enable the ISAPI filter for all Web sites.

- b. Select the specific Web site to enable the ISAPI filter for a particular Web site, for example, the default Web site.

and then select **Properties**.
4. Go to the **ISAPI Filters** page, and click **Add**.
5. Enter the name of the ISAPI filter. If you copied it to your own directory, click **Browse** and select the EWLM IIS filter, `EWLMISFilter.dll`.
6. Click **OK**.
7. Stop the **IISADMIN** service by performing either of the following steps:
 - a. From a command prompt, enter `net stop iisadmin /y`.
 - b. From the Control Panel, open **Administrative Tools > Services**. Right-click **IISADMIN** and select **Stop**.
8. Restart the **World Wide Web Publishing** service by performing either of the following steps:
 - a. From a command prompt, enter `net start w3svc`.
 - b. From the Control Panel, open **Administrative Tools > Services**. Right-click **World Wide Web Publishing** and select **Restart**.
9. Verify that the filter has been loaded properly.
 - a. Go back to the **ISAPI Filters** page by repeating steps 2 through 4.
 - b. Look for the EWLM IIS filter in the list of ISAPI filters. A green arrow in the Status column indicates that the filter has been loaded and is active.
10. To enable authentication for IIS 6.0, complete the following steps:
 - a. Add **Service** to the **eWLMArm4Users** group by selecting **Start > Control Panel > Administrative Tools > Computer Management**.
 - b. In the **Computer Management** window, go to **System Tools > Local Users and Groups > Groups**.
 - c. Right-click **eWLMArm4Users** and select **Properties**.
 - d. On the Properties page, click **Add**.
 - e. Click **Object types**.
 - f. Ensure that Built-in security principals and Users are selected in the **Object types** field.
 - g. Click **OK** to close the Object Types page and return to the Select Users page.
 - h. Click **Locations**.
 - i. Ensure that the system with IIS Manager is specified.
 - j. Click **OK** to close the Location page and return to the Select Users page.
 - k. On the Select Users page, click **Advanced**.
 - l. Click **Find Now**.
 - m. Select **SERVICE** so that it is highlighted. If **SERVICE** does not appear in the list, try specifying **SERVICE** in the **Name** field and performing another search.
 - n. Click **OK** to add **SERVICE** to the **eWLMArm4Users** group.
 - o. Click **OK** to close the properties page.
11. Right click the Application Pool folder and click **New** to create a new application pool from the IIS Manager.
12. Select Application Pool.
13. Specify a **name** and click **OK**.
14. Right click the new Application Pool and select **Properties**.
15. Select the **Identity** page.
16. Select **Configurable** and specify the user name and password to use. IBM recommends using the built-in account for Internet Information Services, which is a user name of (`IWAM_%COMPUTER_NAME%`) and its corresponding password.

17. Click **OK**.
18. Select the Web sites from the directory to specify access control:
 - a. Right click on the EWLM plug-in IIS Web site.
 - b. Click **Properties**.
 - c. On the **Directory Security** page. Update the information as necessary for Authentication and access control.
19. Enable anonymous access and enter the built-in account for anonymous access (IUSR_%COMPUTER_NAME%) and click **OK**.
20. On the **Home Directory** page, select the Application pool that you created and click **OK**.
21. If necessary, uninstall the EWLM IIS plug-in:
 - a. Open **IIS Manager**, expand the local computer, right-click the Web server or Web site from which you want to remove the plug-in:
 - Select the Web server or Server Name to remove the ISAPI filter for all Web sites and select **Properties**.
 - Select the specific Web site to remove the ISAPI filter for a particular Web site, for example, the default Web site and select **Properties**.
 - b. Go to the **ISAPI Filters** page, select the EWLM IIS filter and click **Remove**.
 - c. Click **Apply**.
 - d. Stop and Restart IIS and the Web site or Web server where the filter was registered.

As part of its instrumentation, the EWLM plug-in for IIS has defined filters that you can use to classify transactions into service classes. The EWLM plug-in for IIS provides its instrumented filters to the domain manager in an XML document, so the EWLM Control Center is aware of which filters apply specifically to EWLM plug-in for IIS. The EWLM plug-in for IIS filters are preloaded into the EWLM Control Center; all you must do is set up the application in the EWLM Control Center and specify what filters you want to use. The filters you specify during the set up are available in the domain policy when you create transaction class rules. See “EWLM-supplied filter types” on page 35 for a complete list of filters available to all applications and “Application-specific filter types” on page 40 for additional filters for Web serving plug-ins.

Enabling ARM on HTTP Server

Use this topic to select the type of HTTP plug-in you are using.

IBM HTTP Server has two types of Web serving plug-ins that are instrumented with ARM:

- WebSphere HTTP plug-ins that are included and installed with WebSphere Application Server
- Independent HTTP plug-ins that are used by non-WebSphere Application Server applications

EWLM supports ARM instrumentation for the following versions of IBM HTTP Server:

- IBM HTTP Server 1.3.2x (compatible with WebSphere Application Server HTTP plug-ins)
- IBM HTTP Server 2.0.47 or 6.0.x (compatible with WebSphere Application Server HTTP plug-ins and independent plug-ins)
- IBM HTTP Server for iSeries (powered by Apache) V5R3

How you enable ARM on IBM HTTP Server depends on which version of HTTP Server you have running, as well as which type of plug-in you are using.

Select from below the type of HTTP plug-in you are using.

Enabling ARM on the HTTP Server with a WebSphere Application Server HTTP plug-in:

Use this topic to enable ARM on the HTTP Server with a WebSphere Application Server HTTP plug-in.




The specific tasks you must complete to enable ARM vary depending on the target operating system. Select from the list below a topic collection that is specific to the operating system on which you want to enable ARM on the HTTP Server with a WebSphere Application Server HTTP plug-in.

Enabling ARM on WebSphere Application Server HTTP plug-in on AIX:

Follow the instructions in this topic to enable ARM on WebSphere Application Server HTTP plug-in on AIX.

- | These instructions apply to HTTP Server 1.3.2x, HTTP Server 2.0.4x, and HTTP Server 6.0.x, as well as Apache server 1.3.2x and 2.0.4x.

Note:

- |
 - Throughout this topic, *WAS_PLUGIN_HOME* refers to the path to which the WebSphere Application Server plugin is installed. For WebSphere Application Server v5.1, this is the same as *WAS_HOME*. See [Manually configuring IBM HTTP Server powered by Apache 2.0](#)  or [Configuring IBM HTTP Server Version](#)  for further information.
 - For more information about IBM HTTP Server and working with Web serving plug-ins, refer to the [HTTP Server Information Center](#) .
- 1. Ensure that the EWLM managed server is configured and started, and that it is communicating properly with its EWLM domain manager.
- 2. Ensure that the middleware application is installed and configured. Use the specific product documentation if you need assistance.
- 3. Ensure that the user account under which the application will run is properly authorized to the ARM services. For more information, see “Authorizing AIX users to ARM services” on page 120.
- 4. Edit the HTTP Server configuration file, `httpd.conf`, to load the plug-in:

HTTP Server 1.3.2x

Note: You can skip this step for HTTP Server 1.3.28, because this configuration is done automatically when you install WebSphere Application Server.

Add the following lines to the end of *IHS_HOME/conf/httpd.conf*:

```
| LoadModule ibm_app_server_http_module WAS_PLUGIN_HOME/bin/mod_ibm_app_server_http.so
| WebSpherePluginConfig WAS_PLUGIN_HOME/config/cells/plugin-cfg.xml
```

HTTP Server 2.0.47 or 6.0.x

Add the following lines to the end of *IHS2_HOME/conf/httpd.conf*:

```
| LoadModule was_ap20_module WAS_PLUGIN_HOME/bin/mod_was_ap20_http.so
| WebSpherePluginConfig WAS_PLUGIN_HOME/config/cells/plugin-cfg.xml
```

- | 5. Optional. Regenerate the WebSphere Application Server plugin configuration file
- | 6. Open the plug-in configuration file found at *WAS_PLUGIN_HOME/config/cells/plugin-cfg.xml* and look for this line:

```
| <RequestMetrics armEnabled="true" loggingEnabled="false" rmEnabled="true" traceLevel="HOPS">
```

Edit the file if necessary.

- | 7. Ensure that the user who starts the application is authorized to ARM services. See “Authorizing AIX non-root users to start EWLM on an AIX managed server” on page 118 for instructions.
- | 8. Restart the HTTP Server.
- | 9. Verify that ARM is enabled by using the `lsarm` command.

Note: The `lsarm` command displays information about applications instrumented with ARM and registered with the operating system. For more information on using this command, see “`lsarm`” on page 327.

Each of the HTTP Server plug-ins have defined filters that you can use to classify transactions into service classes for your domain policy. These filters are preloaded into the EWLM Control Center; all you must do is set up the appropriate application in the EWLM Control Center and specify what filters you want to use. The filters you specify during the setup are available in the domain policy when you create your transaction class rules for the application. See “EWLM-supplied filter types” on page 35 for a complete list of filters available to all applications and “Application-specific filter types” on page 40 for additional filters for Webserving plug-ins which includes the HTTP server.

Enabling ARM on WebSphere Application Server HTTP plug-in on HP-UX:

Follow the instructions in this topic to enable ARM on WebSphere Application Server HTTP plug-in on HP-UX.

These instructions apply to HTTP Server 1.3.2x, HTTP Server 2.0.4x, and HTTP Server 6.0.x, as well as Apache server 1.3.2x and 2.0.4x.

Note:

- Throughout this topic, `WAS_PLUGIN_HOME` refers to the path to which the WebSphere Application Server plugin is installed. For WebSphere Application Server v5.1, this is the same as `WAS_HOME`. See [Manually configuring IBM HTTP Server powered by Apache 2.0](#) or [Configuring IBM HTTP Server Version](#) for further information.
- For more information about IBM HTTP Server and working with Web serving plug-ins, refer to the [HTTP Server Information Center](#).

1. Ensure that the EWLM managed server is configured and started, and that it is communicating properly with its EWLM domain manager.
2. Ensure that the middleware application is installed and configured. Use the specific product documentation if you need assistance.
3. Ensure that the user account under which the application will run is properly authorized to the ARM services. For more information, see “Authorizing HP-UX users to ARM services” on page 126.
4. Ensure that the instrumented application can locate the EWLM ARM libraries. To do this, set the `SHLIB_PATH` environment variable for the application instrumented with ARM in either the user’s login environment or in a script file that is used to launch the application.

For 32-bit applications instrumented with ARM, set the following environment variable:

```
export SHLIB_PATH="/opt/IBM/VE2/EWLM/usr/lib"
```

For 64-bit applications instrumented with ARM, set the following environment variable:

```
export SHLIB_PATH="/opt/IBM/VE2/EWLM/usr/lib/pa20_64"
```

Note: If you start the application from a Telnet session, you can update the `.profile` for the user to reflect the `SHLIB_PATH` setting as shown above. If you start the application from a script, the script should export the `SHLIB_PATH` environment variable before starting the application.

5. Edit the HTTP Server configuration file, `httpd.conf`, to load the plug-in:

HTTP Server 1.3.2x

Note: You can skip this step for HTTP Server 1.3.28, because this configuration is done automatically when you install WebSphere Application Server.
Add the following lines to the end of `IHS_HOME/conf/httpd.conf`:

```
| LoadModule ibm_app_server_http_module WAS_PLUGIN_HOME/bin/mod_ibm_app_server_http.s1  
| WebSpherePluginConfig WAS_PLUGIN_HOME/config/cells/plugin-cfg.xml
```

HTTP Server 2.0.47 or 6.0.x

Add the following lines to the end of `IHS2_HOME/conf/httpd.conf`:

```
| LoadModule was_ap20_module WAS_PLUGIN_HOME/bin/mod_was_ap20_http.s1  
| WebSpherePluginConfig WAS_PLUGIN_HOME/config/cells/plugin-cfg.xml
```

6. Optional. Regenerate the WebSphere Application Server plugin configuration file
7. Open the plug-in configuration file found at `WAS_PLUGIN_HOME/config/cells/plugin-cfg.xml` and look for this line:

```
| <RequestMetrics armEnabled="true" loggingEnabled="false" rmEnabled="true" traceLevel="HOPS">
```

Edit the file if necessary.

8. Ensure that the user who starts the application is authorized to ARM services. See “Authorizing HP-UX users to ARM services” on page 126 for instructions.
9. Restart the HTTP Server.
10. Verify that ARM is enabled by using the `lsarm` command.

Note: The `lsarm` command displays information about applications instrumented with ARM and registered with the operating system. For more information on using this command, see “`lsarm`” on page 327.




Each of the HTTP Server plug-ins have defined filters that you can use to classify transactions into service classes for your domain policy. These filters are preloaded into the EWLM Control Center; all you must do is set up the appropriate application in the EWLM Control Center and specify what filters you want to use. The filters you specify during the setup are available in the domain policy when you create your transaction class rules for the application. See “EWLM-supplied filter types” on page 35 for a complete list of filters available to all applications and “Application-specific filter types” on page 40 for additional filters for Webserving plug-ins which includes the HTTP server.

Enabling ARM on WebSphere Application Server HTTP plug-in on Linux:

Follow the instructions in this topic to enable ARM on WebSphere Application Server HTTP plug-in on Linux.

These instructions apply to HTTP Server 1.3.2x, HTTP Server 2.0.4x, and HTTP Server 6.0.x, as well as Apache server 1.3.2x and 2.0.4x.

Note:

- Throughout this topic, `WAS_PLUGIN_HOME` refers to the path to which the WebSphere Application Server plugin is installed. For WebSphere Application Server v5.1, this is the same as `WAS_HOME`. See [Manually configuring IBM HTTP Server powered by Apache 2.0](#)  or [Configuring IBM HTTP Server Version](#)  for further information.
- For more information about IBM HTTP Server and working with Web serving plug-ins, refer to the [HTTP Server Information Center](#) .

1. Ensure that the EWLM managed server is configured and started, and that it is communicating properly with its EWLM domain manager.
2. Ensure that the middleware application is installed and configured. Use the specific product documentation if you need assistance.
3. Ensure that the user account under which the application will run is properly authorized to the ARM services. For more information, see “Authorizing Linux users to ARM services” on page 148.
4. Edit the HTTP Server configuration file, `httpd.conf`, to load the plug-in:

HTTP Server 1.3.2x

Note: You can skip this step for HTTP Server 1.3.28, because this configuration is done automatically when you install WebSphere Application Server.

Add the following lines to the end of `IHS_HOME/conf/httpd.conf`:

```
LoadModule ibm_app_server_http_module WAS_PLUGIN_HOME/bin/mod_ibm_app_server_http.so
WebSpherePluginConfig WAS_PLUGIN_HOME/config/cells/plugin-cfg.xml
```

HTTP Server 2.0.47 or 6.0.x

Add the following lines to the end of `IHS2_HOME/conf/httpd.conf`:

```
LoadModule was_ap20_module WAS_PLUGIN_HOME/bin/mod_was_ap20_http.so
WebSpherePluginConfig WAS_PLUGIN_HOME/config/cells/plugin-cfg.xml
```

- Optional. Regenerate the WebSphere Application Server plugin configuration file
- Open the plug-in configuration file found at `WAS_PLUGIN_HOME/config/cells/plugin-cfg.xml` and look for this line:

```
<RequestMetrics armEnabled="true" loggingEnabled="false" rmEnabled="true" traceLevel="HOPS">
```

Edit the file if necessary.

- Ensure that the user who starts the application is authorized to ARM services. See “Authorizing Linux users to ARM services” on page 148 for instructions.
- Restart the HTTP Server.
- Verify that ARM is enabled by using the `lsarm` command.

Note: The `lsarm` command displays information about applications instrumented with ARM and registered with the operating system. For more information on using this command, see “`lsarm`” on page 327.

Each of the HTTP Server plug-ins have defined filters that you can use to classify transactions into service classes for your domain policy. These filters are preloaded into the EWLM Control Center; all you must do is set up the appropriate application in the EWLM Control Center and specify what filters you want to use. The filters you specify during the setup are available in the domain policy when you create your transaction class rules for the application. See “EWLM-supplied filter types” on page 35 for a complete list of filters available to all applications and “Application-specific filter types” on page 40 for additional filters for Webserving plug-ins which includes the HTTP server.

Enabling ARM on WebSphere Application Server HTTP plug-in on Solaris:

Follow the instructions in this topic to enable ARM on WebSphere Application Server HTTP plug-in on Solaris.

These instructions apply to HTTP Server 1.3.2x, HTTP Server 2.0.4x, and HTTP Server 6.0.x, as well as Apache server 1.3.2x and 2.0.4x.

Note:

- Throughout this topic, `WAS_PLUGIN_HOME` refers to the path to which the WebSphere Application Server plugin is installed. For WebSphere Application Server v5.1, this is the same as `WAS_HOME`. See [Manually configuring IBM HTTP Server powered by Apache 2.0](#) or [Configuring IBM HTTP Server Version](#) for further information.
- For more information about IBM HTTP Server and working with Web serving plug-ins, refer to the [HTTP Server Information Center](#).

- Ensure that the EWLM managed server is configured and started, and that it is communicating properly with its EWLM domain manager.

2. Ensure that the middleware application is installed and configured. Use the specific product documentation if you need assistance.
3. Ensure that the user account under which the application will run is properly authorized to the ARM services. For more information, see “Authorizing Solaris users to ARM services” on page 156.
4. Edit the HTTP Server configuration file, `httpd.conf`, to load the plug-in:

HTTP Server 1.3.2x

Note: You can skip this step for HTTP Server 1.3.28, because this configuration is done automatically when you install WebSphere Application Server.

Add the following lines to the end of `IHS_HOME/conf/httpd.conf`:

```
| LoadModule ibm_app_server_http_module WAS_PLUGIN_HOME/bin/mod_ibm_app_server_http.so
| WebSpherePluginConfig WAS_PLUGIN_HOME/config/cells/plugin-cfg.xml
```

HTTP Server 2.0.47 or 6.0.x

Add the following lines to the end of `IHS2_HOME/conf/httpd.conf`:

```
| LoadModule was_ap20_module WAS_PLUGIN_HOME/bin/mod_was_ap20_http.so
| WebSpherePluginConfig WAS_PLUGIN_HOME/config/cells/plugin-cfg.xml
```

5. Optional. Regenerate the WebSphere Application Server plugin configuration file
6. Open the plug-in configuration file found at `WAS_PLUGIN_HOME/config/cells/plugin-cfg.xml` and look for this line:

```
| <RequestMetrics armEnabled="true" loggingEnabled="false" rmEnabled="true" traceLevel="HOPS">
```

Edit the file if necessary.

7. Ensure that the user who starts the application is authorized to ARM services. See “Authorizing Solaris users to ARM services” on page 156 for instructions.
8. Restart the HTTP Server.
9. Verify that ARM is enabled by using the `lsarm` command.

Note: The `lsarm` command displays information about applications instrumented with ARM and registered with the operating system. For more information on using this command, see “`lsarm`” on page 327.



Each of the HTTP Server plug-ins have defined filters that you can use to classify transactions into service classes for your domain policy. These filters are preloaded into the EWLM Control Center; all you must do is set up the appropriate application in the EWLM Control Center and specify what filters you want to use. The filters you specify during the setup are available in the domain policy when you create your transaction class rules for the application. See “EWLM-supplied filter types” on page 35 for a complete list of filters available to all applications and “Application-specific filter types” on page 40 for additional filters for Webserving plug-ins which includes the HTTP server.


Enabling ARM on WebSphere Application Server HTTP plug-in on Windows:

Follow the instructions in this topic to enable ARM on WebSphere Application Server HTTP plug-in on Windows.

- These instructions apply to HTTP Server 1.3.2x, HTTP Server 2.0.4x, and HTTP Server 6.0.x, as well as Apache server 1.3.2x and 2.0.4x.

Note:

- Throughout this topic, `WAS_PLUGIN_HOME` refers to the path to which the WebSphere Application Server plugin is installed. For WebSphere Application Server v5.1, this is the same as `WAS_HOME`. See [Manually configuring IBM HTTP Server powered by Apache 2.0](#)  or [Configuring IBM HTTP Server Version](#)  for further information.

- For more information about IBM HTTP Server and working with Web serving plug-ins, refer to the HTTP Server Information Center  .

1. Ensure that the EWLM managed server is configured and started, and that it is communicating properly with its EWLM domain manager.
2. Ensure that the middleware application is installed and configured. Use the specific product documentation if you need assistance.
3. Ensure that the user account under which the application will run is properly authorized to the ARM services. For more information, see “Authorizing Windows users to ARM services” on page 166.
4. Edit the HTTP Server configuration file, `httpd.conf`, to load the plug-in:

HTTP Server 1.3.2x

Note: You can skip this step for HTTP Server 1.3.28, because this configuration is done automatically when you install WebSphere Application Server.

Add the following lines to the end of `IHS_HOME\conf\httpd.conf`:

```
| LoadModule ibm_app_server_http_module WAS_PLUGIN_HOME\bin\mod_ibm_app_server_http.dll
| WebSpherePluginConfig WAS_PLUGIN_HOME\config\cells\plug-in-cfg.xml
```

HTTP Server 2.0.47 or 6.0.x

Add the following lines to the end of `IHS2_HOME\conf\httpd.conf`:

```
| LoadModule was_ap20_module WAS_PLUGIN_HOME\bin\mod_was_ap20_http.dll
| WebSpherePluginConfig WAS_PLUGIN_HOME\config\cells\plug-in-cfg.xml
```

5. Optional. Regenerate the WebSphere Application Server plugin configuration file
6. Open the plug-in configuration file found at `WAS_PLUGIN_HOME\config\cells\plug-in-cfg.xml` and look for this line:

```
| <RequestMetrics armEnabled="true" loggingEnabled="false" rmEnabled="true" traceLevel="HOPS">
```

Edit the file if necessary.

7. Ensure that the user who starts the application is authorized to ARM services. See “Authorizing Windows users to ARM services” on page 166 for instructions.
8. Restart the HTTP Server.
9. Verify that ARM is enabled by using the `ewlmWinADTool.exe` program.

Note: `ewlmWinADTool` displays information about applications instrumented with ARM and registered with the operating system. For more information on using this command, see “`ewlmWinADTool.exe`” on page 328.

Each of the HTTP Server plug-ins have defined filters that you can use to classify transactions into service classes for your domain policy. These filters are preloaded into the EWLM Control Center; all you must do is set up the appropriate application in the EWLM Control Center and specify what filters you want to use. The filters you specify during the setup are available in the domain policy when you create your transaction class rules for the application. See “EWLM-supplied filter types” on page 35 for a complete list of filters available to all applications and “Application-specific filter types” on page 40 for additional filters for Webserving plug-ins which includes the HTTP server.

Enabling ARM on WebSphere Application Server HTTP plug-in on z/OS:

Follow the instructions in this topic to enable ARM on WebSphere Application Server HTTP plug-in on z/OS.

- These instructions apply to HTTP Server 1.3.2x, HTTP Server 2.0.4x, and HTTP Server 6.0.x, as well as Apache server 1.3.2x and 2.0.4x.

Note:

- Throughout this topic, *WAS_PLUGIN_HOME* refers to the path to which the WebSphere Application Server plugin is installed. For WebSphere Application Server v5.1, this is the same as *WAS_HOME*. See [Manually configuring IBM HTTP Server powered by Apache 2.0](#) or [Configuring IBM HTTP Server Version](#) for further information.
- For more information about IBM HTTP Server and working with Web serving plug-ins, refer to the [HTTP Server Information Center](#).

1. Ensure that the EWLM managed server is configured and started, and that it is communicating properly with its EWLM domain manager.
2. Ensure that the middleware application is installed and configured. Use the specific product documentation if you need assistance.
3. Ensure that the user account under which the application will run is properly authorized to the ARM services. For more information, see “Authorizing z/OS users to ARM services” on page 188.
4. Edit the HTTP Server configuration file, `httpd.conf`, to load the plug-in:

HTTP Server 1.3.2x

Note: You can skip this step for HTTP Server 1.3.28, because this configuration is done automatically when you install WebSphere Application Server.

Add the following lines to the end of *IHS_HOME/conf/httpd.conf*:

```
LoadModule ibm_app_server_http_module WAS_PLUGIN_HOME/bin/mod_ibm_app_server_http.so
WebSpherePluginConfig WAS_PLUGIN_HOME/config/cells/plugin-cfg.xml
```

HTTP Server 2.0.47 or 6.0.x

Add the following lines to the end of *IHS2_HOME/conf/httpd.conf*:

```
LoadModule was_ap20_module WAS_PLUGIN_HOME/bin/mod_was_ap20_http.so
WebSpherePluginConfig WAS_PLUGIN_HOME/config/cells/plugin-cfg.xml
```

5. Optional. Regenerate the WebSphere Application Server plugin configuration file
6. Open the plug-in configuration file found at *WAS_PLUGIN_HOME/config/cells/plugin-cfg.xml* and look for this line:

```
<RequestMetrics armEnabled="true" loggingEnabled="false" rmEnabled="true" traceLevel="HOPS">
```

Edit the file if necessary.


7. Ensure that the user who starts the application is authorized to ARM services. See “Authorizing z/OS users to ARM services” on page 188 for instructions.
8. Restart the HTTP Server.
9. Verify that ARM is enabled. For more information on how to do this, see “Verifying the status of ARM services on z/OS” on page 189.

Each of the HTTP Server plug-ins have defined filters that you can use to classify transactions into service classes for your domain policy. These filters are preloaded into the EWLM Control Center; all you must do is set up the appropriate application in the EWLM Control Center and specify what filters you want to use. The filters you specify during the setup are available in the domain policy when you create your transaction class rules for the application. See “EWLM-supplied filter types” on page 35 for a complete list of filters available to all applications and “Application-specific filter types” on page 40 for additional filters for Webserving plug-ins which includes the HTTP server.

Enabling ARM on the HTTP Server with an independent HTTP plug-in:

Use this topic to enable ARM on the HTTP Server with an independent HTTP plug-in.

The specific tasks you must complete to enable ARM vary depending on the target operating system. Select from the list below a topic collection that is specific to the operating system on which you want to enable ARM on the HTTP Server with an independent HTTP plug-in. Steps are provided for only AIX




and Windows, though the setup should be similar for other HTTP server platforms. See The Apache Software Foundation (ASF)  (<http://apache.org/>) for further information.

Enabling ARM on Independent HTTP plug-in on AIX:


Follow the instructions in this topic to enable ARM on Independent HTTP plug-in on AIX.

The following procedures apply to IBM HTTP Server 2.0.x and later, as well as Apache server 2.0.47 and later.

Note:

- Throughout this topic, *WAS_PLUGIN_HOME* refers to the path to which the WebSphere Application Server plugin is installed. For WebSphere Application Server v5.1, this is the same as *WAS_HOME*. See Manually configuring IBM HTTP Server powered by Apache 2.0  or Configuring IBM HTTP Server Version  for further information.
- For more information about IBM HTTP Server and working with Web serving plug-ins, refer to the HTTP Server Information Center .


1. Ensure that the EWLM managed server is configured and started, and that it is communicating properly with its EWLM domain manager.
2. Ensure that the middleware application is installed and configured. Use the specific product documentation if you need assistance.
3. Ensure that the user who starts the application is authorized to ARM services. See “Authorizing AIX non-root users to start EWLM on an AIX managed server” on page 118 for instructions.
4. Download the `mod_arm4_ap20.so` plug-in module and build it.

For a download link and instructions, see The Apache Software Foundation (ASF)  (<http://apache.org/>). If necessary, modify the application source to specify the application and transaction names as defined in the IBM Web serving plug-in filter.

Note: If you use the source of the application instrumented with ARM 4.0 to monitor the independent plug-in for the HTTP server, the ARM application name and transaction names will not match the names used in the Web serving plug-in application filter defined in the EWLM Control Center. Therefore, if you create a domain policy using the Web serving plug-in application filter, you are not able to monitor this independent HTTP server application. To monitor this independent HTTP server application, you must either modify the application source code to use the names that the IBM Web serving plug-in filter uses in the domain policy or create a new application filter in the domain policy that specifies the application and transaction names used by the HTTP server application.

If you do not modify the source of the application, you must monitor the ASF-downloadable `mod_arm4.c` plug-in in an EWLM service policy using the *Apache HTTP Server* application name, and any EWLM transaction class rule references to **EWLM: Transaction name** only resolve to true for a filter value of *HTTP Request*.

- a. Obtain the source of the application instrumented with ARM 4.0 for the HTTP server application.

You can find the source on the Apache Software Foundation (ASF)  (http://svn.apache.org/repos/asf/httpd/mod_arm4/trunk/) Web site.

- b. Open the `mod_arm4.c` file in a text editor.
- c. Change lines of code as follows: Change
`#define DEFAULT_APP_NAME "Apache HTTP Server"`

to

- ```

| #define DEFAULT_APP_NAME "IBM Webserving Plug-in"
|
| d. Change lines of code as follows: Change
| #define DEFAULT_TRAN_NAME "HTTP Request"
|
| to
| #define DEFAULT_TRAN_NAME "WebRequest"
|
| e. Compile the application source and continue setting up the application as described in the
| application documentation.
|
| 5. Add the plug-in module to the IHS2_HOME/modules directory.
| 6. Add the following line after all other LoadModule lines in the IHS2_HOME/conf/httpd.conf file:
| LoadModule arm4_module IHS2_HOME/modules/mod_arm4_ap20.so
| 7. Restart the HTTP Server.
| 8. Verify that ARM is enabled by using the lsarm command.

```

**Note:** The lsarm command displays information about applications instrumented with ARM and registered with the operating system. For more information on using this command, see “lsarm” on page 327.




Each of the HTTP Server plug-ins have defined filters that you can use to classify transactions into service classes for your domain policy. These filters are preloaded into the EWLM Control Center; all you must do is set up the appropriate application in the EWLM Control Center and specify what filters you want to use. The filters you specify during the setup are available in the domain policy when you create your transaction class rules for the application. See “EWLM-supplied filter types” on page 35 for a complete list of filters available to all applications and “Application-specific filter types” on page 40 for additional filters for Webserving plug-ins which includes the HTTP server.


*Enabling ARM on Independent HTTP plug-in on Windows:*

Follow the instructions in this topic to enable ARM on Independent HTTP plug-in on Windows.

The following procedures apply to IBM HTTP Server 2.0.x and later, as well as Apache server 2.0.47 and later.

**Note:**


- Throughout this topic, *WAS\_PLUGIN\_HOME* refers to the path to which the WebSphere Application Server plugin is installed. For WebSphere Application Server v5.1, this is the same as *WAS\_HOME*. See [Manually configuring IBM HTTP Server powered by Apache 2.0](#)  or [Configuring IBM HTTP Server Version](#)  for further information.
  - For more information about IBM HTTP Server and working with Web serving plug-ins, refer to the [HTTP Server Information Center](#) .
1. Ensure that the EWLM managed server is configured and started, and that it is communicating properly with its EWLM domain manager.
  2. Ensure that the middleware application is installed and configured. Use the specific product documentation if you need assistance.
  3. Ensure that the user account under which the application will run is properly authorized to the ARM services. For more information, see “Authorizing Windows users to ARM services” on page 166.
  4. Download the mod\_arm4\_ap20.dll plug-in module and build it.

For a download link and instructions, see [The Apache Software Foundation \(ASF\)](#)  (<http://apache.org/>). If necessary, modify the application source to specify the application and transaction names as defined in the IBM Web serving plug-in filter.

**Note:** If you use the source of the application instrumented with ARM 4.0 to monitor the independent plug-in for the HTTP server, the ARM application name and transaction names will not match the names used in the Web serving plug-in application filter defined in the EWLM Control Center. Therefore, if you create a domain policy using the Web serving plug-in application filter, you are not able to monitor this independent HTTP server application. To monitor this independent HTTP server application, you must either modify the application source code to use the names that the IBM Web serving plug-in filter uses in the domain policy or create a new application filter in the domain policy that specifies the application and transaction names used by the HTTP server application.

If you do not modify the source of the application, you must monitor the ASF-downloadable mod\_arm4.c plug-in in an EWLM service policy using the *Apache HTTP Server* application name, and any EWLM transaction class rule references to **EWLM: Transaction name** only resolve to true for a filter value of *HTTP Request*.

- a. Obtain the source of the application instrumented with ARM 4.0 for the HTTP server application.

You can find the source on the Apache Software Foundation (ASF)  ([http://svn.apache.org/repos/asf/httpd/mod\\_arm4/trunk/](http://svn.apache.org/repos/asf/httpd/mod_arm4/trunk/)) Web site.

- b. Open the mod\_arm4.c file in a text editor.
- c. Change lines of code as follows: Change

```
#define DEFAULT_APP_NAME "Apache HTTP Server"
```

to

```
#define DEFAULT_APP_NAME "IBM Webserving Plug-in"
```

- d. Change lines of code as follows: Change

```
#define DEFAULT_TRAN_NAME "HTTP Request"
```

to

```
#define DEFAULT_TRAN_NAME "WebRequest"
```

- e. Compile the application source and continue setting up the application as described in the application documentation.

5. Add the plug-in module to the *IHS2\_HOME\modules* directory.

6. Add the following line after all other LoadModule lines in the *IHS2\_HOME\conf\httpd.conf* file:

```
LoadModule arm4_module IHS2_HOME\modules\mod_arm4_ap20.dll
```

7. Ensure that the user who starts the application is authorized to ARM services. See “Authorizing Windows users to ARM services” on page 166 for instructions.

8. Restart the HTTP Server.

9. Verify that ARM is enabled by using the *ewlmWinADTool.exe* program.

**Note:** *ewlmWinADTool* displays information about applications instrumented with ARM and registered with the operating system. For more information on using this command, see “*ewlmWinADTool.exe*” on page 328.

Each of the HTTP Server plug-ins have defined filters that you can use to classify transactions into service classes for your domain policy. These filters are preloaded into the EWLM Control Center; all you must do is set up the appropriate application in the EWLM Control Center and specify what filters you want to use. The filters you specify during the setup are available in the domain policy when you create your transaction class rules for the application. See “EWLM-supplied filter types” on page 35 for a complete list of filters available to all applications and “Application-specific filter types” on page 40 for additional filters for Webserving plug-ins which includes the HTTP server.

### Enabling ARM on the HTTP Server for iSeries with an iSeries HTTP plug-in:

Use this topic to enable ARM on the HTTP Server for iSeries with an iSeries HTTP plug-in.

The specific tasks you must complete to enable ARM vary depending on the target operating system. Select from the list below a topic collection that is specific to the operating system on which you want to enable ARM on the HTTP Server for iSeries with an iSeries HTTP plug-in.

*Enabling ARM on HTTP plug-in for iSeries on i5/OS:*

Follow the instructions in this topic to enable ARM on HTTP plug-in for iSeries on i5/OS.

The following procedures apply to IBM HTTP Server for iSeries.

**Note:**

- Throughout this topic, *WAS\_PLUGIN\_HOME* refers to the path to which the WebSphere Application Server plugin is installed. For WebSphere Application Server v5.1, this is the same as *WAS\_HOME*. See [Manually configuring IBM HTTP Server powered by Apache 2.0](#) or [Configuring IBM HTTP Server Version](#) for further information.
  - For more information about IBM HTTP Server and working with Web serving plug-ins, refer to the [HTTP Server Information Center](#).
1. Ensure that the EWLM managed server is configured and started, and that it is communicating properly with its EWLM domain manager.
  2. Ensure that the middleware application is installed and configured. Use the specific product documentation if you need assistance.
  3. Ensure that the user account under which the application will run is properly authorized to the ARM services. For more information, see “Authorizing i5/OS users to ARM services” on page 138.
  4. Ensure you have the latest required PTFs installed for EWLM to monitor the HTTP Server for iSeries application.
  5. Add the following directives to the configuration file:

```
LoadModule arm4_module /QSYS.LIB/QHTTPSVR.LIB/QZSRARM.SRVPGM
ArmLoadLibrary /QSYS.LIB/QSYS2.LIB/LIBARM4.SRVPGM
```

To edit the configuration file, follow these steps:

- a. Start the IBM Web Administration for iSeries interface.
  - b. Click the **Manage** tab.
  - c. Click the **HTTP Servers** subtab.
  - d. Select your HTTP Server (powered by Apache) from the **Server** list.
  - e. Expand **Tools**.
  - f. Click **Edit Configuration File**.
- Note:** The line mode editor functions as a simple text editor only and does not error check any changes to the configuration file.
- g. Click **OK** when you finish editing the configuration file.
  - h. Ensure that the user who starts the application is authorized to ARM services.
  - i. Stop and restart the HTTP Server.
6. Use the WRKARMJOB CL command to verify that ARM support on your system is enabled and that an application is currently calling ARM APIs.

**Note:** This command will show DB2 as an ARM-enabled application only after WebSphere Application Server is restarted and is driving a workload to DB2.

Each of the HTTP Server plug-ins have defined filters that you can use to classify transactions into service classes for your domain policy. These filters are preloaded into the EWLM Control Center; all you must do is set up the appropriate application in the EWLM Control Center and specify what filters you want to use. The filters you specify during the setup are available in the domain policy when you create your transaction class rules for the application. See “EWLM-supplied filter types” on page 35 for a complete list of filters available to all applications and “Application-specific filter types” on page 40 for additional filters for Webserving plug-ins which includes the HTTP server.

Module `mod_arm4_ap20`

View this topic to learn more about the i5/OS specific values used in the ARM directives.

## Enabling EWLM load balancing

Use this topic to learn how to enable EWLM to communicate with a load balancer.

EWLM has the ability to provide goal-based recommendations, called weights, to a load balancer so that the load balancer can distribute incoming work requests based on those recommendations. The load balancer maintains a connection with the domain manager which enables the domain manager to continuously send its weight recommendations to the load balancer. The load balancer then forwards the incoming requests based on the recommended distribution.

EWLM load balancing is supported on the CISCO Content Switch Module for the CISCO Catalyst family of switches, which is a load balancer.

To enable EWLM load balancing, you must ensure the following:

- The load balancer must identify the domain manager by the IP address or host name specified for the **-ma** parameter as well as the port specified with the **-lbp** or **-lps** parameter when configuring the domain manager.
- When you configure the domain manager, you must specify a value for either the **-lbp** parameter, the **-lbs** parameter, or both. The **-lbp** parameter describes the port used for non-SSL connections between load balancers and the domain manager. If you do not provide a value for the **-lbp** parameter, load balancers will not be able to connect to the domain manager without using SSL.

The **-lbs** parameter is the port used for SSL connections to the domain manager. Use the same certificate keystore (**-sslks**) and password (**-sslpw**) parameters that you specify on the domain manager for domain manager to managed server SSL. Both the load balancer and the domain manager must import the other's public certificate as a trusted certificate in their respective keystores. If you do not provide a value for the **-lbs** parameter, the load balancer will not be able to connect to the domain manager with an SSL connection.

If you do not provide a value for either the **-lbp** parameter, the **-lbs** parameters, or both, EWLM load balancing will be deactivated.

- The managed servers must be able to identify the application or system by the identification criteria of the load balancer. For example, the load balancer might know a particular system, on which the managed server is installed, by a different IP address than that which was made known to the domain manager. To view a list of the operating systems that the EWLM load balancing function supports, see “Enterprise Workload Manager,” on page 1

**Note:** To enable load balancing on a managed server running on AIX, you must install the load balancing `netwlm` fileset before you start the managed server. Ensure the you have the required fixes.

“Configuring a domain manager” on page 108

View this topic to learn how to configure a domain manager.

“EWLM load balancing” on page 23

Load balancing is an optional function of EWLM that is valuable if you want EWLM to send load balancing recommendations to a load balancer. This functions allows the load balancer to evenly distribute incoming work to ensure that performance goals are met.



## Enabling partition management

Use this topic to learn how to enable logical partition management in EWLM.

EWLM provides the capability for EWLM to automatically adjust the processing units among partitions defined in a partition workload group. EWLM calculates how to adjust the partition's resources by comparing the actual performance of work on the system to the expected performance as defined in the domain policy. This function enables EWLM to automatically adjust resources to help ensure that the goals defined in the domain policy are met.

To enable logical partition management, complete the following subtopics.

### Related concepts

"Partition workload group" on page 16

A *partition workload group* identifies a set of partitions that reside on the same physical system. The partition profile specifies the name of the partition workload group that it belongs to, if applicable.

"Partition management" on page 26

*Partition management* refers to the ability of EWLM to automatically adjust the processing power among partitions within a partition workload group. The partitions in a partition workload group must reside on the same physical machine.

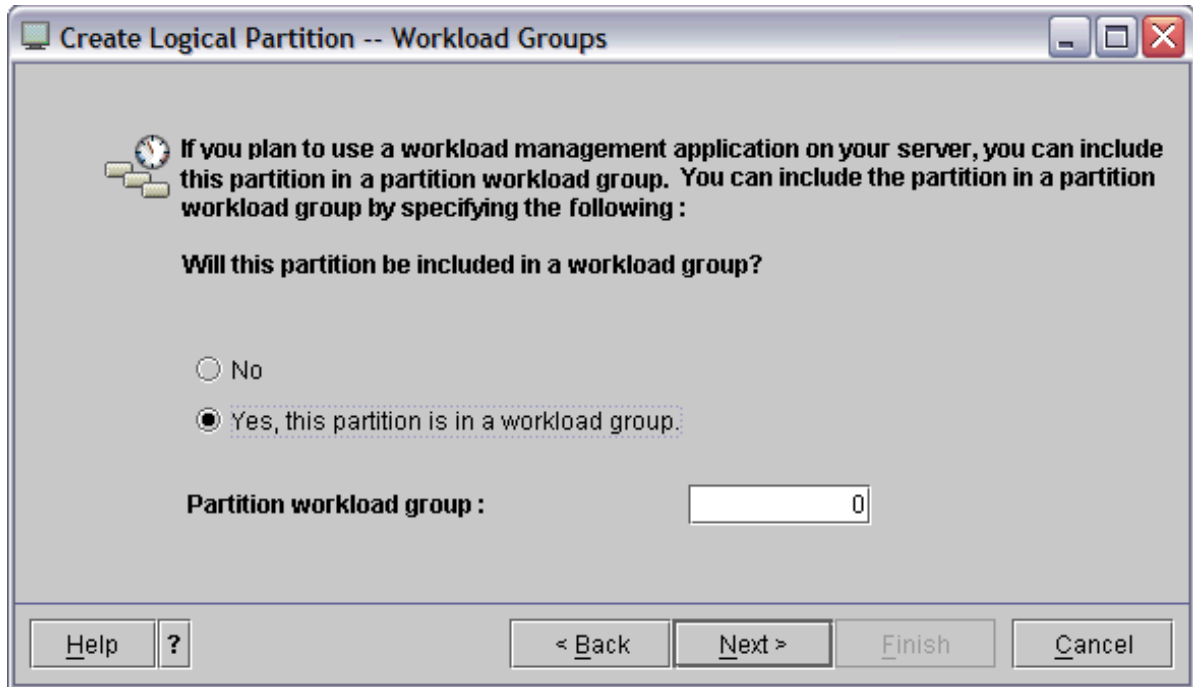
## Configuring the partition profile to specify a partition workload group

For EWLM to monitor a partition, the partition profile must indicate that it belongs to a partition workload group.

EWLM can adjust the resources of partitions that belong to a partition workload group. EWLM automatically identifies the partitions that it is to manage if the partition is defined in a partition workload group.

Complete the following steps to define the partition in a **Partition Workload Group**.

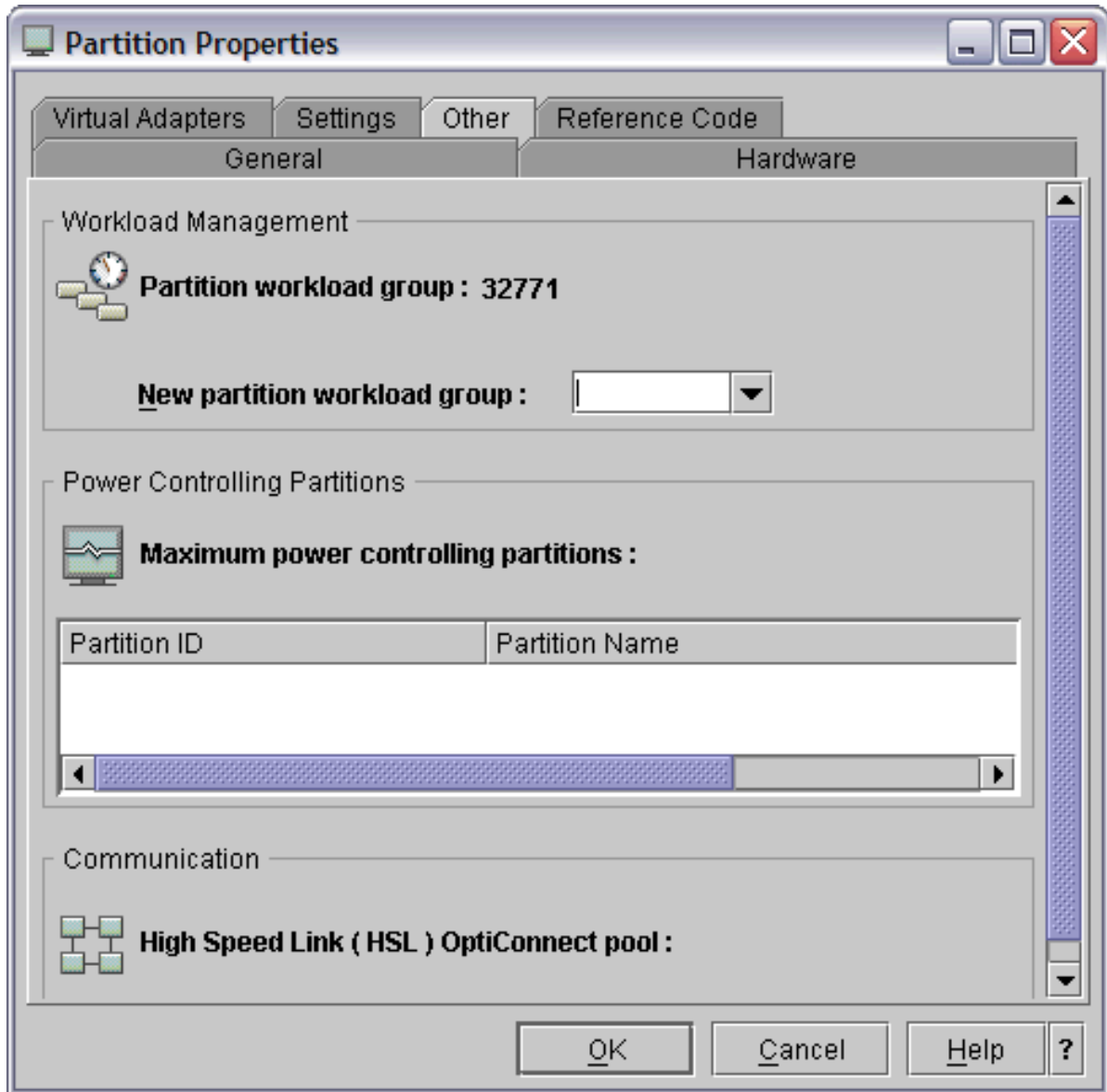
1. Log on to the Hardware Management Console.
2. In the navigation area, select **Server and Partition**.
3. Select **Server Management**.
4. In the contents area, open the server on which you want to create or change a logical partition profile.
5. If you want to create a new partition, complete the following steps:
  - a. Right-click **Partitions**, and select **Create** → **Logical Partition**.
  - b. Complete the Create Logical Partition wizard as indicated to create a logical partition. On the **Create Logical Partition - Workload groups** page, specify a **Partition Workload Group**. The following screen indicates where to define the logical partition workload group.



- c. On the **Create Logical Partition Profile - Processors** page, select **Shared**. By selecting this option, EWLM can move processing power from the shared processor pool to this partition, if necessary.
- d. Right-click the logical partition profile and select **Properties**, then **Hardware → Processors and Memory**. Ensure that **Allow shared processor pool utilization authority** is selected. By selecting this option, EWLM has the proper authority to make resource adjustments, if necessary.

**Note:** This change only applies to the logical partition properties. You must make a similar change to the **Partition Profile** if you want to apply this change to all subsequent profile activation and deactivation requests.

- 6. If you want to change an existing partition's properties, complete the following steps:
  - a. Right-click the logical partition profile and select **Properties**.
  - b. On the **Other** page, ensure that a **Partition workload group** is specified. The following screen illustrates the page in the Partition Properties page that indicates to which partition workload group this logical partition belongs.



c. If you want to change the partition workload group of the partition, follow the steps in “Changing a partition’s partition workload group” on page 252.

7. Repeat this process for each logical partition that you want to belong to a **Partition workload group**.

**Special consideration:** Changes to the logical partition properties take effect immediately. Changes to the logical partition profile take effect after you restart the logical partition that uses the profile. If you use the Hardware Management Console Version 5 Release 1 or later, the logical partition workload group on the logical partition profile level overwrites the logical partition workload group on the logical partition level when you activate the logical partition profile.

**Related information**

Logical partitions in the iSeries Information Center

View this topic in the iSeries Information Center to learn more about logical partitions.

**Installing and configuring the EWLM managed server on the partition**

For EWLM to manage a partition, you must define the logical partition as an EWLM managed server.

EWLM manages each logical partition as a separate EWLM managed server.

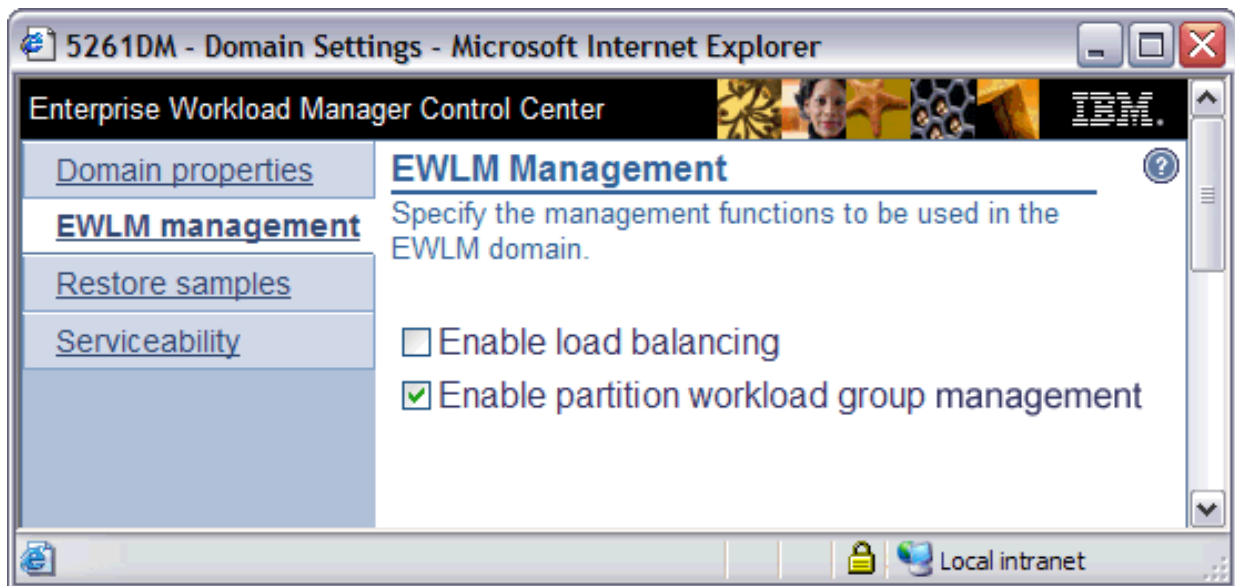
- | Complete the following steps to install and configure EWLM managed server on a partition.
- | 1. Ensure that the partition meets the hardware and software requirements for an EWLM managed server. For more information, see “Specified operating environments for EWLM” on page 74.
- | 2. Complete the managed server installation. For more information about how to install the EWLM managed server, see “Installing the EWLM managed servers” on page 95. This step assumes that the domain manager is already installed.
- | 3. Configure the managed server. For more information, see “Configuring a managed server” on page 111.
- | 4. Repeat this process for each partition that you want to be an EWLM managed server.

### Enabling EWLM partition management

| You must select to enable the partition management function of EWLM. This indicates to EWLM that it is to take action on the partition workload groups defined in the domain.

| Complete the following steps to enable partition management:

- | 1. Log in to the EWLM Control Center. For more information about what URL to use to access the Control Center, see “Starting the EWLM Control Center” on page 192.
- | 2. In the EWLM Control Center, click **Domain Settings**.
- | 3. In the left pane of the Domain Settings, click **EWLM Management**. The following page displays.



- | 4. Select **Enable partition workload group management**.
- | 5. Click **OK**.

| Partition management is enabled on the domain. EWLM will make resource adjustments based on the goals defined in the active service policy.

| To view the amount of resources allocated to each logical partition in a partition workload group, select **Monitor** → **Partitions** in the EWLM Control Center. EWLM adjusts the **Processing units** for each partition, as needed.

### Changing a partition’s partition workload group

| Within your IT environment, you may determine that you need to move a partition from one partition workload group to another. If this occurs in an EWLM domain, ensure that you complete some preliminary steps to ensure that EWLM will accept this change dynamically, without having to restart the partition.

- | Complete the following steps to change the partition workload group of a partition.
- | 1. Use the Hardware Management Console (HMC) to change the partition's partition workload group to None.
- | 2. In the EWLM Control Center, view the Partition Workload Group Details report for the partition's old partition workload group. Ensure that the partition no longer displays in the list of partitions for the old group in the report.
- | 3. Use the HMC to change the partition workload group for this partition from None to *user-defined-value*.
- | 4. View the Partition Workload Group Details report for the new partition workload group. Ensure that the partition displays in the list of partitions for the new group in the report.

## Customizing EWLM configuration properties

Use this topic to learn how to customize the EWLM configuration properties.

You might find it necessary, after you have already configured EWLM on your systems, to change a specific property, add firewall broker information, or set up problem determination data for one or all of the EWLM components. View the following topics for information about customizing EWLM configuration properties.

### Changing domain manager configuration properties

Use this topic to learn how to change domain manager configuration properties.

After you configure the domain manager on your system, you might want to change a parameter that you have already set up, change a security function, set up load balancing, add firewall broker support, or control the amount of tracing information the domain manager collects. The `changeDM` command enables you to customize the properties of your EWLM domain manager.

If you are familiar with the domain manager configuration wizard, you can use the wizard to change the managed server properties. To use the configuration wizard, see "Configuring a domain manager" on page 108.

- | For more information on issuing the command on your operating system, if supported, see "Command usage considerations" on page 91.

### Run `changeDM`

To change or customize the domain manager configuration properties, run the following command, replacing the example parameter values shown in italics with your own and adding the appropriate extension. Each parameter on this command, except `configID`, is optional.

```
changeDM configID -auth [None | ServerSSL | ClientServerSSL]
-slsks file -sslpw password
-ma address -mp port -jp port
-dn domainName
-fp port -fb list
-sa address -sp port
-lbp port -lbs port
-nt value -ct value
-et value -rt value
-m1 value -lbt value
-tcomm value -t1 value
-tlog class -dpa value
-dpn value -fl value
```

The following table describes each of the parameters and provides a place for you to enter your specific values if you print this page for reference.

Table 63. Parameters for changeDM command

| Parameter | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      | Your value |
|-----------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------|
| -auth     | <p>Security level used for communications in the EWLM domain between the EWLM domain manager, EWLM managed servers, and EWLM firewall brokers. This security level must be the same on all instances of EWLM within an EWLM domain. "level" must be one of the following values:</p> <p><b>None</b> No SSL keystore is supplied.<br/> <b>Note:</b> EWLM encrypts the data, but no authentication is performed.</p> <p><b>ServerSSL</b><br/>                     SSL is used to authenticate all communications within the EWLM domain. The EWLM domain manager or EWLM firewall broker authenticates to the EWLM managed servers.<br/> <b>Note:</b> If ServerSSL is selected, then you must also supply <b>-sslks</b> and <b>-sslpw</b>.</p> <p><b>ClientServerSSL</b><br/>                     SSL is used to authenticate all communications within the EWLM domain. The EWLM domain manager or EWLM firewall broker authenticates to the EWLM managed servers, and the EWLM managed servers authenticate to the EWLM domain manager or EWLM firewall broker.<br/> <b>Note:</b> If ClientServerSSL is selected, then you must also supply <b>-sslks</b> and <b>-sslpw</b>.</p> |            |
| configID  | Configuration ID for an EWLM domain manager instance.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |            |
| -ct       | Used to control the size of the component trace table. The value can be a number between 100 and 1000, or set to <b>Off</b> if no tracing is desired. If no value is provided, this property is set to the default value of 250.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |            |
| -dn       | Name of this EWLM domain. The EWLM domain name must be a character string. All characters are valid except the colon (:), question mark (?) symbol, double quotation mark (") symbol, greater than (>) symbol, less than (<) symbol, or bar ( ) symbol. The length of the ID must not exceed 64 characters.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |            |
| -dpa      | Used to control the maximum time, in days, that EWLM diagnostic logs are retained. The value can be a number between 0 and 365, or it can be set to <b>Off</b> . If the value is 0 or <b>Off</b> , no automatic cleanup of these logs occurs. If no value is provided, this property is set to the default value of 30.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |            |
| -dpn      | Used to control the number of EWLM diagnostic logs that are retained. The value can be a number between 0 and 250, or it can be set to <b>Off</b> . If the value is 0 or <b>Off</b> , no automatic cleanup of these logs occurs. If no value is provided, this property is set to the default value of 25.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |            |
| -et       | Used to control the size of the event trace table. The value can be a number between 100 and 1000 or set to <b>Off</b> if no tracing is desired. If no value is provided, this property is set to the default value 250.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |            |

Table 63. Parameters for changeDM command (continued)

| Parameter   | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       | Your value |
|-------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------|
| <b>-fb</b>  | <p>Used to define the EWLM firewall brokers that are a part of this EWLM domain. The list must have the following format:</p> <p><i>addr:port:SOCKS,addr:port:SOCKS</i></p> <p>where:</p> <p><i>addr</i> IP address or host name of the EWLM firewall broker.</p> <p><i>port</i> Port used by EWLM firewall broker to communicate with this domain manager. Port value must be between 1000 and 65535. This port value must match the <b>-fp</b> parameter on the EWLM firewall broker.</p> <p><i>SOCKS</i> Optional indicator designating that this firewall broker must be accessed by using a SOCKS server.</p> <p>Each item describing the firewall broker is separated by the colon character (:) and each firewall broker is separated from the next one with the comma character (,).</p> <p>If no value is provided, this property is set to the default value of null which means "no list defined."</p> |            |
| <b>-fl</b>  | Used to control how many error failures this EWLM instance can tolerate in one hour. The value can be a number between 0 and 100, or set to <b>Off</b> . If the value is 0 or <b>Off</b> , an unlimited number of failures is allowed. If no value is provided, this property is set to the default value of 50.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |            |
| <b>-fp</b>  | Port used by the EWLM domain manager to listen for communications from the EWLM firewall brokers in this domain. Port value must be between 1000 and 65535. If no value is provided, this property is set to the default value of null which means "no port defined."                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |            |
| <b>-jp</b>  | Port used by the EWLM domain manager to listen for communications from the EWLM Control Center. Port value must be between 1000 and 65535.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |            |
| <b>-lbp</b> | Port used by the EWLM domain manager to listen for public communications from network load balancing servers. The value can be any valid port number between 1 and 65535, or set to <b>Off</b> if no load balancing capability is desired. If no value is provided, this property is set to the default value of <b>Off</b> .                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |            |
| <b>-lbs</b> | Port used by the EWLM domain manager to listen for secure communications from network load balancing servers. The value can be any valid port number between 1 and 65535, or it can be set to <b>Off</b> if no load balancing capability is desired. If no value is provided, this property is set to the default value of <b>Off</b> .<br><b>Note:</b> If the <b>-lbs</b> load balancing is activated by providing a secure port, then the <b>-sslks</b> and <b>-sslpw</b> parameters must be supplied.                                                                                                                                                                                                                                                                                                                                                                                                          |            |
| <b>-lbt</b> | Used to control the size of the load balancing log table. The value can be a number between 100 and 1000, or set to <b>Off</b> if no tracing is desired. If no value is provided, this property is set to the default value of 250.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |            |
| <b>-ma</b>  | IP address or host name of this server.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |            |

Table 63. Parameters for changeDM command (continued)

| Parameter | Description                                                                                                                                                                                                                                                                                                                                                                                                                   | Your value |
|-----------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------|
| -ml       | Used to control the size of the message log table. The value can be a number between 100 and 1000, or set to <b>Off</b> if no tracing is desired. If no value is provided, this property is set to the default value of 250.                                                                                                                                                                                                  |            |
| -mp       | Port used by the EWLM domain manager to listen for communications from the EWLM managed servers. Port value must be between 1000 and 65535.                                                                                                                                                                                                                                                                                   |            |
| -nt       | Used to control the size of the communications trace table. The value can be a number between 100 and 1000, or set to <b>Off</b> if no tracing is desired. If no value is provided, this property is set to the default value of 250.                                                                                                                                                                                         |            |
| -rt       | Used to control the size of the performance data reporting trace table. The value can be a number between 100 and 1000, or set to <b>Off</b> if no tracing is desired. If no value is provided, this property is set to the default value of 250.                                                                                                                                                                             |            |
| -sa       | IP address or host name of a SOCKS server. This is required when the EWLM domain manager must access an EWLM firewall broker in the EWLM domain by using a SOCKS server. If no value is provided, this property is set to the default value of null which means "no SOCKS address defined."                                                                                                                                   |            |
| -sp       | Port of the SOCKS server. This is required when the EWLM domain manager must access an EWLM firewall broker in the EWLM domain by using a SOCKS server. The port number must be between 1 and 65535. If no value is provided, this property is set to the default value of null which means "no SOCKS server port."                                                                                                           |            |
| -sslks    | Path of the SSL keystore. If no value is provided, this property is reset to the default value of null which means "no keystore defined."                                                                                                                                                                                                                                                                                     |            |
| -sslpw    | Password used to protect the SSL keystore associated with the -sslks parameter. If no value is provided, this property is set to the default value of null which means "no keystore password defined."                                                                                                                                                                                                                        |            |
| -tcomm    | Used to generate a log file for debugging communications within an EWLM domain. The possible values are as follows:<br><br><b>0</b> Does not generate a log file<br><b>1</b> Generates a log file<br>The log file can become quite large in a short period of time, so this option should only be used when directed by IBM service. If no value is provided, this property is set to the default value of 0.                 |            |
| -tl       | Used to control the amount of data that is stored in the trace log. The value must be set to one of the following values:<br><br><b>Min</b> The minimum amount of trace information is captured.<br><b>Mid</b> A modest level of additional information is captured.<br><b>Max</b> The maximum amount of trace information is captured.<br>If no value is provided, this property is set to the default value of <b>Min</b> . |            |



Table 63. Parameters for changeDM command (continued)

| Parameter    | Description                                                                                                                                                                                                                                                                 | Your value |
|--------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------|
| <b>-tlog</b> | Name of class to be used as a trace plug-in. This plug-in is given control with trace information generated by this EWLM instance. If the value is <b>Off</b> , the plug-in is disabled. If no value is provided, this property is set to the default value of <b>Off</b> . |            |

## Changing managed server configuration properties

Use this topic to learn how to change managed server configuration properties.

After you configure your managed servers, you might need to change a parameter that you have already set up, add security function, set up firewall broker support, or control the amount of tracing information that the managed server collects. The changeMS command enables you to customize the properties of your EWLM managed servers.

If you are familiar with the managed server configuration wizard, you can use the wizard to change the managed server properties. To use the configuration wizard, see “Configuring a managed server” on page 111.

For more information on issuing the command on your operating system, if supported, see “Command usage considerations” on page 91.

### Run changeMS

To change or customize the managed server configuration properties, run this command, replacing the example parameter values shown in *italics* with your own and adding the appropriate extension to the command. Each parameter on this command, except **configID**, is optional.

**Note:** You must stop the managed server before you run this command.

```
changeMS configID -auth [None | ServerSSL | ClientServerSSL]
-sslks file -sslpw password
-ma address -mp port
-va address -vp port
-jp port
-nt value -ct value
-et value -jt value
-ml value -tl value
-tlog class
-dpa value -dpn value
-fl value
```

The following table describes each of the parameters and provides a place for you to enter your specific values if you print this page for reference.

Table 64. Parameters for changeMS command

| Parameter | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         | Your value |
|-----------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------|
| -auth     | <p>Security level used for communications in the EWLM domain between the EWLM domain manager, EWLM managed servers, and EWLM firewall brokers. This security level must be the same on all instances of EWLM within an EWLM domain. "level" must be one of the following values:</p> <p><b>None</b> No SSL keystore is supplied.<br/><b>Note:</b> EWLM encrypts the data, but no authentication is performed.</p> <p><b>ServerSSL</b><br/>SSL is used to authenticate all communications within the EWLM domain. The EWLM domain manager or EWLM firewall broker authenticates to the EWLM managed servers.<br/><b>Note:</b> If ServerSSL is selected, then you must also supply <b>-sslks</b> and <b>-sslpw</b>.</p> <p><b>ClientServerSSL</b><br/>SSL is used to authenticate all communications within the EWLM domain. The EWLM domain manager or EWLM firewall broker authenticates to the EWLM managed servers, and the EWLM managed servers authenticate to the EWLM domain manager or EWLM firewall broker.<br/><b>Note:</b> If ClientServerSSL is selected, then you must also supply <b>-sslks</b> and <b>-sslpw</b>.</p> |            |
| configID  | Configuration ID for an EWLM managed server instance.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |            |
| -ct       | Used to control the size of the component trace table. The value can be a number between 100 and 1000, or set to <b>Off</b> if no tracing is desired. If no value is provided, this property is set to the default value of 250.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |            |
| -dpa      | Used to control the maximum time, in days, that EWLM diagnostic logs are retained. The value can be a number between 0 and 365, or it can be set to <b>Off</b> . If the value is 0 or <b>Off</b> , no automatic cleanup of these logs occurs. If no value is provided, this property is set to the default value of 30.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |            |
| -dpn      | Used to control the number of EWLM diagnostic logs that are retained. The value can be a number between 0 and 250, or it can be set to <b>Off</b> . If the value is 0 or <b>Off</b> , no automatic cleanup of these logs occurs. If no value is provided, this property is set to the default value of 25.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |            |
| -et       | Used to control the size of the event trace table. The value can be a number between 100 and 1000 or set to <b>Off</b> if no tracing is desired. If no value is provided, this property is set to the default value 250.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |            |
| -fl       | Used to control how many error failures this EWLM instance can tolerate in one hour. The value can be a number between 0 and 100, or set to <b>Off</b> . If the value is 0 or <b>Off</b> , an unlimited number of failures is allowed. If no value is provided, this property is set to the default value of 50.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |            |
| -jp       | Port used by the EWLM managed server to listen for communications from Systems Management products. The port can be a valid port number between 1000 and 65535, or set to <b>Off</b> if no access is desired. If no value is provided, this property is set to the default value of <b>Off</b> .                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |            |

Table 64. Parameters for changeMS command (continued)

| Parameter     | Description                                                                                                                                                                                                                                                                                                                                                                                                                   | Your value |
|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------|
| <b>-jt</b>    | Used to control the size of the platform interface trace table. The value can be a number between 100 and 1000, or set to <b>Off</b> if no tracing is desired. If no value is provided, this property is set to the default value of 250.                                                                                                                                                                                     |            |
| <b>-ma</b>    | IP address or host name of the EWLM domain manager, or the IP address or host name of the EWLM firewall broker if the EWLM managed server cannot directly connect to the domain manager.                                                                                                                                                                                                                                      |            |
| <b>-ml</b>    | Used to control the size of the message log table. The value can be a number between 100 and 1000, or set to <b>Off</b> if no tracing is desired. If no value is provided, this property is set to the default value of 250.                                                                                                                                                                                                  |            |
| <b>-mp</b>    | Port of the EWLM domain manager, or the port of the EWLM firewall broker if the EWLM managed server cannot directly connect to the domain manager. The port value must match the <b>-mp</b> parameter of the EWLM domain manager or EWLM firewall broker. Port value must be between 1000 and 65535.                                                                                                                          |            |
| <b>-nt</b>    | Used to control the size of the communications trace table. The value can be a number between 100 and 1000, or set to <b>Off</b> if no tracing is desired. If no value is provided, this property is set to the default value of 250.                                                                                                                                                                                         |            |
| <b>-sslks</b> | Path of the SSL keystore. If no value is provided, this property is reset to the default value of null which means "no keystore defined."                                                                                                                                                                                                                                                                                     |            |
| <b>-sslpw</b> | Password used to protect the SSL keystore associated with the <b>-sslks</b> parameter. If no value is provided, this property is set to the default value of null which means "no keystore password defined."                                                                                                                                                                                                                 |            |
| <b>-tcomm</b> | Used to generate a log file for debugging communications within an EWLM domain. The possible values are as follows:<br><br><b>0</b> Does not generate a log file<br><b>1</b> Generates a log file<br>The log file can become quite large in a short period of time, so this option should only be used when directed by IBM service. If no value is provided, this property is set to the default value of 0.                 |            |
| <b>-tl</b>    | Used to control the amount of data that is stored in the trace log. The value must be set to one of the following values:<br><br><b>Min</b> The minimum amount of trace information is captured.<br><b>Mid</b> A modest level of additional information is captured.<br><b>Max</b> The maximum amount of trace information is captured.<br>If no value is provided, this property is set to the default value of <b>Min</b> . |            |
| <b>-tlog</b>  | Name of class to be used as a trace plug-in. This plug-in is given control with trace information generated by this EWLM instance. If the value is <b>Off</b> , the plug-in is disabled. If no value is provided, this property is set to the default value of <b>Off</b> .                                                                                                                                                   |            |
| <b>-va</b>    | IP address or host name of an HTTP proxy server. This is required when the managed server accesses the domain manager by using an HTTP proxy server. If no value is provided, this property is set to the default value of null which means "no HTTP proxy address defined."                                                                                                                                                  |            |

Table 64. Parameters for changeMS command (continued)

| Parameter | Description                                                                                                                                                                                                                                                                                               | Your value |
|-----------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------|
| -vp       | Port of the HTTP proxy server. This is required when the managed server accesses the EWLM domain manager by using an HTTP proxy server. The port number must be between 1 and 65535. If no value is provided, this property is set to the default value of null which means "no HTTP proxy port defined." |            |

## Changing firewall broker properties

Use this topic to learn how to change firewall broker properties.

After you configure your firewall brokers, you might need to change a parameter that you have already set up or you might want to control the amount of tracing information that the firewall broker collects. The changeFB command enables you to customize the properties of your firewall brokers.

If you are familiar with the managed server configuration wizard, you can use the wizard to change the firewall broker properties. To use the configuration wizard, see "Configuring a managed server" on page 111.

For more information on issuing the command on your operating system, if supported, see "Command usage considerations" on page 91.

### Run changeFB

To change or customize the firewall broker properties, run this command, replacing the example parameters shown in *italics* with your own and adding the appropriate extension for the command. Each parameter on this command, except **configID**, is optional.

**Note:** You must stop the firewall broker before you run this command.

```
changeFB configID -auth [None | ServerSSL | ClientServerSSL]
-sslks file -sslpw password
-ma address -mp port
-da address -dp port
-fp port
-sa address -sp port
-nt value -ct value
-et value -ml value
-tcomm value -tl value
-tlog class
-dpa value -dpn value
-fl value
```

The following table describes each of the parameters, alphabetically, and provides a place for you to enter your specific values if you print this page for reference.

Table 65. Parameters for changeFB command

| Parameter       | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         | Your value |
|-----------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------|
| <b>-auth</b>    | <p>Security level used for communications in the EWLM domain between the EWLM domain manager, EWLM managed servers, and EWLM firewall brokers. This security level must be the same on all instances of EWLM within an EWLM domain. "level" must be one of the following values:</p> <p><b>None</b> No SSL keystore is supplied.<br/><b>Note:</b> EWLM encrypts the data, but no authentication is performed.</p> <p><b>ServerSSL</b><br/>SSL is used to authenticate all communications within the EWLM domain. The EWLM domain manager or EWLM firewall broker authenticates to the EWLM managed servers.<br/><b>Note:</b> If ServerSSL is selected, then you must also supply <b>-sslks</b> and <b>-sslpw</b>.</p> <p><b>ClientServerSSL</b><br/>SSL is used to authenticate all communications within the EWLM domain. The EWLM domain manager or EWLM firewall broker authenticates to the EWLM managed servers, and the EWLM managed servers authenticate to the EWLM domain manager or EWLM firewall broker.<br/><b>Note:</b> If ClientServerSSL is selected, then you must also supply <b>-sslks</b> and <b>-sslpw</b>.</p> |            |
| <b>configID</b> | Configuration ID for an EWLM firewall broker instance.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |            |
| <b>-ct</b>      | Used to control the size of the component trace table. The value can be a number between 100 and 1000, or set to <b>Off</b> if no tracing is desired. If no value is provided, this property is set to the default value of 250.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |            |
| <b>-dpa</b>     | Used to control the maximum time, in days, that EWLM diagnostic logs are retained. The value can be a number between 0 and 365, or it can be set to <b>Off</b> . If the value is 0 or <b>Off</b> , no automatic cleanup of these logs occurs. If no value is provided, this property is set to the default value of 30.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |            |
| <b>-dpn</b>     | Used to control the number of EWLM diagnostic logs that are retained. The value can be a number between 0 and 250, or it can be set to <b>Off</b> . If the value is 0 or <b>Off</b> , no automatic cleanup of these logs occurs. If no value is provided, this property is set to the default value of 25.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |            |
| <b>-et</b>      | Used to control the size of the event trace table. The value can be a number between 100 and 1000 or set to <b>Off</b> if no tracing is desired. If no value is provided, this property is set to the default value 250.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |            |
| <b>-fl</b>      | Used to control how many error failures this EWLM instance can tolerate in one hour. The value can be a number between 0 and 100, or set to <b>Off</b> . If the value is 0 or <b>Off</b> , an unlimited number of failures is allowed. If no value is provided, this property is set to the default value of 50.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |            |
| <b>-fp</b>      | Port used by the EWLM firewall broker to listen for communications from the EWLM domain manager. Port value must be between 1000 and 65535.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |            |
| <b>-ma</b>      | IP address or host name of this server.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |            |

Table 65. Parameters for changeFB command (continued)

| Parameter     | Description                                                                                                                                                                                                                                                                                                                                                                                                                   | Your value |
|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------|
| <b>-ml</b>    | Used to control the size of the message log table. The value can be a number between 100 and 1000, or set to <b>Off</b> if no tracing is desired. If no value is provided, this property is set to the default value of 250.                                                                                                                                                                                                  |            |
| <b>-mp</b>    | Port used by the EWLM firewall broker to listen for communications from the EWLM managed servers. Port value must be between 1000 and 65535.                                                                                                                                                                                                                                                                                  |            |
| <b>-nt</b>    | Used to control the size of the communications trace table. The value can be a number between 100 and 1000, or set to <b>Off</b> if no tracing is desired. If no value is provided, this property is set to the default value of 250.                                                                                                                                                                                         |            |
| <b>-sa</b>    | IP address or host name of a SOCKS server. This is required when the EWLM domain manager must be accessed by using a SOCKS server. If no value is provided, this property is set to the default value of null which means "no SOCKS address defined."                                                                                                                                                                         |            |
| <b>-sp</b>    | Port of the SOCKS server. This is required when the domain manager must be accessed by using a SOCKS server. The port number must be between 1 and 65535. If no value is provided, this property is set to the default value of null which means "no SOCKS port defined."                                                                                                                                                     |            |
| <b>-sslks</b> | Path of the SSL keystore. If no value is provided, this property is reset to the default value of null which means "no keystore defined."                                                                                                                                                                                                                                                                                     |            |
| <b>-sslpw</b> | Password used to protect the SSL keystore associated with the <b>-sslks</b> parameter. If no value is provided, this property is set to the default value of null which means "no keystore password defined."                                                                                                                                                                                                                 |            |
| <b>-tcomm</b> | Used to generate a log file for debugging communications within an EWLM domain. The possible values are as follows:<br><br><b>0</b> Does not generate a log file<br><br><b>1</b> Generates a log file<br>The log file can become quite large in a short period of time, so this option should only be used when directed by IBM service. If no value is provided, this property is set to the default value of 0.             |            |
| <b>-tl</b>    | Used to control the amount of data that is stored in the trace log. The value must be set to one of the following values:<br><br><b>Min</b> The minimum amount of trace information is captured.<br><b>Mid</b> A modest level of additional information is captured.<br><b>Max</b> The maximum amount of trace information is captured.<br>If no value is provided, this property is set to the default value of <b>Min</b> . |            |
| <b>-flog</b>  | Name of class to be used as a trace plug-in. This plug-in is given control with trace information generated by this EWLM instance. If the value is <b>Off</b> , the plug-in is disabled. If no value is provided, this property is set to the default value of <b>Off</b> .                                                                                                                                                   |            |

## Changing the WebSphere Application Server administrator

Use this topic to learn how to change the WebSphere Application Server administrator.

To change the administrator of the WebSphere Application Server instance associated with the EWLM Control Center, you must run the `changeCC -admin` configuration command. If the WebSphere Application Server instance is started when you run the command, then the server is restarted so that your changes take effect immediately. If the WebSphere Application Server is stopped when you run the command, then the server is left in the stopped state when the command finishes running.

For more information on issuing the command on your operating system, if supported, see “Command usage considerations” on page 91.

To change the administrator of the WebSphere Application Server instance associated with a domain manager, run this command, replacing the values in italics with your own and adding the appropriate extension:

```
changeCC -admin configID -currentAdminUser userid -currentAdminPW password
 -newAdminUser userid -newAdminPW password
```

where each parameter is defined as follows.

Table 66. Parameter definitions

| Parameter                | Description                                                                                                                | Your value |
|--------------------------|----------------------------------------------------------------------------------------------------------------------------|------------|
| <b>configID</b>          | The configuration identifier, from the <code>createDM</code> command, used to identify the EWLM domain manager server job. |            |
| <b>-currentAdminPW</b>   | The password associated with the current administrator user ID.                                                            |            |
| <b>-currentAdminUser</b> | The user ID that is currently setup as the administrator of this WebSphere Application Server instance.                    |            |
| <b>-newAdminPW</b>       | The password associated with the new administrator user ID.                                                                |            |
| <b>-newAdminUser</b>     | The new user ID for administering this WebSphere Application Server instance.                                              |            |

## Changing the EWLM Control Center configuration

Use this topic to learn how to use commands and parameters to change your EWLM Control Center configuration.

You can change the ports that the EWLM Control Center uses. In addition, you can update the Control Center if you install a domain manager fix package.

To enable Web browsers to connect to the Control Center, you must run the `changeCC -controlCenterPorts` configuration command.

To update the Control Center to use the latest domain manager fix package that you installed, you must run the `changeCC -earLocation` configuration command.

If the WebSphere Application Server instance is started when you run the command, then the server is restarted so that your changes take effect immediately. If the WebSphere Application Server is stopped when you run the command, then the server is left in the stopped state when the command finishes running.

For more information on issuing the command on your operating system, if supported, see “Command usage considerations” on page 91.

Use the following command and parameters, as needed, replacing the values in italics with your own.

```
changeCC -controlCenterPorts configID
 -adminUser userid
 -adminPW password
 -httpPort [port1 | Same]
 -httpsPort [port2 | Same]
 -earLocation installPath
```

Where each parameter is defined as follows.

Table 67. Parameter definitions

| Parameter    | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    | Your value |
|--------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------|
| -adminPW     | The password associated with the administrator user ID.                                                                                                                                                                                                                                                                                                                                                                                                                                                        |            |
| -adminUser   | The user ID of the administrator of this WebSphere Application Server instance. This user ID must conform to the WebSphere Application Server requirements for an administrator.                                                                                                                                                                                                                                                                                                                               |            |
| configID     | The configuration identifier, from the createDM command, used to identify the EWLM domain manager.                                                                                                                                                                                                                                                                                                                                                                                                             |            |
| -httpPort    | The new port to enable HTTP access to the EWLM Control Center. Specify <b>Same</b> if you want to continue to use the currently configured port. Requests sent to this port will be redirected to the secure port.                                                                                                                                                                                                                                                                                             |            |
| -httpsPort   | The new secure port to which the port you specify for <b>-httpPort</b> redirects HTTP connections to the EWLM Control Center. Specify <b>Same</b> if you want to continue to use the currently configured port.                                                                                                                                                                                                                                                                                                |            |
| -earLocation | Specify the path to where the fix package was installed. The domain manager default installation paths are the following:<br><br>The default installation paths are the following: <ul style="list-style-type: none"> <li>• <b>AIX, Linux:</b> /opt/IBM/VE2/</li> <li>• <b>i5/OS :</b> /QIBM/ProdData/VE2/</li> <li>• <b>Windows:</b> C:\Program Files\IBM\VE2\</li> </ul> Use this parameter if you install a domain manager fix package and want to update the Control Center to use the latest fix package. |            |

### Related tasks

“Changing the EWLM Control Center configuration on z/OS”

View this topic to learn how to use scripts and parameters to change your EWLM Control Center configuration on z/OS.

### Changing the EWLM Control Center configuration on z/OS:

View this topic to learn how to use scripts and parameters to change your EWLM Control Center configuration on z/OS.

You can change the ports that the EWLM Control Center uses with the changeCC -WebPorts configuration command.

If the WebSphere Application Server instance is started when you run the command, then the server is restarted so that your changes take effect immediately. If the WebSphere Application Server is stopped when you run the command, then the server is left in the stopped state when the command finishes running.

For more information on issuing the command on your operating system, if supported, see “Command usage considerations” on page 91.

Use the following command and parameters, as needed, replacing the values in italics with your own.



```

| changeCC -webPorts configID
|
| -adminUser userid
| -adminPW password
| -httpPort port1 | Same
| -httpsPort port2 | Same

```

| Where each parameter is defined as follows.

| Parameter  | Description                                                                                                                                                                                                                                                                   |
|------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| -adminPW   | The password associated with the administrator user ID.                                                                                                                                                                                                                       |
| -adminUser | The user ID of the administrator of this WebSphere Application Server instance.                                                                                                                                                                                               |
| configID   | The configuration identifier, from the createDM command, used to identify the EWLM domain manager.                                                                                                                                                                            |
| -httpPort  | The new port to allow HTTP access to the EWLM Control Center and the WebSphere Application Server administration console. Specify <b>Same</b> if you want to continue to use the currently configured port. Requests sent to this port will be redirected to the secure port. |
| -httpsPort | The new secure port to which the port you specify for -httpPort redirects HTTP connections to the EWLM Control Center and the WebSphere Application Server administration console. Specify <b>Same</b> if you want to continue to use the currently configured port.          |

## | **Displaying EWLM configuration properties**

Use this topic to learn how to display the current properties for domain managers, managed servers, firewall brokers, and the EWLM Control Center.

You can display the current properties for each of the following elements of an EWLM domain:

### | **Displaying domain manager properties**

Use this topic to learn how to display domain manager properties.

Run the following command to display the current properties of your domain manager.

```

| Note: You can also use the configuration wizard to determine the properties. If you make changes to
| EWLM, the wizard will prefill the panels with the current settings. To use the configuration
| wizard, see “Configuring a domain manager” on page 108.

```

```

| For more information on issuing the command on your operating system, if supported, see “Command
| usage considerations” on page 91.

```

### | **Run displayDM**

```
displayDM configID
```

where **configID** is the EWLM working directory you specified when you ran the createDM command.

### | **Displaying managed server properties**

Use this topic to learn how to display managed server properties.

To display the current properties of a particular managed server, run the following command on the system where the managed server is installed.

| **Note:** You can also use the configuration wizard to determine the properties. If you make changes to  
 | EWLM, the wizard will prefill the panels with the current settings. To use the configuration  
 | wizard, see “Configuring a domain manager” on page 108.

| For more information on issuing the command on your operating system, if supported, see “Command  
 | usage considerations” on page 91.

**Note:** You must stop the managed server before you can display its properties.

### Run displayMS

displayMS **configID**

where **configID** is the EWLM working directory you specified when you ran the createMS command.

### Displaying firewall broker properties

Use this topic to learn how to display firewall broker properties.

To display the current properties of a particular firewall broker, run the following command on the system where the firewall broker is installed.

| **Note:** You can also use the configuration wizard to determine the properties. If you make changes to  
 | EWLM, the wizard will prefill the panels with the current settings. To use the configuration  
 | wizard, see “Configuring a domain manager” on page 108.

| For more information on issuing the command on your operating system, if supported, see “Command  
 | usage considerations” on page 91.

**Note:** You must stop the firewall broker before you can display its properties.

### Run displayFB

displayFB **configID**

where **configID** is the EWLM working directory you specified when you ran the createFB command.

### Displaying EWLM Control Center properties

Use this topic to learn how to display EWLM Control Center properties.

You can display the following EWLM Control Center-related properties by running the displayCC command on the domain manager.

- EWLM Control Center ports
- EWLM Control Center users
- EWLM Control Center groups

| For more information on issuing the command on your operating system, if supported, see “Command  
 | usage considerations” on page 91.

### Run displayCC

displayCC [-ports | -users | -groups] **configID** -adminUser **userid** -adminPW *password*

The following table describes each parameter alphabetically.

*Table 68. Parameter definitions*

| Parameter | Description                                             | Your value |
|-----------|---------------------------------------------------------|------------|
| -adminPW  | The password associated with the administrator user ID. |            |

Table 68. Parameter definitions (continued)

| Parameter         | Description                                                                                                                                          | Your value |
|-------------------|------------------------------------------------------------------------------------------------------------------------------------------------------|------------|
| <b>-adminUser</b> | The user ID of administrator of the WebSphere instance on which the EWLM Control Center runs.                                                        |            |
| <b>configID</b>   | The configuration identifier, from the createDM command, used to identify the EWLM domain manager server job.                                        |            |
| <b>-groups</b>    | The parameter to signify that you want to display the groups that are currently mapped to EWLM Control Center roles.                                 |            |
| <b>-ports</b>     | The parameter to signify that you want to display the ports currently used by the WebSphere Application Server instance for the EWLM Control Center. |            |
| <b>-users</b>     | The parameter to signify that you want to display the users that are currently mapped to EWLM Control Center roles.                                  |            |

## Managing an EWLM domain with the EWLM Control Center

Use this topic to learn about the various tasks that you can perform when you use the EWLM Control Center to manage the domain. In addition, you can view examples of how to read and interpret the performance data that EWLM reports in the Control Center. These examples will help you pinpoint the exact cause of a performance problem. Furthermore, you can learn about other tasks that relate to the EWLM Control Center such as how to enable Control Center tracing and deny users access to the Control Center.

### Related concepts

“EWLM Control Center” on page 13

The EWLM Control Center is a Web-based user interface that allows you to manage and monitor the performance of servers in an EWLM management domain. It is installed with and hosted by the domain manager.

## Managing a domain policy

View this topic to learn about the tasks you can perform with a domain policy in the Control Center. In addition, you will learn how to use a domain policy to monitor your EWLM domain.

In the EWLM Control Center, you can have multiple domain policies. You can work with the deployed domain policy or any domain policy that is available to the domain manager. The domain policies that are included in the **Set up** → **Domain policies** view of the Control Center include all domain policies that have been imported or created by using the EWLM Control Center wizard.

To create a new domain policy or import a domain policy, select **Set up** → **Domain policies** in the left pane of the EWLM Control Center. Then, click **Import** or **New**.

Only one domain policy can be deployed per EWLM domain. The domain manager of the EWLM domain is specified in the URL that you use to access the EWLM Control Center. To work with multiple EWLM domains, you need separate EWLM Control Center Web browser sessions for each domain manage with which you want to work.

In the EWLM Control Center, you can perform the following tasks from **Set up** → **Domain policies**:

- **New:** You can create a new domain policy or create a new domain policy based on an existing domain policy.
- **Import:** You can import a domain policy that is saved to your local file system. When the policy is imported, EWLM will verify that the syntax is correct. If not, the policy will not open in the EWLM Control Center. Use this with the **Export** function to copy domain policies from one EWLM domain to another. This makes it easier to share domain policies from one domain to another.

- **Export:** You can export a domain policy to your local file system. Use this with the **Import** function to copy domain policies from one EWLM domain to another. This is particularly useful if you have two EWLM domains with similar performance objectives. Then, you can use the same domain policy but modify it slightly on the second EWLM domain. Also, it is useful to export a copy of the domain policy to a backup system for disaster recovery purposes. Ensure that the directories that you export policies to have the proper security settings to protect the data that is contained in the domain policies.
- **Edit:** You can edit the domain policies that are stored on the domain manager that the EWLM Control Center is communicating with. In addition, you can edit the domain policy that is deployed onto the EWLM domain. Then, after you are finished editing the deployed domain policy, the domain policy will automatically deploy again. However, the service policy will not be reactivated.
- **Deploy:** When you want to make a domain policy available to the EWLM domain, you need to deploy it. When you deploy a domain policy, you can select to activate a service policy immediately or not. If a service policy is activated, EWLM will begin collecting performance data that is specific to the service policy activated. To deploy a domain policy, EWLM must be started on the domain manager.

To work specifically with the domain policy's service policies, you can perform the following task from **Manage** → **Service policies**:

- **Activate:** You can activate a service policy. The service policies listed are specific to the deployed domain policy. To activate a service policy, EWLM must be started on the domain manager.

To learn more about how to deploy a domain policy or how to create or edit a domain policy, see the EWLM Control Center online help. To view the domain policy task help, click **How do I** → **Work with** → **Domain policy** from the EWLM Control Center help.

#### Creating a domain policy:

The process to create a domain policy is the same for every user in that there is a core set of components of a domain policy. However, how you define each component depends on your specific business environment.

The following worksheets outline help you assess your business environments workload monitoring needs. Use your answers to create a domain policy that is based on your IT environment and its performance goals. These worksheets are intended to help you create a domain policy.

*Table 69. General assessment of business environment to identify what systems and applications EWLM is to monitor.*

| Question                                                                                                   | Your Answer | Tips                                                                                                                                                                                                                                            |
|------------------------------------------------------------------------------------------------------------|-------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| What systems process workloads that you want EWLM to monitor? How will you identify these systems to EWLM? |             | Use one or more of the following filters to identify these systems in the domain policy: <ul style="list-style-type: none"> <li>• EWLM: System Name</li> <li>• EWLM: OS Platform</li> <li>• EWLM: OS Level</li> <li>• EWLM: Hostname</li> </ul> |

Table 69. General assessment of business environment to identify what systems and applications EWLM is to monitor. (continued)

| Question                                                                                               | Your Answer | Tips                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
|--------------------------------------------------------------------------------------------------------|-------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p>What middleware applications do you want EWLM to monitor?</p>                                       |             | <p>An application may or may not instrument the Application Response Measurement (ARM) 4.0 standard.</p> <p>If it does, EWLM can collect granular, hop-level performance data. This is defined in a domain policy with a transaction class.</p> <p>If it does not, EWLM can monitor the system on which the application runs as a single entity. This is defined in a domain policy with a partition class.</p>                                                                                                                                                                           |
| <p>On which systems do those middleware applications run?</p>                                          |             | <p>These systems must run the EWLM managed server and that managed server must be connect to the EWLM domain.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| <p>Do you want to monitor application-level transactions separate from operating system processes?</p> |             | <p>If you monitor these types of work separately, EWLM allows you to identify which set of work is contributing the most to the overall system processor utilization rate.</p> <p>To monitor application-level transactions, you must define transaction classes in your domain policy. Each transaction class is specific to an application.</p> <p>To monitor operating system processes, you must define process classes in your domain policy. Each process class is specific to a platform. For example, you can monitor all of you AIX processes separate from i5/OS processes.</p> |

Table 70. Assessment of application environment for EWLM to monitor.

| Question                                                                                                                                                                    | Your Answer | Tips                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p>Do the applications instrument the Application Response Measurement (ARM) 4.0 standard?</p>                                                                              |             | <p>If yes, EWLM will collect granular performance data and can track the transactions as they flow from one application to another.</p> <p>If no, then use a partition class to monitor all work that runs on the system, which includes the application-level transactions and operating system processes.</p> <p>Note that this option, a partition class, does not provide as granular performance data as a transaction class.</p>                                                                                                                                                                                                                                                    |
| <p>For multi-tiered middleware application environments, what ARM-instrumented applications are the first applications in the domain that will process the transaction?</p> |             | <p>EWLM begins monitoring a transaction when it is processed by an ARM-instrumented application. This is referred to as an entry application because it is the first application in the domain to process the transaction.</p> <p>Create a transaction class in the domain policy for each application that is an entry application.</p>                                                                                                                                                                                                                                                                                                                                                  |
| <p>For applications that instrument ARM, what specific transactions do you want EWLM to monitor? How will you identify these transactions to EWLM?</p>                      |             | <p>Use any combination of the application-specific filters or any of the following EWLM filters to identify these transactions in a transaction class within your domain policy.</p> <ul style="list-style-type: none"> <li>• EWLM: Application Instance</li> <li>• EWLM: Hostname</li> <li>• EWLM: Transaction name</li> <li>• EWLM: User name</li> <li>• EWLM: URI</li> </ul> <p>If you want to monitor all of the transactions for an application or do not want to use filters to identify transactions, you can use the default transaction class that contains a wildcard '(\\*)' value. This default class allows you to monitor <i>all</i> of the application's transactions.</p> |

Table 70. Assessment of application environment for EWLM to monitor. (continued)

| Question                                                                                                  | Your Answer | Tips                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|-----------------------------------------------------------------------------------------------------------|-------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p>What performance goals is EWLM to measure the application-level transaction's performance against?</p> |             | <p>Each transaction class defines the transaction's performance goal.</p> <p>If you do not know what performance goal to use, create a domain policy with your best estimate of a performance goal. Then, allow EWLM to run for a period of time. Then, review the performance data to determine the 'actual' performance of the workload. Based on the 'actual' performance data, you can reexamine the goals defined in the domain policy to ensure that they are appropriate and realistic. If not, you can make adjustments to your domain policy and reactivate it.</p> |

Table 71. Assessment of operating system processes for EWLM to monitor

| Question                                                                                                                                          | Your Answer | Tips                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|---------------------------------------------------------------------------------------------------------------------------------------------------|-------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p>Do you want EWLM to monitor the performance of operating system processes?</p>                                                                 |             | <p>If yes, you need to add the applicable platforms to the domain policy and define process classes for each operating system.</p>                                                                                                                                                                                                                                                                                                                                                                                                     |
| <p>Do you want to monitor all of an operating system's processes or a specific subset? If yes, how will you identify these processes to EWLM?</p> |             | <p>To monitor all of an operating system's processes, create a process class that contains a wildcard '(\*)' value.</p> <p>To monitor a specific subset of operating system processes, create a process class for that platform and use any combination of the platform-specific filters or any of the following EWLM filters in a process class within your domain policy:</p> <ul style="list-style-type: none"> <li>• EWLM: System Name</li> <li>• EWLM: OS Platform</li> <li>• EWLM: OS Level</li> <li>• EWLM: Hostname</li> </ul> |

Table 71. Assessment of operating system processes for EWLM to monitor (continued)

| Question                                                                                       | Your Answer | Tips                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|------------------------------------------------------------------------------------------------|-------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| What performance goals is EWLM to measure the operating system processes' performance against? |             | Each process class defines which performance goal the transactions are to adhere to.<br><br>If you do not know what performance goal to use, create a domain policy with your best estimate of a performance goal. Then, allow EWLM to run for a period of time. Then, review the performance data to determine the 'actual' performance of the workload. Based on the 'actual' performance data, you can reexamine the goals defined in the domain policy to ensure that they are appropriate and realistic. If not, you can make adjustments to your domain policy and reactivate it. |

After you have evaluated your IT environment, the workloads for EWLM to monitor, and identified the performance goals to which those workloads are to be measured against, you can create a domain policy. Use the EWLM Control Center to create a domain policy. The domain policy wizard will guide you through the process of defining each component of a domain policy.

**Related tasks**

“Managing service policies”

View this topic to learn about the tasks you can perform with service policy in the Control Center. Also, you will learn how to use service policies within your business environment.

“Managing managed servers” on page 273

View this topic to learn about the tasks that you can perform with your managed servers. You can also learn how to use the Control Center to monitor the performance of your managed servers.

“Managing applications in EWLM” on page 274

View this topic to learn how to manage applications in your EWLM domain. EWLM can monitor application-level transactions for applications that use the Application Response Measurement (ARM) 4.0 standard APIs. The EWLM Control Center contains a predefined list of applications that use the ARM 4.0 standard.

“Managing platforms in EWLM” on page 278

View this topic to learn how to manage a platform in EWLM.

“Managing partitions in EWLM” on page 279

View this task to learn how you can manage a partition in EWLM. You will also learn how you can use the EWLM Control Center to manage the partitions that EWLM is to monitor.

“Enabling EWLM Control Center tracing” on page 280

View this topic to learn how to define the EWLM Control Center trace level.

“Removing EWLM Control Center users or groups” on page 281

Use this topic to learn how to remove EWLM Control Center access for a user ID.

**Managing service policies**

View this topic to learn about the tasks you can perform with service policy in the Control Center. Also, you will learn how to use service policies within your business environment.



Using the EWLM Control Center, you can define multiple service policies within a domain policy. A service policy contains all of the service classes that are defined in the domain policy. The unique aspect of each service policy is that you can specify different goals in each service policy. Only one service policy can be active at a time.

For example, in one domain policy, you can specify a service policy for different times of the day or days of the week. In addition, in each service policy, you can specify different goals. These goals can directly relate to service level agreements (SLA) that you have with business partners. In addition, the goals can relate to your own business goals. Therefore, if you have a contract with a company that guarantees that they receive an average response time that is less than one second when they access your Web database, you can specify this as a service class goal. You can create a service class for each SLA. In addition, if the SLA has other requirements for various times of the day or day of the week, you can create different service policies that contain different goals.

You should limit the number of service classes in a service policy to only the classes of work that have distinguishably different performance objectives. If you are considering to use more than 100 transaction, process, or service classes, you need to monitor the memory and processor usage of the managed servers to ensure optimal performance.

In the EWLM Control Center, select **Set up** → **Domain policies** and select to edit an existing domain policy or to create a new domain policy. When editing a domain policy or creating a new one, you can perform the following actions:

- **Create:** You can create multiple service policies in a domain policy. You can create a new service policy based on an existing service policy or create a service policy from scratch.
- **Edit:** You can edit a service policy in a domain policy.
- **Delete:** You can remove a service policy from the domain policy by selecting to delete it. After the delete request is completed, you are no longer able to access it in the EWLM Control Center.

To work specifically with the domain policy's service policies, you can perform the following tasks by selecting **Manage** → **Service policies**:

- **Activate:** You can activate a service policy on the EWLM domain. The service policies listed are specific to the deployed domain policy. To activate a service policy, EWLM must be started on the domain manager.

To learn more about how to activate a service policy or how to create or edit a service policy, see the EWLM Control Center online help. To view the service policy task help, click **How do I** → **Work with** → **Domain policy** from the EWLM Control Center help. Then, click **Work with service policies**.

#### **Related tasks**

“Stopping the EWLM Control Center” on page 196

Use this topic to learn how to stop the EWLM Control Center. Even if you close the Web session for the EWLM Control Center, the EWLM Control Center processes are still running. To stop it completely, you must complete the task outlined in this topic.

## **Managing managed servers**

View this topic to learn about the tasks that you can perform with your managed servers. You can also learn how to use the Control Center to monitor the performance of your managed servers.

In the EWLM Control Center, you can view the managed servers that are included in the EWLM domain. You can also view the properties of each managed server and stop the EWLM function on a managed server.

To work with the managed servers in the EWLM Control Center, click **Manage** → **Managed servers**.

In the EWLM Control Center, you can perform the following tasks with a managed server:

- **Properties:** You can view information that is specific to the managed server. For example, you can determine what service policy is active on the server or the state of the server from EWLM’s perspective.
- **Disable:** You can stop the EWLM function on a managed server. The disable request ends the communication between the domain manager and the managed server. After the disable request is completed, the managed server name remains in the list of managed servers for seven days and changes to a state of disabled. After seven days, the managed server is removed from the list. To enable EWLM on the managed system, you must manually start the EWLM function on the managed server. To start EWLM on the managed server, see “Starting EWLM on the managed server” on page 190.

To learn more about viewing the properties of a managed server or disabling EWLM on the managed server, see the EWLM Control Center online help. To view the managed server task help, click **How do I** → **Work with** → **Managed servers** from the EWLM Control Center online help.

In addition to working with the managed servers, you can also view performance data that is specific to the managed servers. You can learn about the number of transactions that each managed server is processing. More specifically, you can determine which servers are processing work requests for a specific service class. Also, you can view the processor utilization monitor to determine the amount of resources that are being used on the managed server.

To work with the managed server’s performance data in the EWLM Control Center, click **Monitor** → **Managed servers**. Then, select a managed server and an action to view performance data, such as the **Processor Utilization Monitor** or **Managed server details**.

#### Related tasks

“Stopping the EWLM Control Center” on page 196

Use this topic to learn how to stop the EWLM Control Center. Even if you close the Web session for the EWLM Control Center, the EWLM Control Center processes are still running. To stop it completely, you must complete the task outlined in this topic.

## Managing applications in EWLM

View this topic to learn how to manage applications in your EWLM domain. EWLM can monitor application-level transactions for applications that use the Application Response Measurement (ARM) 4.0 standard APIs. The EWLM Control Center contains a predefined list of applications that use the ARM 4.0 standard.

To manage an application, complete the following:

1. Ensure that the application for EWLM to monitor uses the ARM 4.0 standard APIs. This allows the application to communicate with EWLM properly.
2. Ensure that ARM is enabled. For more information on how to enable ARM, see “Enabling ARM on IBM-instrumented applications” on page 205.
3. Define the application that EWLM is to monitor in the EWLM Control Center.  
EWLM has a predefined list of applications in the EWLM Control Center in **Set up** → **Applications**. The following is a list of predefined applications and their corresponding application names as defined in the EWLM Control Center. If the application you want to monitor is predefined, continue to the next step. If the ARM-instrumented application that EWLM is to monitor is not predefined, you must create an application filter policy and set up the application in the EWLM Control Center. For more information, see “Managing an application filter policy” on page 276.

Table 72. Predefined list of applications set up in the EWLM Control Center.

| Application Name | Description                                                                                   |
|------------------|-----------------------------------------------------------------------------------------------|
| WebSphere        | Defines the filters that you can use to monitor WebSphere Application Server v5 transactions. |

Table 72. Predefined list of applications set up in the EWLM Control Center. (continued)

| Application Name                 | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|----------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| WebSphere:<br>APPLICATION_SERVER | Defines the filters that you can use to monitor WebSphere Application Server v6 transactions.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| IBM DB2 Universal Database       | Defines the filters that you can use to monitor DB2 Universal Database transactions.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| IBM Webserving plugins           | Defines the filters that you can use to monitor the WebSphere Application Server plug-in or the independent plug-in transactions or both. Use transaction class rules that are unique to each plug-in to distinguish between the various application plug-ins.<br><br>To learn more about the applications that are included in this Web serving plug-in group, see “Application instrumentation” on page 20.                                                                                                                                                                                     |
| User-defined application         | If an application that you want EWLM to monitor is not in the predefined list of applications, you must add the application by defining the application name, ARM supported filters, and so on. You can use the Control Center to define the application, which creates an exportable application policy definition XML file, or you can import an existing XML file. However, the XML file must adhere to syntax rules for EWLM to import it successfully. To learn more about the syntax of an application policy definition XML file, see “Managing an application filter policy” on page 276. |

- Add the application to the domain policy. By adding it to a domain policy, you are able to create transaction class rules that are specific to the application.  
To add an application to a domain policy, see the EWLM Control Center online help. From the Control Center online help, click **How do I** → **Work with** → **Applications**. Then, complete the steps that describe how to add an application to the domain policy.
- View the Monitor section of the EWLM Control Center to determine if the performance goals for the application are met. More specifically, view the following reports that contain performance data that is specific to application-level transaction data.

Table 73. EWLM Control Center views to use when managing the applications in your EWLM domain.

| Monitor view               | Description                                                                                                                                                                                                                                                                                                                                                                                              |
|----------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Transaction classes report | Displays the average response time for each transaction class in the domain policy. This view contains a comprehensive list of transaction classes for all applications that EWLM is monitoring.                                                                                                                                                                                                         |
| Service classes report     | Indicates whether the work associated with the service classes is meeting their performance goals. View the service classes that contain work of the type transaction to determine if the application-level transactions meet or exceed their performance goals.                                                                                                                                         |
| Application topology       | Provides high-level view of all of the applications that are processing transactions in your EWLM domain. This allows you to view the number of transactions that flow from one application to the next. In addition, the topology allows you to pinpoint the exact location of a performance bottleneck, if one exists.<br><br>For more information, see “Monitoring application topology” on page 288. |

### Related tasks

“Stopping the EWLM Control Center” on page 196

Use this topic to learn how to stop the EWLM Control Center. Even if you close the Web session for

the EWLM Control Center, the EWLM Control Center processes are still running. To stop it completely, you must complete the task outlined in this topic.

### Managing an application filter policy:

The application filter policy defines the application-specific filter types that you can use in classification rules to identify transactions for EWLM to monitor. Each application that you want EWLM to monitor must have an application filter policy. The EWLM Control Center contains a predefined list of applications and their filter policies. However, you may need to edit these application filter policies or create a new application filter policy for an application that is not predefined in the Control Center.

The application filter policy defines the application name, version, and filters. The filters defined in the application filter policy must correspond exactly to the Application Response Measurement (ARM) filters that the application supports. For example, WebSphere Application Server supports the ARM transaction filter **BusName**. Therefore, **BusName** must be defined in the application filter policy for WebSphere Application Server. Then, you can use the filter type **BusName** to identify the transactions for EWLM to monitor when you create classification rules for WebSphere Application Server.

The filter policy for an application specifies the application-specific filter types, not the “EWLM-supplied filter types” on page 35. The EWLM-supplied filters are made available to any application or platform that supports them, and they do not need to be called out in the application filter policy.

If any of the following situations occur, you will need to edit or create a new application filter policy:

- An application changes its ARM-instrumentation support for filters; therefore, the application filter policy defined in the EWLM Control Center is inaccurate. If the application name changes, you must create a new application filter policy using the new application name. If only the application filter support changes, you must edit the existing application filter policy.
- An application that you want EWLM to monitor is not in the list of predefined applications in the EWLM Control Center. You must create a new application filter policy.

The predefined applications in the EWLM Control Center already have an application filter policy XML file. You can view these application filter policies in the EWLM Control Center by selecting **Set up** → **Applications**. You might need to edit these predefined application filter policies if the application changes its ARM support. For example, if an application adds or removes support for an ARM filter type, the corresponding change might need to be made in the application filter policy.

If you have an ARM-instrumented application that you want EWLM to monitor that is not in the predefined list of applications, you must create an application filter policy. The filter policy must be set up in the EWLM Control Center. To create an application filter policy and set up the application in the EWLM Control Center, you can use one of the following two methods:

- Create an application filter policy using the EWLM Control Center
- Manually create an application filter policy

To edit an existing application filter policy or to create a new application filter policy based on an existing one, complete the following steps:

1. Log in to the EWLM Control Center.
2. Select **Set up** → **Applications**.
3. Select the application whose filter policy so that it is highlighted.
4. Click **Edit** or **New based on**, depending on which action you want to perform, from the **Select Action** field and click **Go**.
5. Edit the fields as necessary.
6. Click **OK** to save the policy.

Refer to your application or operating system product documentation to determine which specific ARM classification filters the application or operating system supports.

After you set up an application in the EWLM Control Center, you can add the application to a domain policy. Then, you can create transaction classes for the application so that EWLM can monitor specific application-level transactions. When defining the transaction classes, you can use the filter types that correspond directly to the filter names that are defined in the application filter policy. Therefore, it is essential that the application filter policy defines the application and filter names exactly as they are used in the ARM instrumentation. Otherwise, EWLM will not classify and monitor the application's transactions properly.

*Create an application filter policy using the EWLM Control Center:*

To use the Control Center to create the application filter policy, complete the following steps:

1. Log in to the EWLM Control Center.
2. Select **Set up** → **Applications**.
3. Click **Add** to define an application.
4. Complete the Application page as indicated.

*Manually create an application filter policy:*

To manually create the XML file for an application, complete the following steps:

1. Open a text editor of your choice.
2. Save the file with a unique name that uses the *.xml* extension in a directory that you can access from the EWLM Control Center.
3. Define the application using the syntax as shown in the example.

Consider the following when defining the application:

- Ensure that you adhere to the syntax shown in the example.
- Ensure that the value for *<Name>* exactly matches the name defined in the ARM support for the application.
- Ensure that the value for *<FilterType>* exactly matches the name of the ARM supported filter for the application.
- Use the *<Description>* field to differentiate between filter types that might vary from one release of the application to the next.
- Replace the text in italic with your own values. Ensure that the application and filter names correspond exactly to what the application uses. If they do not match, EWLM will not classify and monitor the application's transactions properly.

**Example:**

```
<xml>
<FilterPolicy>
 <Name>ApplicationName</Name>
 <Description>Description of application.</Description>
 <Version>Application version</Version>
 <FilterType>FilterType</FilterType>
 <Filters>
 <Filter>
 <Name>FilterName</Name>
 <Description>Description of filter.</Description>
 </Filter>
 <Filter>
 <Name>FilterName2</Name>
 <Description>Description of filter 2.</Description>
 </Filter>
 </Filters>
</FilterPolicy>
```

4. Import the application filter policy into the EWLM Control Center by completing the following steps:
  - a. Import the application filter policy into the EWLM Control Center by completing the following steps:
  - b. Select **Set up** → **Applications**.
  - c. Click **Import**.
  - d. Complete the fields as indicated to import the application policy.

## Managing platforms in EWLM

View this topic to learn how to manage a platform in EWLM.

For EWLM to monitor a system or partition, you must identify to EWLM which operating systems, which are referred to as platforms in the EWLM Control Center, to monitor. These systems or partitions are also referred to as EWLM managed servers.

To manage a platform, complete the following:

1. Ensure that the platform is a supported EWLM managed server. For a list of supported operating environments, see “Specified operating environments for EWLM” on page 74.
2. Ensure that ARM is enabled on the system that is the EWLM managed server.
3. If you want to monitor operating system processes by using a process class, ensure that the platform is defined in the EWLM Control Center. EWLM has a predefined list of platforms already set up in the EWLM Control Center in **Set up** → **Platforms**. The following is a list of predefined platforms as defined in the EWLM Control Center. You only need to edit the predefined platforms when indicated to do so by IBM.

**Note:** Only platforms that support process classes are defined in the Control Center. If an operating system does not support process classes, it is not specified in the list of platforms.

Table 74. Predefined list of platforms, which support process classes, set up in the EWLM Control Center.

Platform	Description
AIX	Defines the filters that you can use to monitor AIX operating system processes.
HP-UX	Defines the filters that you can use to monitor HP-UX operating system processes.
Linux	Defines the filters that you can use to monitor Linux operating system processes.
OS/400	Defines the filters that you can use to monitor i5/OS operating system processes.
Solaris	Defines the filters that you can use to monitor Solaris operating system processes.
Windows	Defines the filters that you can use to monitor Windows operating system processes.

4. Add the platform to the domain policy. For EWLM to monitor operating system processes for an EWLM managed server, you need to add the platform to the domain policy. By adding it to a domain policy, you are able to create process classes or partition class rules that are specific to the platform. To add a platform to a domain policy, see the EWLM Control Center online help. From the Control Center online help, click **How do I** → **Work with** → **Platforms**. Then, complete the steps that describe how to add a platform to the domain policy.
5. View the Monitor section of the EWLM Control Center to determine if the performance goals for the platform are met. More specifically, view the following reports that contain performance data that is specific to operating system process data.

Table 75. EWLM Control Center views to use when managing platforms in your EWLM domain.

Monitor view	Description
Process classes report	Displays the average response time for each process class in the domain policy. This view contains a comprehensive list of process classes for all platforms whose operating system processes EWLM is monitoring.
Service classes report	Indicates whether the work associated with the service classes is meeting their performance goals. View the service classes that contain work of the type "process" to determine if the operating system processes meet or exceed their performance goals.
Server topology	Provides high-level view of all of the servers that are processing work requests in your EWLM domain. This allows you to view the number of transactions that flow from one server to the next. In addition, the topology allows you to pinpoint the exact location of a performance bottleneck, if one exists.  For more information, see "Monitoring server topology" on page 289.

#### Related tasks

"Stopping the EWLM Control Center" on page 196

Use this topic to learn how to stop the EWLM Control Center. Even if you close the Web session for the EWLM Control Center, the EWLM Control Center processes are still running. To stop it completely, you must complete the task outlined in this topic.

## Managing partitions in EWLM

View this task to learn how you can manage a partition in EWLM. You will also learn how you can use the EWLM Control Center to manage the partitions that EWLM is to monitor.

Use one or both of the following methods to manage the work that a partition processes:

- **Partition classes:** For EWLM to monitor the work that a partition processes, you must define a partition class. The partition class identifies the partition that EWLM is to monitor. EWLM monitors all of the work that the partition processes according to the service class goal that you define for that partition class. Then, you can use the EWLM Control Center reports to determine if the work that the partition processes adhere to the performance goal that you defined for the partition.

To view the performance data for the work assigned to a partition class, select **Monitor** → **Partition classes**.

- **Partition workload groups:** For EWLM to not only monitor the work that a partition processes but also manage the processing power for a partition, you must define the partition in a partition workload group and enable partition power management. Then, EWLM adjusts the processing power of the partitions in the partition workload group to alleviate performance problems. EWLM uses the performance goals that are defined in the partition classes to determine if a resource adjustment is necessary. EWLM does not make resource adjustments that could negatively effect the performance of other partitions in the partition workload group.

To view the performance data for partitions in a partition workload group, select **Monitor** → **Partitions**.

For EWLM to monitor a partition using either of the methods available, partition classes or partition workload groups, you must identify to EWLM which partitions to monitor. To do this, you must use a partition class.

**Note:** You can only create a partition class for platforms that support the EWLM partition management function.

To manage a partition, complete the following:

1. Ensure that the partition is an EWLM managed server. For a list of supported operating environments, see “Specified operating environments for EWLM” on page 74.
2. Ensure that ARM is enabled on the partition that is the EWLM managed server.
3. If you want to monitor a partition, ensure that the partition’s platform is defined in the EWLM Control Center. EWLM has a predefined list of platforms already set up in the EWLM Control Center in **Set up** → **Platforms**. The following is a list of predefined platforms as defined in the EWLM Control Center. You only need to edit the predefined platforms when indicated to do so by IBM.

**Note:** You can only create partition classes for the platforms that support the EWLM partition management function. Not all of the predefined platforms support this function.

Table 76. Predefined list of platforms set up in the EWLM Control Center that also support partition classes.

Platform	Description
AIX	Defines the filters that you can use to monitor AIX partitions.
Linux	Defines the filters that you can use to monitor Linux partitions.
OS/400	Defines the filters that you can use to monitor i5/OS partitions.

4. Add the platform to the domain policy. For EWLM to monitor a partition, you need to add the its platform to the domain policy. By adding it to a domain policy, you are able to create partition class rules that are specific to the platform.  
To add a platform to a domain policy, see the EWLM Control Center online help. From the Control Center online help, click **How do I** → **Work with** → **Platforms** from the EWLM Control Center online help. Then, complete the steps that describe how to add a platform to the domain policy.
5. Monitor the performance of the partition. For more information on the reports and monitors available to monitor the performance of a partition, see “Monitoring partitions in EWLM” on page 295

#### Related tasks

“Stopping the EWLM Control Center” on page 196

Use this topic to learn how to stop the EWLM Control Center. Even if you close the Web session for the EWLM Control Center, the EWLM Control Center processes are still running. To stop it completely, you must complete the task outlined in this topic.

“Partition classes” on page 50

A *partition class* identifies partitions that are to be managed with a single goal and business importance. The partition class is associated with a service class that contains the goal and business importance. You use partition classes as part of EWLM’s partition workload group management. EWLM manages the CPU resources based on the overall performance of partitions in a partition class.

“Partition workload group” on page 16

A *partition workload group* identifies a set of partitions that reside on the same physical system. The partition profile specifies the name of the partition workload group that it belongs to, if applicable.

“Partition management” on page 26

*Partition management* refers to the ability of EWLM to automatically adjust the processing power among partitions within a partition workload group. The partitions in a partition workload group must reside on the same physical machine.

“Enabling partition management” on page 249

Use this topic to learn how to enable logical partition management in EWLM.

## Enabling EWLM Control Center tracing

View this topic to learn how to define the EWLM Control Center trace level.



The WebSphere Application Server instance must be active before you run this command. To start the WebSphere Application Server, see “Starting the EWLM Control Center” on page 192.

The WebSphere Application Server instance under which the EWLM Control Center runs, sets tracing, by default to **None** when it restarts. If you want to trace events that occur on the EWLM Control Center, you must run the `changeCC -trace` command on the server where the domain manager is running.

To trace events that occur on the EWLM Control Center complete the following:

1. Go to the installation path for the domain manager. To view a list of default installation paths and usage notes for each operation system on which you can set the trace level, see “Command usage considerations” on page 91.
2. Run the following command:

```
changeCC -trace configID
 -adminUser userid -adminPW password -state [None | Event | All]
```

where each parameter is defined as follows:

Parameter	Description
-adminPW	The password associated with the WebSphere Application Server administrator user ID.
-adminUser	The user ID of the administrator of the WebSphere Application Server instance on which the EWLM Control Center runs.
configID	The configuration identifier that EWLM uses to identify the EWLM domain manager. You defined this value when you create the domain manager. You cannot change this value.
-state	The level of EWLM Control Center tracing to occur on the currently active WebSphere Application Server instance. This trace level only affects this session and is not saved when the WebSphere Application Server instance stops. Possible values follow:  <b>None</b> No tracing is enabled. <b>Event</b> Event tracing is enabled. <b>All</b> Full tracing, including event tracing, is enabled.

### Related tasks

“Stopping the EWLM Control Center” on page 196

Use this topic to learn how to stop the EWLM Control Center. Even if you close the Web session for the EWLM Control Center, the EWLM Control Center processes are still running. To stop it completely, you must complete the task outlined in this topic.

### Removing EWLM Control Center users or groups

Use this topic to learn how to remove EWLM Control Center access for a user ID.

To deny a user ID access to the EWLM Control Center, the user ID must exist on the domain manager’s local user registry and be currently assigned to an EWLM role. If it is not assigned to an EWLM role, then it is already denied access to the EWLM Control Center.

To remove user IDs to EWLM Control Center roles, use one of the following methods:

- Run the EWLM domain manager configuration wizard
- Run the `changeCC` command

If the WebSphere Application Server instance is active when you run the command, then the EWLM Control Center is restarted so that your changes take effect immediately. If the WebSphere Application Server is stopped when you run the command, then the server is left in the stopped state when the command finishes running.

- **Run the EWLM domain manager configuration wizard**

The domain manager configuration wizard enables you to remove user IDs from EWLM Control Center roles. To access the domain manager configuration wizard, follow these steps:

1. Go to the `$D(install)/IBM/VE2/EWLM/bin` directory where `$D(install)` maps to the operating system-specific default directory in which products are typically installed. If you installed the domain manager to a different directory, then run the command from the `EWLM/bin` subdirectory of that directory.

2. Run the following command:

```
configWizardDM
```

For more information about starting the wizard on your operating system, if supported, see, “Configuration wizard usage considerations” on page 106.

3. Select the **Add or remove users to EWLM Control Center roles** task. Complete the wizard as indicated. Click **Help** if you need assistance.

- **Run the changeCC command**

To remove users from EWLM Control Center roles, run this command, replacing the example parameters shown in italics with your own and adding the appropriate command extension.

```
changeCC -removeUser configID
 -adminUser userid -adminPW password
 -roleUser user -role [Administrator | Monitor | Operator]
```

To remove groups from EWLM Control Center roles, run this command, replacing the example parameters shown in italics with your own and adding the appropriate command extension.

```
changeCC -removeGroup configID
 -adminUser userid -adminPW password
 -roleGroup group -role [Administrator | Monitor | Operator]
```

For more information about issuing the command on your operating system, if supported, see, “Command usage considerations” on page 91.

The following table describes each of the parameters and provides a place for you to enter your specific values if you print this page for reference.

Table 77. Parameters for the `changeCC -removeGroup` and `changeCC -removeUser` commands.

Parameter	Description	Your value
<b>-adminPW</b>	The password associated with the <b>-adminUser</b> parameter.	
<b>-adminUser</b>	The WebSphere Application Server administrator user ID.	
<b>configID</b>	The configuration identifier that EWLM uses to identify the EWLM domain manager. You defined this value when you create the domain manager. You cannot change this value.	
<b>-role</b>	The role that the user or group is currently assigned. Possible values are Administrator, Monitor, or Operator.	
<b>-roleGroup</b>	The group or groups that you want to remove. The same syntactical rules apply to the <b>-roleGroup</b> parameter as the <b>-roleUser</b> parameter.	
<b>-roleUser</b>	The user or users that you want to access the EWLM Control Center. Specify multiple user IDs by separating each user ID with a colon (:). For example, <code>user1:user2:user3</code> .	

### Related tasks

“Stopping the EWLM Control Center” on page 196

Use this topic to learn how to stop the EWLM Control Center. Even if you close the Web session for

the EWLM Control Center, the EWLM Control Center processes are still running. To stop it completely, you must complete the task outlined in this topic.

“Assigning users to EWLM Control Center roles” on page 117

View this topic to learn how to assign user IDs that exist on the domain manager’s local user registry to access the EWLM Control Center.

## Monitoring performance data with EWLM

View this topic collection to learn about the reports and monitors that you can use to monitor the performance of work processed by the EWLM domain in which EWLM monitors. You can use any combination of the reports and monitors to obtain the performance data that suites your needs.

One of the main benefits of EWLM is the ability to monitor the performance of servers and applications in your EWLM domain. The Monitor section of the EWLM Control Center provides many views to obtain performance data. This topic describes the views that are available. You need to use a combination of these views to determine if a problem exists in your EWLM domain.

The monitor function of EWLM provides, but is not limited to, the following:

- Service classes that meet or do not meet their performance goals.
- | • Hop details that indicate the amount of time that incurs at each hop that contributes to the overall transaction time.
- | • Processor usage for work assigned to a specific transaction, process, or partition class.
- | • Processor usage for an entire managed server.
- | • Topology data that shows the specific applications and servers that process work that EWLM classifies to a specific transaction or process class.
- | • Topology data for all work using a particular service class performance goal.
- Transaction statistics for a specific time interval.
- Performance index for a specific time interval.
- Transaction statistics that include the number of transactions that have completed successfully or unsuccessfully.

In the EWLM Control Center, you can access the performance data by selecting **Monitor** in the left pane of the Web browser. The **Monitor** section allows you to work with the following types of information:

### High-level views

When determining whether a problem exists in your EWLM domain, you need to begin by reviewing the high-level views.

#### Exception reports

Specifies the service classes that are not meeting their performance objectives. Use this report to view performance statistics that are specific to service classes that are not meeting their goals. A service class is listed in the exception report if its performance index (PI) is greater than 1. The higher the PI, the greater variance of the service class achieving its goal. The PI should be 1.0 or less. If no services classes are listed in this report, all of the service classes are meeting or exceeding their goals.

#### Service classes

Specifies all of the service classes that are defined in the deployed domain policy. You can view performance statistics for each service class. In addition, you can identify which transaction, process, or partition classes are using the performance goal defined for each service class.

#### Transaction classes

Specifies all of the transaction classes for each application defined in the deployed domain policy. You can view performance statistics for each transaction class. This view is particularly useful if a service class is not meeting its goal. You can use this view to

pinpoint the transaction class, within the service class, that may be causing the service class to miss its goal. You can view the average response times for each transaction class in a particular service class to determine which are missing their goals. In addition, you can view the transaction classes' hop detail to pinpoint the specific hop that may be the cause of the performance goal being missed.

#### **Process classes**

Specifies all of the process classes for each platform defined in the deployed domain policy. You can view performance statistics for each process class. You can use this view to determine if the operating system processes assigned to each process class are adhering to the performance goal defined. In addition, you can determine if the performance of the operating system processes for a specific platform are not meeting your performance goals.

#### **Partition classes**

Specifies all of the partitions defined in the deployed domain policy that EWLM is monitoring. You can use this view to determine if the work that a partition processes is meeting its performance goal.

#### **Managed servers**

Identifies all of the managed servers in the EWLM domain. You can view the status of each server and view data specific to each server. The managed server details include the services classes that are processed by the managed server. This is particularly useful if a service class is missing its goal due to the managed server not having enough resources. You can view the managed server's details to determine if the processor usage is acceptable for the number of transactions that it is processing.

#### **Load balancers**

Use this view to determine what weight recommendations EWLM sends to the external load balancers, if enabled. In addition, you can use this view to determine which application and server group members are participating in the load balancer group.

#### **Partitions**

Identifies the partitions in the EWLM domain that participate in a partition workload group. If partition management is enabled, this view allows you to identify which partitions EWLM is managing. You can use this view to examine partition information such as the number of processing units that the partition is entitled to use. In addition, you can view the partition weight. Use this information to view how EWLM is adjusting the processing power among the partitions in a partition workload group to ensure performance goals are met.

#### **Low-level views**

When you access a high-level view, you can select a low-level view in the **Select Action** field. Use the low-level views to obtain more detailed statistics.

#### **Details reports**

Specifies transaction and performance details for a specific report. The specific data contained in each details report depends on what type of report you are viewing. For example, you can view a Service class details report, Process class details report, or Managed server details report. Use these details to help you determine whether a problem exists with a specific managed server or a problem exists with work associated with a specific process or service class.

#### **Topologies**

Provides a graphical view of the managed server or applications in your EWLM domain. You can view an application topology or managed server topology. From the topology view, you can determine which servers are processing work for each application in the EWLM domain. In addition, you can view the amount of resources that each application or server is absorbing to process work requests that are processed in the EWLM domain. For each topology, you can also view more granular data such as the average response

times, number of delays, number transactions that completed successfully or unsuccessfully, and many other granular statistics. You can use all of this data to determine if there is a performance problem in your EWLM domain.

### Real-time performance monitors

Provides graphical views that EWLM updates automatically. Use these graphs to determine how work is processed on your EWLM domain relative to time. The following are specific monitor views that are available:

- Goal achievement monitor
- Processor utilization monitor
- Transaction rate monitor

You can use any combination of these views to examine the performance of your EWLM domain. The views that you use to solve a problem are specific to your environment. For example, if a service class is missing its goal, the transaction class details reports that are specific to that service class can be useful. These reports specify the number of transactions associated with the transaction class and how they are performing. Also, if there is a problem with a particular application or server, the server topology or application topology can be useful. These topologies provide details about each application or server in the EWLM domain.

In addition, when you open the monitor views, you can learn about the classification of work requests in your EWLM domain. You can set up your policy so that EWLM classifies all transactions to a service class other than the default. Then, if EWLM classifies a work request to the default, you can determine that the classification is not set up properly. To ensure EWLM classifies work requests properly, create a transaction class for each application that specifies a wildcard filter value and specifies a unique service class. You also need to verify that the application is set up and instrumented using ARM 4.0 properly.


The following table specifies the low-level views that are available from a high-level view. Use the **Select Action** field to access a low-level view that is specific to the high-level view.

Table 78. Specifies the low-level views that you can access from a high-level monitor view.

	Exceptions	Service classes	Transaction classes	Partition classes	Process classes	Managed servers	Load balancers	Partitions
Service class details	X	X						
Transaction class details			X					
Partition class details				X				
Process class details					X			
Managed server details						X		
Load balancer details							X	

Table 78. Specifies the low-level views that you can access from a high-level monitor view. (continued)

	Exceptions	Service classes	Transaction classes	Partition classes	Process classes	Managed servers	Load balancers	Partitions
Partition workload group details								X
Hop details	X	X	X		X			
Application topology	X	X	X					
Server topology	X	X	X					
PI index monitor	X	X						
Goal achievement monitor	X	X	X		X			
Transaction rate monitor	X	X	X		X			
Processor utilization monitor								X

Click  in the EWLM Control Center to learn more about the specific fields that display in these reports and monitors.

EWLM retains the most recent 24 hour period of data. You can save a history of the performance data by using the **Export** function in the EWLM Control Center. In addition, you can change the interval of data that EWLM displays by changing the **Interval**.

“Data collection and aggregation” on page 23

A valuable capability of EWLM is its ability to collect and aggregate performance data. This performance data collection capability allows EWLM to report to you if performance goals are met. In addition, it allows EWLM to make automatic resource adjustments if goals are not met.

### Examining performance data

EWLM displays performance data fields in the EWLM Control Center. The performance data fields described in this topic are the most common fields. In addition, the performance data fields described in this topic allow you to quickly determine if work is adhering to its performance goal.

**Note:** Click  to view the EWLM Control Center online help for details about other fields displayed in the EWLM Control Center.

The performance data that EWLM reports is specific to the time interval displayed. If the Control Center’s domain properties specify to collect data every 10 minutes, the data is cumulative for the entire 10 minute time interval. Longer time intervals include a greater number of work requests. Shorter time intervals include fewer work requests.

The following list identifies performance data fields that EWLM displays on many reports and monitors throughout the **Monitor** section of the EWLM Control Center.

| **Goal** Specifies the performance goal as defined in the service class. This is the goal that you *expect* the workload to meet. For more information on possible performance goals, see “Performance goals” on page 46

| **Performance**  
| Specifies the *actual* performance of the work.

| **Performance index**  
| Specifies a value that compares the expected **goal** to the actual **performance**. The index value indicates the degree to which the workload does or does not meet the workload’s performance goal. EWLM calculates the performance index without response time data for transactions that end in a failed state. Failed transactions can inaccurately skew the results if the transaction ends in an abnormally short or long time. See the following table for a list of possible values for a performance index. In addition, learn how to examine performance index values that indicate a performance problem might exist:

| *Table 79. Possible values for performance index.*

Performance index value	Description	Recommended action:
>1	Indicates that the goal is not met. The service class is reported in the Exceptions report because the performance goal is not met.	<p>You must examine the reports to determine the exact cause of the performance problem and take the appropriate action.</p> <ol style="list-style-type: none"> <li>1. View the service class details to determine which classes of work (transaction, process, or partition) use that particular service class performance goal. This helps you identify the specific work that uses the performance goal defined in the service class.</li> <li>2. Examine the goal achievement monitor for the service class to determine if the goal is missed consistently or if the goal is missed during a specific time interval where there is a peak in the workload.</li> <li>3. Examine the details report for each class of work that uses that service class performance goal.</li> <li>4. Examine the topology view. When examining the topology determine if a system has enough resources to handle the incoming work. In addition, determine if the number or stopped or failed transactions is acceptable compared to the total number of transactions processed. Does this number fall within an acceptable threshold?</li> </ol>
1	Indicates that the goal is met exactly.	No action necessary.
<1	Indicates that the goal is exceeded.	No action necessary.

Table 79. Possible values for performance index. (continued)

Performance index value	Description	Recommended action:
None	<p>Indicates that EWLM is not aware of any transactions processed. Therefore, EWLM can not calculate a performance index. If this occurs, verify that the classification rules properly identify the work that are to use this class. In addition, verify that the application or server that uses this class is actually processing work.</p> <p>In addition, a value of None can indicate that EWLM cannot calculate a value. For more information, see the Accuracy of data section in the troubleshooting topic “Troubleshooting: EWLM Control Center” on page 329.</p>	<ol style="list-style-type: none"> <li>1. View the service class details to determine which classes of work (transaction, process, or partition) use that particular service class performance goal. This helps you identify the specific work that uses the performance goal defined in the service class.</li> <li>2. View the classification rules in the domain policy to ensure that they are defined correctly.</li> <li>3. If monitoring application-level transactions, ensure that Application Response Measurement (ARM) instrumentation is enabled for the entry application. This is when EWLM classifies the work to a transaction class. For more information, see the troubleshooting topic, “Troubleshooting: Applications” on page 326.</li> </ol>
Not applicable	<p>Indicates that the class uses a discretionary goal. EWLM does not calculate a performance index. Discretionary goals are intended for low priority work that can run when resources are available. In this case, a performance index is not applicable.</p>	<p>If you want EWLM to calculate a performance index, you must change the performance goal in the service class to something other than a discretionary goal.</p>

## Monitoring application topology

View this topic to learn how to use the application topology to monitor your EWLM domain.

When you view a server topology for a particular service class, transaction class, or process class, EWLM provides you with a high-level view of the communication flow between the servers in the EWLM domain. The lines in the topology view indicate the flow of work requests from one server to another. In addition, you can view the ARM-instrumented applications on each managed server.

The server topology provides, but is not limited to, the following data:

- The flow of work requests between servers.
- The servers included in the topology.
- The details of each server that specify transaction, processor, and performance data.

When you view a server topology in the EWLM Control Center, you can display multiple views of the servers in the EWLM domain. To change the view, click the server icons to navigate to more specific details about the application topology. You can click between three views. The following screen capture is an example of a server topology. The first level icon (on the right) specifies the name of the server in the that EWLM is monitoring. The second level icon (in the middle) is clicked once and specifies the name and group of the application on the server. The third icon (on the left) is clicked twice and specifies the application instances on the server. Use these three different views to obtain more specific information about the managed servers in the EWLM domain. The server topology is arranged by hop. If a server has both a hop 0 and hop 1 applications running on it, that server will appear in both the hop 0 and hop 1 position. In the following example topology, localhost.server1 has hop 0, 1, and 2 applications running on it, so it appears in the topology in all three positions.



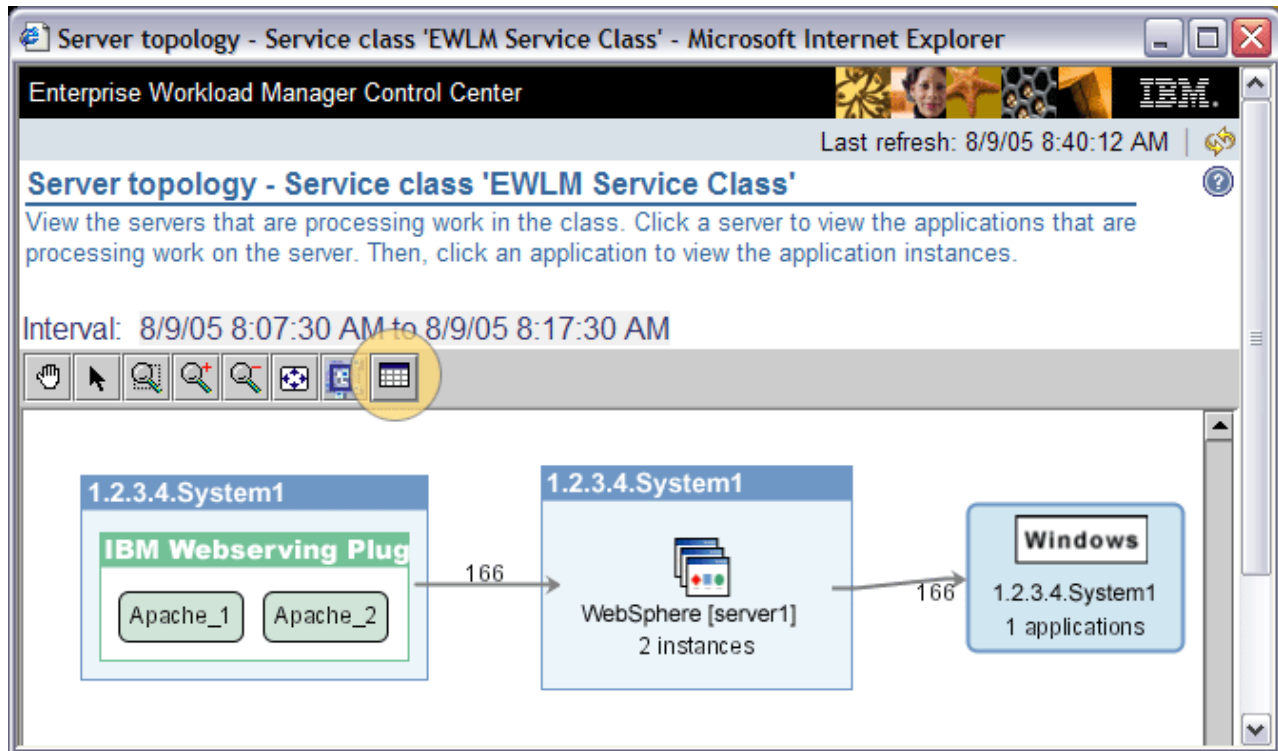


Figure 1. Example server topology that highlights the table icon. Use the table icon to view detailed statistics.

When you view a server topology, click the table icon to obtain more detailed statistics. The data specifies the performance of the work requests processed by the applications and servers. The data that EWLM displays in the table view is the same no matter if you access the table view from the application topology or server topology.

- | In addition, use the topology to determine which applications are *entry applications*. The entry application is the first application in the EWLM domain that processes the transaction. This is also referred to as hop 0. In addition, EWLM classifies work at hop 0. Therefore, you should create a transaction class for all applications that are viewed as entry applications.

## Monitoring server topology

View this topic to learn how to use the server topology to monitor your EWLM domain.

When you view a server topology for a particular service class, transaction class, or process class, EWLM provides you with a high-level view of the communication flow between the servers in the EWLM domain. The lines in the topology view indicate the flow of work requests from one server to another. In addition, you can view the ARM-instrumented applications on each managed server.

The server topology provides, but is not limited to, the following data:

- The flow of work requests between servers.
- The servers included in the topology.
- The details of each server that specify transaction, processor, and performance data.

When you view a server topology in the EWLM Control Center, you can display multiple views of the servers in the EWLM domain. To change the view, click the server icons to navigate to more specific details about the application topology. You can click between three views. The following screen capture is an example of a server topology. The first level icon (on the right) specifies the name of the server in the that EWLM is monitoring. The second level icon (in the middle) is clicked once and specifies the name and group of the application on the server. The third icon (on the left) is clicked twice and specifies the

application instances on the server. Use these three different views to obtain more specific information about the managed servers in the EWLM domain. The server topology is arranged by hop. If a server has both a hop 0 and hop 1 applications running on it, that server will appear in both the hop 0 and hop 1 position. In the following example topology, localhost.server1 has hop 0, 1, and 2 applications running on it, so it appears in the topology in all three positions.

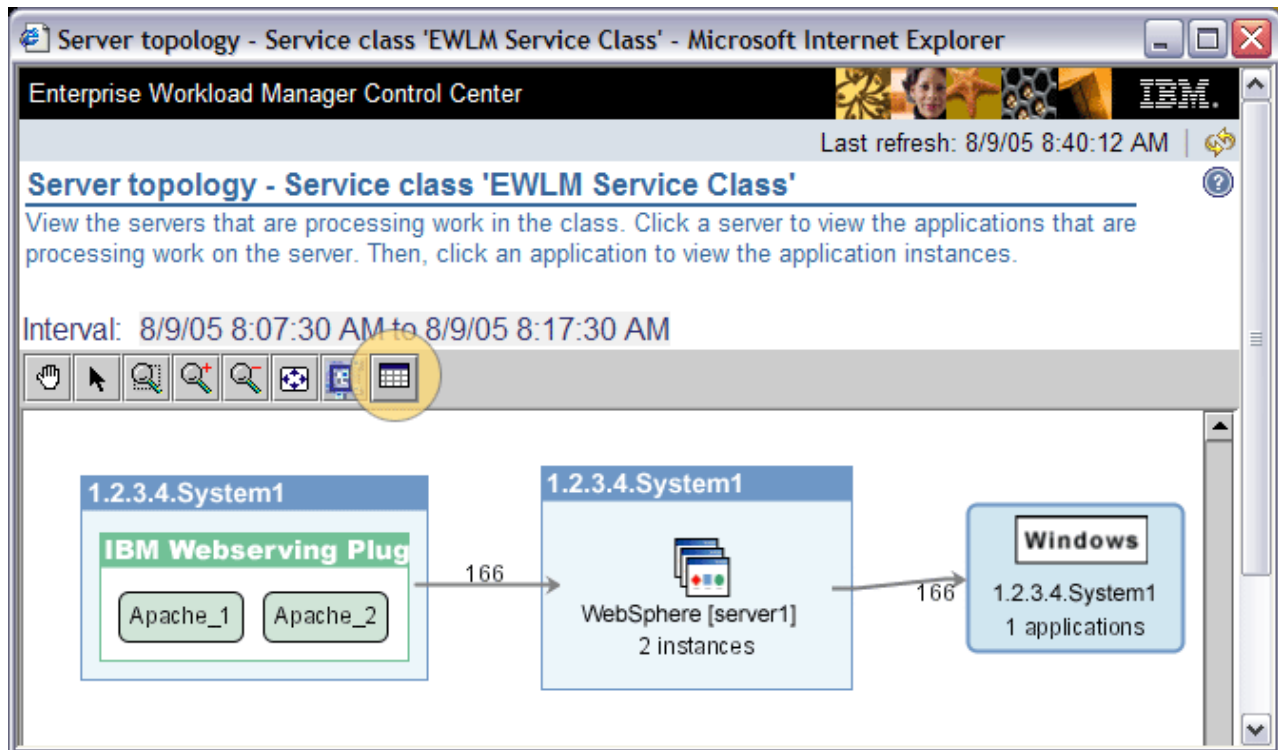


Figure 2. Example server topology that highlights the table icon. Use the table icon to view detailed statistics.

When you view a server topology, click the table icon to obtain more detailed statistics. The data specifies the performance of the work requests processed by the applications and servers. The data that EWLM displays in the table view is the same no matter if you access the table view from the application topology or server topology.

## Examining topology data

View this topic to learn about the topology data fields that EWLM displays in the Monitor section of the EWLM Control Center. More specifically, you can view this topology data from either an application or server topology. In addition, you can view similar data fields in a hop details report.

When viewing an application or a server topology, click the table icon to display granular performance data for each application, server, or instance in the EWLM domain. EWLM displays this data in a chart that resembles a table or spreadsheet format that you can export and save. If you access the topology data from either the server topology or the application topology, EWLM displays the same topology data. The topology data includes both the application and server information. However, the graphical topology views are different in that one provides an application view and the other provides a server view.

The data is specific to the time interval that you specify in the Control Center domain properties. If the domain properties specify to collect data every 10 minutes, the data is cumulative for the entire 10 minute time interval. Longer time intervals include a greater number of transactions; whereas, shorter time intervals include fewer transactions.

The following list identifies the topology fields to use when you examine your EWLM domain for performance problems:

**Hop** Specifies the sequence of an application, server, or instance that processes work requests that EWLM monitors. Typically, multiple servers or applications process work requests before they complete. Each time work requests moves to another application or server for processing a hop occurs. Hop 0 represents the start of a transaction, also referred to as work request. This is the entry application that classifies the work request. Subsequent applications or servers that process the work request represent as additional hops. View the statistics for each hop to determine if there is a problem with a particular application. In addition, you can use this field to determine what applications each hop includes. The applications at hop 0 classify work requests to service classes. This data is particularly useful if EWLM is not classifying transactions correctly because you can investigate the application that is responsible for classifying the work.

**Name** Specifies the name of the server, application, or instance that is processing work requests for a particular hop.

**Type** Specifies what is processing the work request. The type can be server, application, or instance. If the type is server, it must use a supported operating system. If the type is application, the application is ARM-instrumented. If you have multiple copies of an application or server in your domain, this is referred to as an instance. The instance processes transactions independently from the other applications and servers in the domain.

**Platform**

Specifies the operating system on which an application, server, or instance is processing transactions.

**Average response time**

Specifies the amount of time, on average, that the work requests processed on the server, application, or instance incur from the moment they enter the hop. The average response time includes all of the successfully completed transactions processed in a particular hop. When viewing this data, you must consider the type of transactions that the hop is processing. For example, the average response time may be directly related to the number of transactions being processed. In addition, the average response time may be directly related to the type of hop that is processing the work requests. For example, an HTTP server that processes static Web pages is likely to have a quicker response time compared to a WebSphere Application Server. Static Web pages can process transactions quickly, thus, driving down the overall average response time for the hop compared to other hops.

**Average active time**

Specifies how long, on average, the server or application takes to process a work request. A time interval that is long, relative to the expected goal, can indicate a problem. If so, view the details of that particular managed server. Active time does not include the time incurred when the application reports the work request in a blocked state or on the queue. An application blocks a request when the application is waiting for sub-transaction processing to complete.

**Average queue time**

Specifies the amount of time, on average, that incurs from the moment that the middleware application receives the work request to the moment it begins processing it. EWLM calculates the queue time by taking the time stamp from when the start command is run and subtracting the arrival time. The application records the arrival time when it receives the work request. If this value is high, it can indicate the middleware application is not able to start the work requests efficiently if a high volume of transactions are arriving.

**Successful transactions**

Specifies the number of transactions that complete successfully in the associated hop. Use this data to view the hops where transactions are processing without delay or failure. Compare the number of successful transactions to failed transactions to determine if the ratio of successful to failed transactions is acceptable.

### **Failed transactions**

Specifies how many transactions failed on each server or application for the specified hop. Compare this value to the service class's performance index. EWLM calculates the performance index using the successful transaction response times rather than the response times of stopped, successful, and unsuccessful transactions. This comparison allows you to determine if the goal is met because a significant number of transactions are failing. To investigate a stopped transaction further, view the application specific diagnostics to determine why the failure occurs.

### **Stopped transactions**

Specifies the number of transactions that stopped before completing. These transactions neither failed nor completed successfully. A transaction can move to the stopped state if it encounters an error with the application or server that is processing it. For example, if an application detects its caller or client terminates the request before the transaction instance completes, the application may abandon processing for the transaction instance and report it as stopped. The stopped state in EWLM corresponds to the ARM transaction completion status of aborted. To learn more about how the middleware or application reports the ARM status of a transaction, you need to review the documentation for your application.

### **Unknown transactions**

Specifies the number of transactions that did not end in a completed, failed, or aborted state for the associated hop.

### **Not valid transactions**

Specifies the number of transactions associated with this hop that ended in an abandoned state. EWLM does not include these transactions when calculating a performance index.

### **Topology uncertainty count**

Specifies the number of transactions whose correlator identified a condition that deviates from the norm from an EWLM point of view. For example, this might happen when a managed server receives a valid EWLM correlator from a preceding hop with an indication that the preceding hop is not in a valid EWLM domain or if ARM is active and the managed server is not running on the preceding hop. Use this field to determine if a managed server needs to be added to the domain, if EWLM needs to be started on a managed server, or if a server is processing work request which you did not expect. This field is usually expected to be zero. If this field contains a non zero value, you might not see the expected topology graphs for your workload. If this field is consistently increasing, view the **Managed servers** and verify that they are in the Active state.

### **Reporting gap count**

Specifies the number of transactions whose correlator identified a condition that deviates from the norm from an EWLM point of view. For example, this might happen when a managed server receives a valid EWLM correlator from a preceding hop with an indication that the preceding hop is not in a valid EWLM domain or if ARM is active and the managed server is not running on the preceding hop. Another cause for a reporting gap is if the preceding hop has an active service policy that differs from the active service policy of its predecessor hop. This can occur when a policy is the process of being activated on the domain. There is a time delay between the activating the policy on the first server in the domain to the last. This field is usually expected to be zero. If this field contains a non zero value, the transaction reporting data might not be accurate for the interval. If this field is consistently increasing, view the **Managed servers** and verify that they are in the Active state.

### **Processor time**

Indicates the actual cumulative amount of time, in seconds, that all of the transactions for the specific time interval incur on the processor. If this value is high and the **Managed server details** indicate that the **Processor utilization** is high, view this field in the topology data to determine which specific hop, application, or server has the highest processor time. Therefore, you can identify what specific hop, application, or server is contributing the most to the overall processor utilization of the managed server.

### Processor using percentage (%)

Specifies the percentage of time the server, application, or instance can use the processor when it requests to do so. A high number indicates that the processor is available when needed. For example, you can interpret a processor using percentage of 75% to mean that 75 out of 100 transactions could use the processor when they requested to use the processor.

If the using percentage is low and the **Processor delay** is high, the service class is having trouble processing the work requests. To investigate the processing using percentage further, view the **Managed server details** to determine the average processor utilization percentage. If both the **Average processor percentage** and **Average processor utilization** are high, one potential cause is that there are not enough resources to process the work efficiently.

### Processor delay percentage (%)

Specifies the percentage of non-idle time the work requests incur when they request to use the processor in the associated hop but are unable to do so. This field is particularly useful to determine if there is sufficient processing capacity to support the workload. This can indicate that the processor is too small or there are too many transactions on a particular server. If the processor delay is high for one particular hop, view the details of the managed server that is processing the work requests for that hop to determine what transaction and process classes that server processes. Perhaps too many transactions are trying to access the same processor. Also, if the processor delay is high for all instances of the server, application, or instance throughout the topology data, you can determine that the application does not have enough resources to process all of its transactions within the expected goal. In this case, you can consider adding additional CPU resources, moving work to a different server, or changing the goal.

### Page delay percentage (%)

Specifies the percentage of non-idle time the work requests in the associated hop incur while waiting for page faults to resolve. If the page resides in virtual memory, it is resolved rather quickly and a page fault does not occur. If a referenced page does not reside in processor accessible memory, the operating system retrieves the page from its hard disk drive (also referred to as paging space) before processing of a given transaction can proceed. This can take longer to process. In turn, this can increase the page delay. Page delays indicate how heavily the system's memory is being used. If this number is very low, it indicates that most transactions are finding the data they need in memory rather than having to go to the hard disk drive. If this number is high, it means that memory is not sufficient to keep most needed data around for the transactions and response time will probably be slower than expected.

### I/O delay percentage (%)

Specifies the percentage of non-idle time where I/O (input/output) delays the work requests in the associated hop. Input/output delay is the amount of time that a transaction incurs during non-idle time while waiting for I/O to complete. Compare this value to the **Processor delay %** and **Processor using %** values to determine if there is a consistent determination that the processors do not have enough resources to process the work requests efficiently.

### Idle delay percentage (%)

Specifies the percentage of time the processor is available but not processing any transactions. Use this value to determine if the processor is running idle for an appropriate amount of time. If transactions are not completing, goals are missed, and the processor using percent is high, you can use this field to determine if there is a problem with the processor because it is running idle too much.

**Other** Specifies the percentage of non-idle time that transactions incur that does not include **I/O delays**, **Page delays**, **Processor delays**, or **Processor using time**.

### In progress transactions

Specifies the number of transactions that are in progress. These transactions are not complete and are not included in performance calculations. Use this value to determine if an excess number of transactions are in progress. This field only applies to transaction or process type work.

## | In progress elapsed time

| Specifies the amount of time, on average, that the transactions in progress have incurred thus far. The elapsed time includes the amount of time incurred from the start of the transaction to the end of the reporting interval. This field only applies to transactions or process type work. A long elapsed time, relative to the average response time of completed work, can indicate a performance problem. Compare this value to the average response time of completed work to determine if work is not completing as expected. This field only applies to transaction or process type work.

| **Note:** A value of 0 can indicate that either the value is actually 0 or that no data is reported to EWLM. The cause of a 0 value may be due to a communication error between the domain manager and managed server.

| **Note:** A value of **None** is possible in the following fields:

- | • **Average active time, Average queue time, Average response time, Standard deviation:** None indicates that no transactions completed successfully. Therefore, EWLM does not have any data to report in this field.
- | • **Idle %:** None indicates that no transactions were reported. Therefore, EWLM does not have any data to report in this field.
- | • **In progress response time:** None indicates that no transactions are in progress. Therefore, EWLM does not have any data to report in this field.
- | • **I/O delay %, Other delay %, Page delay %, Processor delay %, Processor using %:** None indicates no transactions were reported. Therefore, EWLM does not have any data to report in this field.

| **Note:** A value of **Not available** indicates that the operating system does not support nor supply EWLM with the data. This value is possible for the following fields: **Idle delay %, I/O delay %, Other %, Page delay %, Processor delay %, Processor using %**.

## | Application-specific considerations

### DB2 Universal Database

If you notice DB2 transactions in hop 0, the cause can be that the DB2 Universal Database health monitor agent is enabled. If the DB2 Universal Database health monitor agent (db2hmon) is active, the monitor will communicate with the DB2 instance to monitor its health. The protocol between the health monitor agent and the DB2 instance does not exchange ARM correlators. Therefore, EWLM presumes DB2 ARM calls as being from a hop 0 DB2 instance when the health monitor calls DB2. Therefore, the health monitor initiated transactions to the DB2 instance appear in EWLM reporting as hop 0, the edge hop. Typically, DB2 Universal Database transactions will not appear in hop 0. If the health monitor agent is not enabled, these transactions will not appear in EWLM reporting. Other analogous circumstances can yield similar results, DB2 in hop 0.

## | Platform-specific considerations

### HP-UX

| EWLM includes HP-UX processes that are not doing any work in the **Other** category. Typically, this work is included in the **Idle** category; however, HP-UX does not support the **Idle** category. Consequently, the percentage in the **Other** category for HP-UX can be significantly higher since it includes processes in the idle state as well.

## Formulas used to calculate topology statistics:

View this topic to learn about how EWLM calculates the processor related topology statistics.

The formulas use the following variables:

- *Pu\_samples*: Total process using samples collected over interval.

- *Pd\_samples*: Total processor delay samples collected over the interval.
- *PGd\_samples*: Total page delay samples collected over the interval.
- *IOd\_samples*: Total I/O delay samples collected over the interval.
- *Idle\_samples*: Total idle samples collected over the interval.
- *other\_samples*: Total other samples collected over the interval.

$$\text{Idle \%} = ((\text{Idle\_samples}) / (\text{Pu\_samples}) + (\text{Pd\_samples}) + (\text{PGd\_samples}) + (\text{IOd\_samples}) + (\text{other\_samples}) + \text{Idle\_samples})) \times 100$$

$$\text{Processor using \%} = ((\text{Pu\_samples}) / (\text{Pu\_samples}) + (\text{Pd\_samples}) + (\text{PGd\_samples}) + (\text{IOd\_samples}) + (\text{other\_samples})) \times 100$$

$$\text{Processor delay \%} = (((\text{Pd\_samples}) / (\text{Pu\_samples}) + (\text{Pd\_samples}) + (\text{PGd\_samples}) + (\text{IOd\_samples}) + (\text{other\_samples})) \times 100$$

$$\text{Page delay \%} = (\text{PGd\_samples}) / (\text{Pu\_samples}) + (\text{Pd\_samples}) + (\text{PGd\_samples}) + (\text{IOd\_samples}) + (\text{other\_samples})) \times 100$$

$$\text{I/O delay \%} = ((\text{IOd\_samples}) / (\text{Pu\_samples}) + (\text{Pd\_samples}) + (\text{PGd\_samples}) + (\text{IOd\_samples}) + (\text{other\_samples})) \times 100$$

$$\text{Other delay \%} = ((\text{other\_samples}) / (\text{Pu\_samples}) + (\text{Pd\_samples}) + (\text{PGd\_samples}) + (\text{IOd\_samples}) + (\text{other\_samples})) \times 100$$

## Monitoring partitions in EWLM

View this topic to learn how you can monitor the performance of work that a partition processes.

Prerequisites:

- Partition classes in the domain policy identify the partitions that EWLM is to monitor.
- EWLM partition management is enabled if you want EWLM to manage the amount of processing power that each partition can use.

To monitor a partition in the EWLM Control Center, complete the following steps:

1. Log in to the EWLM Control Center.
2. In the EWLM Control Center, view the partition class reports found in **Monitor** → **Partition classes**. For a detailed list of reports and descriptions, see Table 80.
3. If EWLM partition management is enabled, view the partition reports found in **Monitor** → **Partitions**. For a detailed list of reports and descriptions, see Table 80.

*Table 80. EWLM Control Center views to use when monitoring the partitions in your EWLM domain.*

Monitor view	Description
Partition classes report	Displays the performance index for each partition class in the domain policy. This view contains a comprehensive list of partition classes.
Service classes report	Indicates if the work associated with the service classes are meeting their performance goals. View the service classes that contain work of the type 'partition' to determine if the partitions meet or exceed their performance goals.

Table 80. EWLM Control Center views to use when monitoring the partitions in your EWLM domain. (continued)

Monitor view	Description
Partition workload groups	<p>Provides the processor utilization percentage of the shared processors that the partitions in the group are actually using. This view is only available if EWLM partition management is enabled.</p> <p><b>Note:</b> If a partition moves from one partition workload group to another, EWLM might report the partition in both partition workload groups for a short period of time.</p>
Partition workload group details report	<p>Allows you to view a list of partitions that belong to the partition workload group. In addition, you can view the number of processing units that each partition is entitled to use.</p> <p>If EWLM partition management is enabled, EWLM automatically adjusts the following fields:</p> <p><b>Processing units</b> Specifies the number of processors the partition is entitled to use from the shared processor pool on the machine. If EWLM determines that it needs to take processing power from the shared processing pool and allocate it to the partition, EWLM adjusts the value displayed in this field.</p> <p><b>Weight</b> Specifies a number in the range of 0 through 255 that EWLM sets for each uncapped partition in the shared processor pool. Any available unused capacity is distributed to contending logical partitions in proportion to the weight of each of these partitions.</p> <p>This view is only available if EWLM partition management is enabled.</p> <p>For an example of this report, see Partition Workload Group Details report.</p>
Server topology	<p>Provides high-level view of all of the servers that are processing work requests in your EWLM domain. This allows you to view the number of transactions that flow from one server or partition to the next. In addition, allows you to pinpoint the exact location of a performance bottleneck, if one exists.</p>



Example

Partition Workload Group Details

Enterprise Workload Manager Control Center

Last refresh: 2/15/06 10:00:55 AM

### Partition Workload Group Details

View the details of each partition in the partition workload group.

Interval: 2/15/06 9:59:30 AM to 2/15/06 10:00:30 AM

#### Statistics

Partition name ▲	Weight	Virt process	Phys process	Process utilization	Process units	Cap	Minimu process units	Maxim proces units
partition1_AIX	118	1	8	6.08%	0.65	No	0.10	4.00
partition2_linux	18	1	8	0.52%	0.10	No	0.10	6.00
partition1_linux	219	3	8	17.42%	1.20	No	0.10	6.00
partition2_AIX	43	1	8	0.83%	0.24	No	0.10	4.00
partition3_AIX	217	3	8	0.68%	1.19	No	0.10	4.00

Page 1 of 1      Total: 5   Filtered: 5   Displayed: 5

#### Processor utilization distribution chart

Partition

- partition1\_AIX
- partition2\_AIX
- partition3\_AIX
- partition2\_linux
- partition1\_linux

Close    Export

|  
| This screen shot is an example of the partition workload group details report. View the online help for  
| more details about each field. The following is a brief overview of the most useful fields in this report:

| **Statistics - Processing units**

| Indicates the number of processing units that the partition is entitled to use. If EWLM makes a  
| resource adjustment, this field is updated to reflect the change. Use this data to compare the amount  
| of processing power that each partition in the partition workload group is using. This helps you  
| identify where peak workloads exist.

| **Statistics - Weight**

| Determines how to distribute unused capacity to contending logical partitions. Unused capacity is  
| distributed to contending logical partitions in proportion to the relative value of the partitions'  
| weights. EWLM adjust this value when it make a resource adjustment for a partition.

| **Statistics - Processor utilization**

| Indicates the percentage of the partition's processor allocation (from the shared processor pool) that  
| the partition is using. For example, partition1\_AIX is using 6.08% of the nine processors available for  
| it to use. EWLM has adjusted the partition to .65 processing units but it is not using all of the .65  
| units.

| **Processor utilization distribution chart**

| Indicates what percentage of processing power consumed by the partition workload group is  
| consumed by each specific partition. This chart shows the distribution of used processing power  
| among the partitions in the partition workload group.

| "Partition classes" on page 50

| A *partition class* identifies partitions that are to be managed with a single goal and business  
| importance. The partition class is associated with a service class that contains the goal and business  
| importance. You use partition classes as part of EWLM's partition workload group management.  
| EWLM manages the CPU resources based on the overall performance of partitions in a partition class.

| "Enabling partition management" on page 249

| Use this topic to learn how to enable logical partition management in EWLM.

| **Monitoring EWLM environments that use a load balancer**

| View this topic to learn how you can monitor your EWLM environment if you have EWLM load  
| balancing support enabled.

| **Prerequisites:**

- | • The EWLM domain includes a load balancer, which is an external device that routes incoming work to  
| its members.
- | • The members of the load balancer group are defined as EWLM managed servers.
- | • The domain policy defines performance goals for the EWLM managed servers. For more control,  
| ensure that the domain policy contains specific performance goals for each member of the domain.
- | • EWLM load balancing is configured and enabled.

| To monitor a load balancer group in EWLM, complete the following steps:

- | 1. Log in to the EWLM Control Center.
- | 2. Navigate to **Monitor** → **Load balancers**.
- | 3. Select a load balancer group so that it is highlighted.
- | 4. Click **Details** from the **Select Action** field.

## Example

The screenshot displays the 'Load balancer details - 'loadbalancer1'' page in the Enterprise Workload Manager Control Center. The page title is 'Load balancer details - 'loadbalancer1'' and the subtitle is 'View the details of each group within the load balancer'. The page is last refreshed on 2/15/06 at 9:58:36 AM. The main content area is titled 'Load balancer groups' and includes a table with the following data:

Select	Load balancer group ^	Group type ^	Group Load balancer type ^
<input checked="" type="radio"/>	group1	Application	Application

Page 1 of 1 Total: 1 Filtered: 1 Displayed: 1 Selected: 1

The page also includes a section for 'Application group members' with the subtitle 'Members of application group 'group1''. This section contains a table with the following data:

Server name	IP address ^	Port	Protocol	Application	Weight	Average response	Transa count
system1	1.2.3.4	80	TCP	IBM Webseving	59	00.015	5,292
system2	1.2.3.4	9080	TCP		59	00.000	0
system3	5.6.7.8	80	TCP	IBM Webseving	59	00.014	4,466
system4	5.6.7.8	9080	TCP		59	00.000	0
system5	9.10.11.12	80	TCP	IBM Webseving	49	02.024	4,203
system6	9.10.11.12	9080	TCP		51	00.000	0

Page 1 of 1 Total: 6 Filtered: 6 Displayed: 6

The browser address bar shows 'http://www.ibm.com/us' and the page is titled 'Enterprise Workload Manager Control Center'.

This screen shot is an example of the load balancer details report. View the online help for more details about each field. The following is a brief overview of the most useful fields in this report:

### Load balancer groups

Defines a list of load balancer groups that exist in the EWLM domain. Select a group from the list to view its members and whether the members are applications or systems.

### Application | System group members

Indicates which members belong in the load balancer group. Use this data to identify the members to which the load balancer routes work.

### Application | System group members - Weight

Indicates the degree to which the load balancer is to distribute incoming work to the member. A high value indicates to route work to the member. A low value indicates to divert work from the member. Possible values range from 0 to 100. Use this data to determine the member's weight value relative to

the weight value for all members in the group. In addition, you can use this data to identify the members from which the load balancer is diverting work. Use the EWLM Control Center reports and monitors to investigate these members further and to identify if a performance problem exists.

“EWLM load balancing” on page 23

Load balancing is an optional function of EWLM that is valuable if you want EWLM to send load balancing recommendations to a load balancer. This function allows the load balancer to evenly distribute incoming work to ensure that performance goals are met.

“Enabling EWLM load balancing” on page 248

Use this topic to learn how to enable EWLM to communicate with a load balancer.

## Performance statistics logging

Learn how to configure EWLM to automatically log the performance statistics that it collects. This function provides EWLM with the ability to store EWLM performance statistics in long term storage; whereas, the EWLM Control Center provides an interface to graphically render and export up to 24 hours worth of performance statistics.

Use the performance statistics logging function to store the performance statistics in the form of records in a log file on the domain manager’s file system. To enable this function, complete the following:

1. Ensure that you have installed the latest EWLM domain manager and managed server fix packages.
2. Run the EWLM startDM command to start EWLM on the domain manager. EWLM creates the DataHardeningPlugin.xml plug-in descriptor file when this command is run.
3. Configure and enable the EWLM performance statistics logging plug-in on the domain manager system. To do this, complete the following:
  - a. Go to `EWLMDataDirectory/ewlmdm/servers/configID/Plugins/DataHardeningPlugin.xml` where the variables are defined as the following:

### **EWLMDataDirectory**

The path for the EWLM domain manager data. This is the path that is defined during the domain manager installation.

### **configID**

The configuration ID for the EWLM domain manager. This is the configuration ID that you defined when the EWLM domain manager was configured.

**Note:** If the plug-in descriptor does not exist, you may not have the latest EWLM domain manager fix package installed. Ensure you do, and then rerun the EWLM startDM command.

- b. Open the plug-in descriptor using an editing tool of your choice.
- c. Edit the plug-in descriptor. For more information on the plug-in descriptor’s parameters and possible values, see “Performance statistics logging plug-in descriptor.”
- d. Save the plug-in descriptor file.

## Performance statistics logging plug-in descriptor:

Learn about the parameters and possible values that you can configure in the EWLM performance statistics logging plug-in descriptor. This plug-in descriptor controls how the EWLM performance statistics logging plug-in functions.

Table 81. EWLM performance statistics logging plug-in descriptor parameters and possible values.

Parameter	Possible values	Default value	Description	Change requires plugin restart
<b>RefreshablePluginState</b>	<ul style="list-style-type: none"> <li>Enabled</li> <li>Disabled</li> </ul>	Disabled	<p>Indicates if the EWLM performance statistics logging plug-in is enabled or not. If enabled, EWLM exports the performance statistics that it collects into a file that resides on the domain manager's local file system.</p> <p>Specify <b>Enabled</b> to indicate that EWLM is to output the performance statistics into a file. An enabled state indicates that the plug-in is active.</p> <p>Specify <b>Disabled</b> if you do not want EWLM to output the performance statistics into a separate file.</p>	No
<b>OutputFormat</b>	<ul style="list-style-type: none"> <li><b>SER</b>: Outputs the performance statistics in the Java serializable format.</li> <li><b>XML</b>: Outputs the performance statistics in XML format that uses tags to separate data.</li> <li><b>OLC</b>: Outputs the performance statistics in a binary file that contains multiple components. The components are identified by a triplet with the offset, length and count. This output format is similar to the SMF format available on z/OS.</li> </ul>	XML	<p>Sets the output format of the data. See "Performance statistics logging output formats" on page 304 for more information about the different output formats.</p>	Yes

Table 81. EWLM performance statistics logging plug-in descriptor parameters and possible values. (continued)

Parameter	Possible values	Default value	Description	Change requires plugin restart
<b>OutputEncoding</b>	<ul style="list-style-type: none"> <li>UTF-8</li> <li>UTF-16BE</li> <li>UTF-16LE</li> </ul>	UTF-16BE	Sets the output encoding of all strings.	Yes
<b>OutputFilePrefix</b>	<i>userDefined</i>	EWLMReportData	Sets the filename prefix of the output file. This value combined with a unique number and a timestamp uniquely identifies an output file.	No
<b>DestinationDir</b>	<i>userDefined</i>	<i>EWLMDataDirectory/ewlmdm/servers/configID/Diagnostics</i>	Indicates the location on the domain manager's local file system where EWLM will store the output files.	No
<b>DeltaMode</b>	<ul style="list-style-type: none"> <li><b>True:</b> Indicates that EWLM will only output performance statistics for the last interval of time.</li> <li><b>False:</b> Indicates that EWLM will output all performance statistics since the most recent time that any of the following actions occurred: <ul style="list-style-type: none"> <li>Start EWLM on the domain manager</li> <li>Start EWLM on the managed server</li> <li>Activated a service policy</li> </ul> </li> </ul>	False	Indicates if the output file contains the last interval of performance statistics or all of the performance statistics. The interval of time is specified in the <b>IntervalLength</b> parameter.	Yes
<b>IntervalLength</b>	Any integer greater than or equal to 5. The unit is specified in minutes.	5	Indicates the interval of time for data collection. The performance statistics data is aggregated for the interval of time specified. The minimum value is 5 minutes.	No

Table 81. EWLM performance statistics logging plug-in descriptor parameters and possible values. (continued)

Parameter	Possible values	Default value	Description	Change requires plugin restart
<b>FileSizeLimit</b>	Any integer greater than 0 or the string value NO LIMIT.	NO LIMIT	Indicates the maximum size of the file that contains the performance statistics. The output file may exceed this value by no more than the size of one record. If the value is set to NO LIMIT, the output file has no maximum size per file.	No
<b>RecordsPerFile</b>	Any integer greater than 0 or the string value NO LIMIT.	1	Indicates the maximum number of records to write to an output file. If the value is set to NO LIMIT, the output file has no maximum number of records per file.	No
<b>MaxFileCount</b>	Any integer greater than 0 or the string value NO LIMIT.	NO LIMIT	Indicates the maximum number of output files. If the maximum is reached, the output file with the oldest time stamp will be deleted. If the value is set to NO LIMIT, no limit will be imposed on the number of output files in the output directory.	No
<b>OLCTraceEnabled</b>	<ul style="list-style-type: none"> <li>• True</li> <li>• False</li> </ul>	False	This parameter and value only applies if the output file format is OLC. If this value is set to True, EWLM creates a trace file for each OLC output file. The filename of the trace file is the same as the OLC output file appended with "_trace". The trace file resides in the location specified by the <b>DestinationDir</b> parameter.	No

## Sample

```
<Plugin Type="Runtime">
<Implementation Class="com.ibm.wlm.dh.DataHardeningPlugin">
<Origin origin="IBM">
<Arguments
RefreshablePluginState="Enabled"
OutputEncoding="UTF-16BE"
OutputFilePrefix="ReportData_PluginInstance"
DestinationDir="C:/ProgramFiles/IBM/VE2/EWLM/domain1/HardeningOutput"
DeltaMode="False"
IntervalLength="5"
FileSizeLimit="NO LIMIT"
RecordsPerFile="1"
MaxFileCount="NO LIMIT"
TraceEnabled="False">
</Plugin>
```

## Performance statistics logging output formats:

The performance statistics logging plug-in supports three output formats, each intended for different audiences, for the hardening of the EWLM performance data: XML, Offset Length Count (OLC) format, and Java Serializable (SER) format.

The following comparison provides information and guidance on which format to use.

Table 82. Performance statistics logging output format comparison

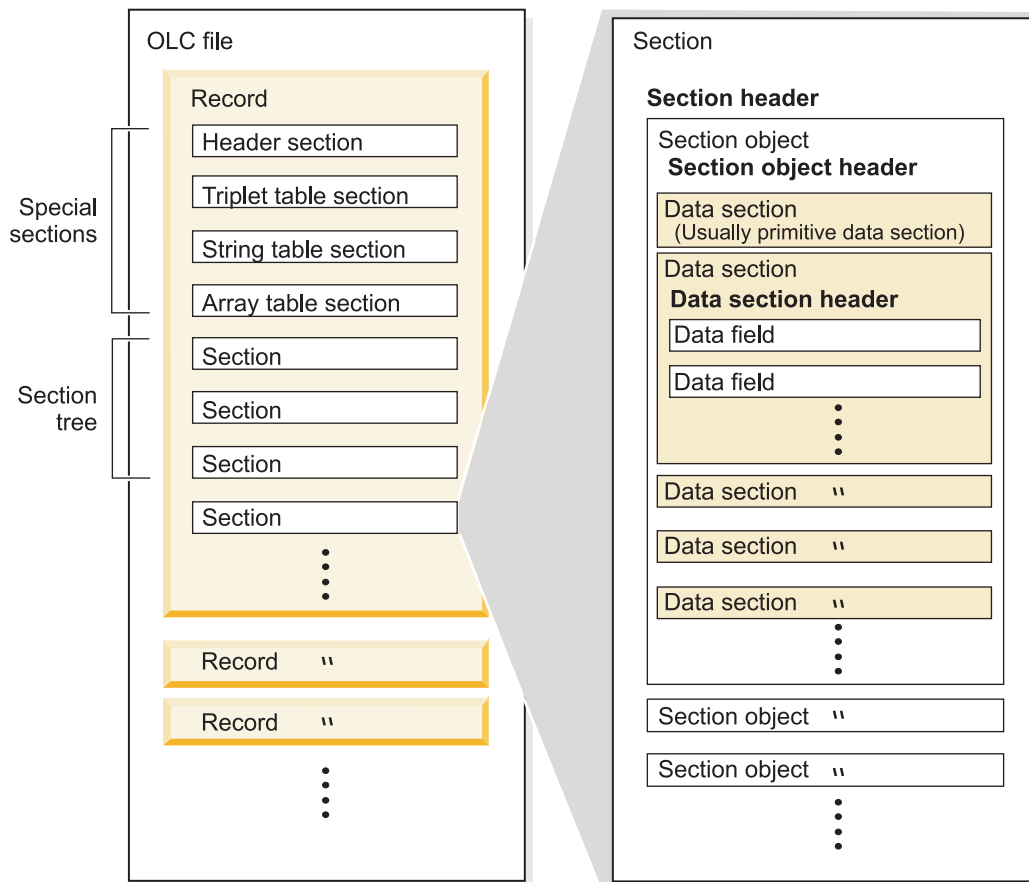
	XML	OLC	SER
<b>Output format</b>	Text in XML	Binary (similar to the SMF format on z/OS)	Binary in Java Serializable Object format
<b>Required tools</b>	Text editor, XML viewer, custom application with XML parser	Parsing application written in any computer language	Java application for parsing
<b>Complexity of parsing</b>	Low (if using an existing XML viewer)	High	Medium
<b>Complexity of analysis</b>	Medium	Medium	Low
<b>Storage requirement</b>	High	Low	Medium
<b>Domain manager processor overhead</b>	High	Low	Medium

The OLC and SER formats are discussed in detail in this topic.

## OLC output format

The OLC format hardens the EWLM performance to disk in a compact binary format that is independent of a specific platform or language. The following graphic illustrates the relationships between and within each of the sections of the OLC format output file.





EICAW504-0

There are several different levels of sections that contain performance statistics in the OLC format file.

### Record

An OLC output file may contain multiple OLC records with each record containing performance statistics captured over a specific time period.

### Section

Performance data is organized into distinct sections, with each section containing a logical subset of the data. For example, the `WLMDataServiceClass` section that is found in all OLC records contains the performance statistics for all service classes defined in the service policy that was active when the data was captured.

### Section object

The performance data contained in an OLC section is further organized into section objects, which contain individual instances of the data type contained in the section. For example, consider an EWLM domain policy that has four service classes defined: `ServiceClassA`, `ServiceClassB`, `ServiceClassC`, and `ServiceClassD`. The `WLMDataServiceClass` section, which contains the performance statistics for all service classes defined in the service policy, will contain four section objects--one for each of the service classes declared in the service policy. Section objects always begin with the `Size_of_object` field which defines the size in bytes of the performance data stored in the section object.

### Data section

Performance statistics for a section object are further divided into data sections, which contain the types of performance data contained in a section object. For example, a data section for a section object may contain performance data in the form of transaction response time statistics, while another

may contain performance data in the form of resource statistics. Section objects can contain multiple data sections in any order, and individual performance data statistics are contained in data fields of the data sections.

There are four special sections contained in an OLC record that do not report performance data. These sections instead contain meta-information about the OLC record and control data that assists navigation of the data contained in the OLC record.

**Header section**

The header section stores meta-information about the data contained in an OLC record. It provides useful information such as the size of the OLC record, the encoding of all string data contained in the OLC record, and the version and release information of the EWLM domain manager that produced the performance data.

**Triplet table section**

The triplet table section serves as a table of contents for the various sections contained in an OLC record. For each section, it contains the following:

- The binary offset of the section in the OLC record
- The size in bytes of the performance data for the section
- The number of section objects contained in the section.

The information contained in the triplet header allows a parser to randomly access any section in the OLC record without the need to sequentially parse through the data contained in all previous sections.

**String table section**

The HardenableStringTable (string table) section serves as a string table for all string data contained in the performance data of an OLC record. All performance data fields that are string values contain an entry in this section, which stores the string data and metadata information on the string. Each string entry in this section consists of the following:

- The size of the string in the encoding specified in the OLC Header section
- The number of characters contained in the string
- The actual string data in bytes.

For example, consider the WLMPolicyInfo section, which contains the data that describes the active service policy in effect when the OLC record performance data was captured. Its domain policy name data field contains an offset to an entry in the HardenableStringTable section that contains the domain policy name string. This offset is a 4-byte integer relative to the beginning of the HardenableStringTable section.

**Array table section**

The array table section serves the same purpose as the HardenableStringTable section, except that it contains array data instead of string data. All performance data fields that are array values contain an entry in this section, which stores the array data and metadata information on the array. Each array entry in this section consists of:

- The size of the array in bytes
- The number of array elements
- The array element type as a byte value with the following designations:

*Table 83. OLC output array element type byte values and designations*

Byte value	Designation
0	Integer
1	Short
2	Byte
3	Long

Table 83. OLC output array element type byte values and designations (continued)

Byte value	Designation
4	Float
5	Double
6	Boolean

- The actual array data in bytes.

A custom parser is required to extract the performance data that is stored in the OLC output file. To aid in the creation of a custom parser, as well as serve as an index to all the performance data captured in an OLC record, a document containing the format of the OLC output file performance data is provided. See OLC format mapping document for the document.

The OLC format mapping document outlines the data fields contained in each OLC section and specifies the following information for each data field:

Table 84. OLC format mapping document data field descriptions

Data field column number	Description
1	Decimal offset in bytes of the data field in its section or data section
2	Hexadecimal offset in bytes of the data field in its section or data section
3	Data field name
4	Decimal size in bytes of the data contained in the data field
5	Data field type (for example, short, int, or long)
6	Description of the reporting object that returned the data contained in the data field (not applicable for section headers)
7	Description of the reporting API that returned the data contained in the data field (not applicable for section headers)

All data fields contained in a section are relative to the beginning of the section except for data fields contained in a data section which are relative to the beginning of the data section. As mentioned previously, data sections contain the different types of performance data that can be contained in a section object. These data sections can be identified by a special control field, Data\_Section\_Format which serves to identify a data section in a section object of a binary OLC output file. The index of a data section in a section of the OLC format mapping document serves as its Data\_Section\_Format number and is stored in the Data\_Section\_Format field of a data section in a binary OLC output file.

For example, the WLMDDataServiceClass section contains seven data sections. Its first data section, the Primitive Data Section, has a Data\_Section\_Format number of 0. The format of a WLMDDataServiceClass section object that is contained in an OLC output file might look like this:

```
Size_of_object 4 bytes
Parent_Node_Offset 4 bytes
Parent_Node_Key 8 bytes
Eye_Catcher 2 bytes
Number_of_Data_Sections 2 bytes
Size_of_Primitive_Data_Section 4 bytes
Data_Section_Format 2 bytes
UniqueNodeId 8 bytes
Name 4 bytes
Key 8 bytes
Size_of_Object_Data_Section 4 bytes
```

Data_Section_Format	2 bytes
BenchMark	4 bytes
SampleCnt	4 bytes
TotalTime	8 bytes
TotalTimeInMicrosec	8 bytes
Version	4 bytes
BucketArray	4 bytes

| The format of the section object shown above has two data sections, both beginning with the size of the data section field. To read and format the data sections, the first twenty bytes must be skipped to remove the section object header. The next two data fields, referred to as the data section header, identify the size and format of the data section that is to follow. This information in the data section header is contained in every data section in the OLC format and is critical to correctly parsing and extracting the data section performance data. In the example shown above, the data section format number read from the binary data of this section object will be zero, signifying the first data section declared in the WLMDDataServiceClass section. Once the data section format number is known, you can consult the OLC format mapping document to identify the type and size of each performance data field contained in the data section. Continuing the earlier example, once the data section format number has been identified as zero, a parser can then know that the data section contains the three performance data fields UniqueNodeId, Name, and Key.

| Data fields contained in an OLC section are of two types:

| **Control fields**

| Control fields have a data field type beginning with the prefix OLCTYPE, and they contain data that assists in the navigation of the performance data contained in the section. Examples of control field types are:

- | • OLCTYPE\_LOS, which represents a data field that stores the size of the section in bytes.
- | • OLCTYPE\_SOO, which represents the size of a section object in an OLC section.

| **Performance data fields**

| Performance data fields have traditional types such as Short, Integer, or Long, and contain actual EWLM performance data.

| **SER output format**

| The SER format is also known as the Java Serializable Object Format. This format outputs the performance statistics encapsulated in various Java objects using the object serialization mechanism that is built into Java.

| There are three steps required to use the performance statistics outputted in SER format:

- | 1. Extract a performance data record from an output file.
- | 2. Extract the desired performance statistics from the record.
- | 3. Post-process the extracted performance statistics for capacity planning, charting etc.

| Each outputted performance data record is of the type SerializableHardenableDataExtendedHeader. The sample code below demonstrates how you can extract performance data records from a SER output file:

```
|
| FileInputStream fis = new FileInputStream(fileName);
| ObjectInputStream ois = new ObjectInputStream(fis);
|
| try
| {
| /* Read the input file until we hit an EOFException. */
| while (true)
| {
| try
| {
| SerializableHardenableDataExtendedHeader ewlMData
```

```

| = (SerializableHardenableDataExtendedHeader)
| ois.readObject();
| /* process the ewlmData */
| }
| catch (ClassNotFoundException e)
| {
| /* Probably the classpath did not include ewlm_interfaces.jar
| * that was shipped with EWL Managed Server */
| }
| }
| }
| catch (EOFException e)
| {
| /* End of File is reached */
| }

```

The `SerializableHardenableDataExtendedHeader` object represents a performance data record and contains the different types of performance data statistics that EWL Managed Server supports. The following types of performance statistics are currently supported:

**Statistics for transactions, processes, and the managed server**

These statistics reside in the `WLMReportData` object that is returned by invoking the `getWlmReportData` method.

**Statistics for EWL load balancing support**

These statistics reside in the `WMLBReportData` object that is returned by invoking the `getWlmLbReportData` method.

**Statistics for EWL Virtual Server support**

These statistics reside in the `WLMVGReportData` object that is returned by invoking the `getWlmVgReportData` method.

See the `SerializableHardenableDataExtendedHeader` API for a list of the performance data types currently supported by `SerializableHardenableDataExtendedHeader`, as well as documentation for the structure of the performance data encapsulated by the `WLMReportData`, `WMLBReportData`, and `WLMVGReportData` classes.

**Related information**

OLC format mapping document

The OLC format mapping document lays out the sections of the performance statistics logging OLC format output and aids in the creation of a tool that parses the output.

Reporting APIs

## Determining where performance problems reside

View this topic to read and interpret example performance data. Use these examples to learn ways to use the reports and monitors in EWL Managed Server that allow you to pinpoint the exact cause of performance problems that may exist in your domain.

When you monitor the performance of your EWL Managed Server domain, EWL Managed Server provides a variety of methods to help you determine the cause of a problem. These methods are available from the Monitor section of the EWL Managed Server Control Center. You will need to use a combination of these views to examine the performance of your EWL Managed Server domain. Each report, monitor, and topology provides performance data that is specific to your EWL Managed Server domain. The specific views that you use depend on your particular situation. You will need to determine which views will provide the most valuable information for the problem that you are investigating. This topic provides you with some example situations and ways to use the Monitor views to investigate the problem.

The first step in determining performance problems in your EWLM environment is to read and interpret the data in the high-level monitor views. These views provide you with high-level performance data and indicate whether a problem exists. These are the views that you can select in the left pane of the EWLM Control Center in the Monitor section.

After reviewing the high-level monitor views, you determine that there might be a performance problem that requires further examination. Use the following reports, monitors, and topologies to obtain more detailed information about the data:

| **Service class details**

| View this report to determine the number of transactions that completed successfully, unsuccessfully, failed, and so on before completing. In addition, you can view a list of classes that contain work that use the service class. For example, one or more transaction classes may use the same service class, performance goal. Use this report to compare the actual performance of the work to the service class goal.

| **Transaction class details**

View this report to examine the transaction states for the transactions that are classified to each transaction class. In addition, this report specifies what service class performance goal the transactions identified in the transaction class are to meet.

| **Process class details**

View this report to determine which managed servers are processing work requests for a specific process class. If the managed servers specified are not correct, edit the rules in the process class to filter the work requests properly.

| **Partition class details**

| View this report to examine performance data for all of the work that a partition processes.

| **Managed server details**

View this report to determine the managed server's processor utilization. In addition, you can identify which services classes contain work that the server is processing.

| **Load balancer details**

View this report to learn more about any load balancers in your environment, and the weight recommendations EWLM sends to for each member of the load balancer group. The EWLM load balancing is an optional function that you can enable.

| **Hop details**

| The performance data included in this view allows you to determine if a bottleneck exists at a particular hop. View the data to determine if there is a hop which incurs more time than expected. Hop details are available for transaction classes.

| **Application topology**

View this high-level topology to understand the relationship between the applications in the EWLM domain that are processing work requests. In addition, you can view detailed statistics for each application as well as the servers on which it runs.

| **Server topology**

View this high-level topology to understand the relationship between the managed servers in the EWLM domain that are processing work requests. In addition, you can view detailed statistics for each server as well as the applications and application instances on which it runs.

| **Performance index monitor**

Use this monitor to view the performance index (PI) of a service class over a period of time. The monitor is updated automatically every 10 seconds. If the PI is less than or equal to 1, the goal is being met. If the PI is greater than 1, the goal is not being met. If the PI is consistently greater than 1, consider viewing additional monitors and reports to determine if there is a performance problem.

### Goal achievement monitor

Use this monitor to compare the actual performance of the work requests to the service class goal. This monitor is only available if the service class specifies an average response time, percentile response time, or velocity goal.

### Transaction rate monitor


Use this monitor to view how many transactions per second are processed during a specific time interval. You can compare the transaction rate monitor to the PI monitor to determine if there is a direct relationship between specific time intervals. For example, you can determine if the performance index increases when the number of transactions increases. Also, if a work request begins during one time interval and continues processing over multiple time intervals, the work request is included in each subsequent interval.

### Processor utilization monitor

Use this monitor to determine if there are specific time intervals when the this managed server's processors are used more than other managed servers. This monitor shows the combined utilization percentage of the managed server and all of its processors. You can compare the time intervals with a high utilization percentage to other managed server monitors to determine if work requests should be reallocated to another managed server.

To open one of these views from the monitor section of the EWLM Control Center, select an item in the table and select the specific report, monitor, or topology in the **Select Action** field. Then, click **Go**.

When determining where a problem resides in your EWLM domain, you need to read and interpret your performance data in a way that is specific to your business environment. EWLM provides many reports, monitors, and topologies that provide performance statistics. But, ultimately, you need to determine the appropriate combination of these reports to examine the performance data that is appropriate for your environment.

For detailed help that is specific to each field in the EWLM Control Center, click  when using the EWLM Control Center.

The following topics provide example data. Use these examples to learn ways to interpret, recover, or avoid specific problems.

#### Related concepts

“Troubleshooting EWLM” on page 326

View this topic to obtain a list of possible errors and recovery methods. The errors are organized by EWLM component such as domain manager, managed server, or application errors.

### Example: Performance index is 0

In this example, you discover that the performance index (PI) of a service class is 0. The PI indicates if a service class is meeting its goal. If the index is equal to or less than 1, the goal is being met. If the PI is greater than 1, the goal is not being met. However, a PI of 0 indicates that no transactions are being processed. There is no data to report; therefore, the PI is 0.

#### Performance data

The following figure provides example performance data that indicates that the Percentile service class has a PI of 0.

Select	Service class <sup>2</sup>	PI <sup>^</sup>	Importance <sup>1</sup>	Performance <sup>^</sup>	Goal <sup>^</sup>
<input type="radio"/>	Response time service class	2.00	Highest	02.000 average	01.000 average
<input type="radio"/>	Percentile service class	<None>	Medium	0% in 45.000	60% in 45.000
<input type="radio"/>	EWLM Service Class	<None>	<None>	<None>	Discretionary

Page 1 of 1      Total: 3   Filtered: 3   Displayed: 3

Figure 3. Service classes.

To verify that the PI is 0 due to no transactions being processed, view the service class details. The following figure contains the service class's transaction data.

## Statistics

Performance: None  
Performance index: None  
Transactions:

Hop	Completed	Failed	Stopped	Ended in unknown state	Not valid	In progress	Total
0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0

Page 1 of 1      Total: 3   Filtered: 3   Displayed: 3

Figure 4. Service class details report.

### Recovery steps

In this example, you need to determine if the work requests are classified properly. If they are classified properly, no action is necessary. To verify the classification of work requests, edit the transaction class rules in the domain policy. Ensure the following:

- The transaction class is created for the appropriate application.
- The transaction class rules are classifying the correct work requests. Ensure that each "Filter" on page 35 is specified correctly.
- The transaction class specifies the appropriate service class.
- The application is ARM instrumented and included in the domain policy.

### Example: Performance index is > 1

If the PI is greater than 1, the service class goal is not being met. There can be numerous problems that can contribute to a goal not being met. For example, the processor utilization is too high, the number of transactions being processed is too high, the work requests are not classified to the correct service class, or the active service policy is not correct. You need to use a combination of detail reports, monitors, and topologies to determine why a service class goal is not being met. The specific monitor views that you use depend on the type of performance data that you obtain as you examine the problem.

### Performance data

The following figure provides example data that indicates that the Response time service class has a PI of 2.00.



Select	Service class <sup>2</sup>	PI <sup>^</sup>	Importance <sup>1</sup>	Performance <sup>^</sup>	Goal <sup>^</sup>
<input type="radio"/>	Response time service class	2.00	Highest	02.000 average	01.000 average
<input type="radio"/>	Percentile service class	0.00	Medium	0% in 45.000	60% in 45.000
<input type="radio"/>	EWLM Service Class	<None>	<None>	<None>	Discretionary

Page 1 of 1      Total: 3   Filtered: 3   Displayed: 3

Figure 5. Service classes.

This data shows that one service class has a PI that is greater than 1. To determine why this service class goal is not being met, begin by viewing the details of the service class. In the details report, examine the transaction statistics and identify which transaction classes are associated with this service class.

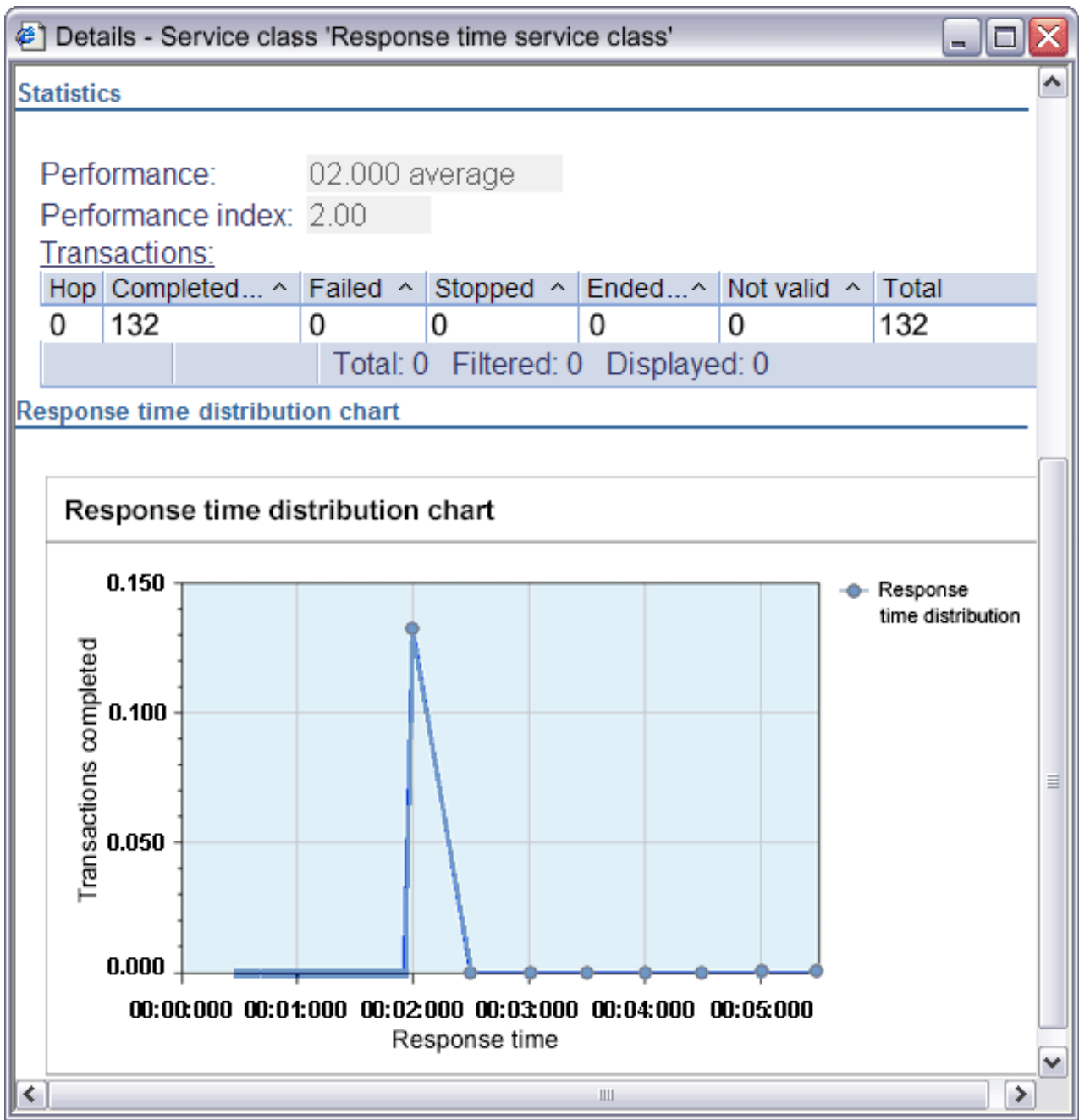


Figure 6. Service class details report that specifies a PI that is greater than 1.

Based on this data, you determine that all of the transactions have completed successfully. In addition, you determine that only one transaction class is associated with this service class. This data does not indicate why the service class goal is not being met. The Response time distribution chart in the transaction class details report or service class details report helps you determine if any transactions have differing response times, and so, how many have a different response time and what is the response time? After viewing the transaction class details and service class details, you determine that all work requests take two seconds to complete. This data, however, still does not indicate the cause of the problem.

To investigate even further, view the server topology for the service class. The PI might be high due to problems with the managed server's processors that are processing the work requests. The server topology displays a high-level view of the servers in the EWLM domain. The server

topology includes the managed servers and the ARM-instrumented applications that are processing work requests that are classified to the Response time service class.

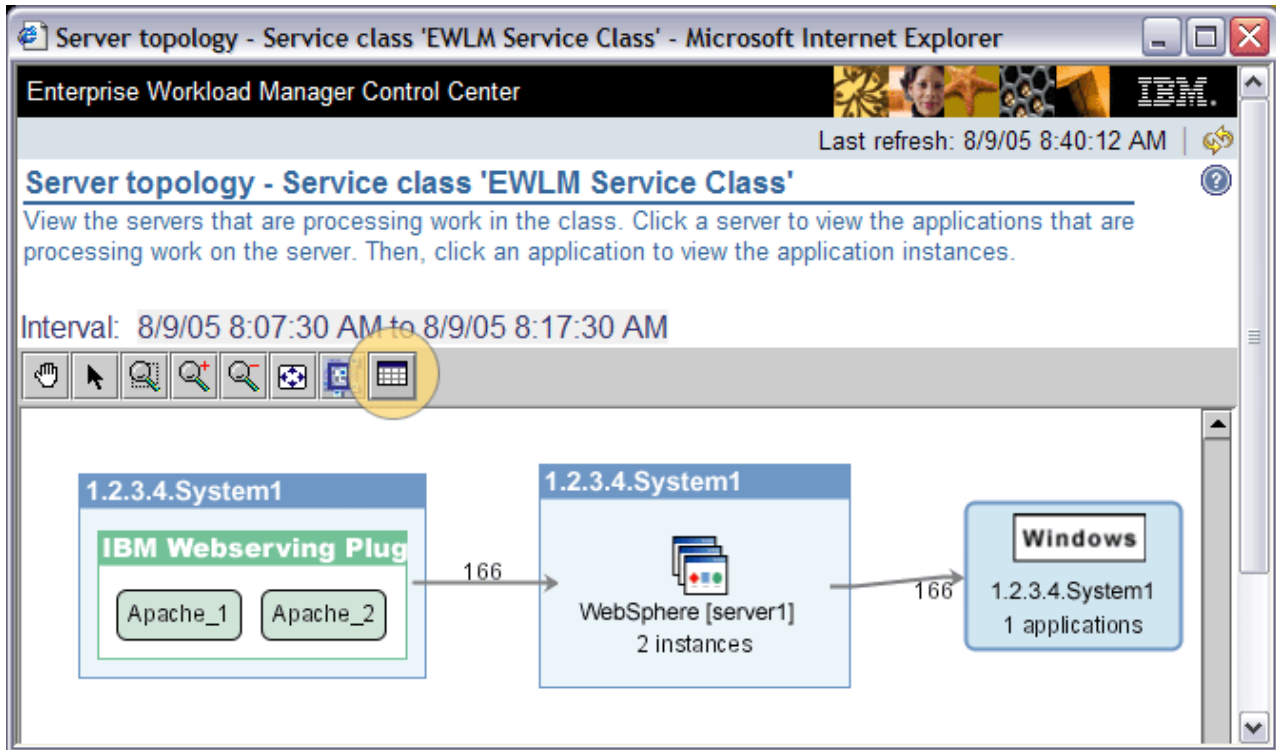


Figure 7. Server topology for the Response time service class.

To obtain more specific data when you view a topology in the EWLM Control Center, click the table icon. The icon is highlighted in Figure 11. Figure 12 shows you the detailed performance statistics for this server topology.

Hop	Name	Type	Average response	Average active time	Processor using (%)
0	localhost.server1	Server	00.000	00.000	79%
0	Apache [ApacheGroup00]	Application	02.000	00.100	79%
0	Apache_1	Instance	02.000	00.100	79%
0	Apache_2	Instance	02.000	00.100	79%
0	localhost.server2	Server	00.000	00.000	56%
0	Apache [ApacheGroup00]	Application	02.000	00.100	57%
0	Apache_1	Instance	02.000	00.100	55%
0	Apache_2	Instance	02.000	00.100	55%
1	localhost.server1	Server	00.000	00.000	79%
1	WebSphere [WebGroup]	Application	01.900	00.400	79%
1	WebSphere_1	Instance	01.900	00.400	81%
1	WebSphere_2	Instance	01.900	00.400	81%
1	localhost.server2	Server	00.000	00.000	56%
1	WebSphere [WebGroup]	Application	01.900	00.400	55%
1	WebSphere_1	Instance	01.900	00.400	60%
1	WebSphere_2	Instance	01.900	00.400	60%
2	localhost.server1	Server	00.000	00.000	79%
2	DB2 [DBGroup]	Application	01.500	01.500	79%
2	DB2_1	Instance	01.500	01.500	79%
2	DB2_2	Instance	01.500	01.500	79%
2	localhost.server2	Server	00.000	00.000	56%
2	DB2 [DBGroup]	Application	01.500	01.500	57%
2	DB2_1	Instance	01.500	01.500	55%
2	DB2_2	Instance	01.500	01.500	55%

Figure 8. Server topology data.

In this example topology data, you are particularly interested in the **Average active time** because work requests associated with this service class are completing, on average, in two seconds rather than one second. The **Average active time** specifies the amount of time, on average, that each server or application uses to process work requests. Use this data to determine if one of the server or applications is processing the work requests slower than expected. As shown in the server topology data, all database application instances in hop 2 have an average active time of 1.5 seconds. This is extremely high when compared to the goal of 1 second.

At this point, you should examine why the DB2 application appears to take longer to process than expected. View the **Processor using %** in the server topology to view the percentage of time the application was able to use the processor when requested. A high number indicates that the processor is available when needed. If the using percent is low, the service class is having trouble processing its work requests. In this example, localhost.server2 has a lower processor using percentage than localhost.server1.

To investigate the server (localhost.server2) with a lower processor using percentage further, view the managed server details to determine the **Average processor utilization**. This value specifies the total percentage of the all processors on the managed server that are being used. The following is example data from the managed server details for localhost.server2.

#### Statistics

Average processor utilization %	89
Real memory	1,024
Number of logical processors	4
Page fault rate (faults per second)	8.0

#### Recovery steps

Based on data that you obtained when investigating the problem, you determine that the main cause of the service class missing its goal is due to the processor utilization being too high. The server is potentially attempting to process too many work requests; therefore, the managed server cannot process the work requests in an efficient amount of time.

The following is a list of ways to recover:

- Move work requests to a different managed server for processing to reduce the processor utilization percent.
- Adjust the service class goal to a higher response time that takes into consideration a managed server with a high processor utilization percentage.
- Add more processor capacity to the managed server to reduce the total processor utilization of the managed server.
- Move high priority work requests to a different managed server. View the managed server details to determine what other service classes contain work requests that are being processed by the same server. Perhaps, you can move some work requests to a different service class. The following is an example of the managed server's details report that identifies the service classes processed by the managed server.

Service class	Processor usage %	Processor delay	Paging delay	I/O delay
EWLM Service Class	79.6	15.9	0.0	0.0
Percentile service class	79.4	15.9	0.0	0.0
Response time service class	79.6	15.9	0.0	0.0
Page 1 of 1		Total: 3 Filtered: 3 Displayed: 3		

Figure 9. Managed server details report.

View the **Processor utilization** field for each service class to determine how often work requests are able to process without being delayed. If the percent is low, the service class is not able to obtain processor resources when needed to complete its work requests. You should consider moving the service class to a different managed server for processing.

View the delay fields to determine if a large percent of a service class' work requests are being delayed. You should consider moving the service class to a different managed server for processing.

### Example: Service classes not meeting their goal

A quick method to determine if a performance problem exists in your EWLM domain is to view the exceptions report. The exceptions reports lists all services classes whose goals are not being met. If a service class is included in the exception report, its performance index is greater than 1. If a service class appears in the exceptions report, use the following list as an aide to examine the problem:

- Is the goal specified for the service class appropriate? Perhaps, you expected a performance goal that is not appropriate for the number of transactions being processed by the applications or servers in the EWLM domain.
- What work requests are classified to the service class? View the transaction or process classes in the domain policy that use the service class. Ensure that the rules are specified correctly so the work requests are filtered properly.
- What servers are processing work requests for this service class? View the server topology to view performance statistics for each server processing work requests for this service class. Perhaps a managed server is processing too many or too few transactions. View the processor utilization for each managed server and ensure that the workload is balanced.
- What applications are processing work requests for this service class? View the application topology to view performance statistics for each application processing work requests for this service class.
- View the transaction rate and transaction count monitors for the service class to determine if there are specific time intervals when more transactions are being processed.
- View the performance index monitor to determine if the PI fluctuates over a specific interval of time. Compare this time interval to the transaction rate and transaction count monitors.

### Example: Allocation of managed server processor utilization

To ensure that there is an adequate amount of processing capacity in your EWLM domain, periodically view the managed servers' high-level monitor view. If there is not an adequate amount available, service classes might miss their goals simply because the servers cannot process the work requests efficiently. Based on the processor utilization, you decide to move high priority work requests to different servers or obtain additional processors to handle the workload required in your environment.

The following screen capture provides example data for EWLM managed servers and specifies their processor utilization percentages:

Select	Managed server	Processor utilization %	State	Active service policy
<input type="radio"/>	localhost.server1	10.0	Active	Weekday policy
<input type="radio"/>	localhost.server2	10.0	Active	Weekday policy

Page 1 of 1      Total: 2   Filtered: 2   Displayed: 2

Figure 10. Managed servers.

If you determine that a managed server's processor utilization is high, you can investigate what service classes are contributing to that managed server.

### Example: Classification of work requests to service classes

Classification of a work request to a service class depends on where the work request originated. If the work request was initiated by an application or user, it is classified by an entry application. The entry application compares it to the transaction classes and assigns it to the appropriate service class. If the work request originated by a server, it is compared to the process class for that operating system. Then, it is assigned to the service class for that process class. Process classes are optional.

For transaction classes, the entry application is the first ARM—instrumented application that receives a work request. The entry application compares the work request to the rules in a transaction class and classifies it to the appropriate service class. In addition, after a work request is classified by an entry application, the work request contains a correlator. The correlator allows all applications and managed

servers in the EWLM domain to know that the work request has been classified. Therefore, additional applications and servers that receive the work request will not classify the work request again.

The classification step in EWLM is important for reporting purposes. The entry application is the first place that a work request is associated with a service class. If the work request is not classified properly, the reporting data will not be accurate.

To view which applications have classified work requests, from the EWLM Control Center, click **Monitor** → **Transaction classes**. View the **Entry application** column for a list of applications that are classifying work requests in your EWLM domain. Verify that all of the applications listed are accurate.

The following is an example list of transaction classes and their associated entry applications:

Select	Transaction class ▲	Response time	Entry application ^
<input type="radio"/>	default-IBM DB2 Universal Database	02.000 average	IBM DB2 Universal Database
<input type="radio"/>	default-IBM Webseving Plugin	02.000 average	IBM Webseving Plugin
<input type="radio"/>	default-WebSphere	02.000 average	WebSphere Application Server
<input type="radio"/>	EWLM Default Application Transaction Class	02.000 average	IBM Webseving Plugin
<input type="radio"/>	Web transactions	02.000 average	IBM Webseving Plugin

Page 1 of 1      Total: 5   Filtered: 5   Displayed: 5

Figure 11. Transaction classes.

## Configuring serviceability options

Administrators can configure the system logging properties for EWLM through the serviceability options in the EWLM Control Center.

- | Serviceability options are primarily intended for use by and as indicated by IBM Service. You can use the serviceability options in the EWLM Control Center to define trace levels for EWLM activity. In addition, you can adjust the trace level.
- | The only trace log intended for you to use is the system log file, System.log. This log file contains information that might help with basic problem determination.
- | **Note:** Higher trace levels might significantly affect the performance of the systems in your EWLM domain.
- | Refer to the EWLM Control Center online help for instructions on how to configure serviceability options.
- | To configure serviceability options, complete the following steps:
  1. Click **Domain Settings** in the EWLM Control Center home page.
  2. Click **Serviceability** to configure trace options for managed servers or partition workload groups, and then click **Advanced**.
  3. Complete the fields as necessary to specify serviceability options. For more information on the types of trace logs, see the EWLM Control Center online help.

### Related concepts

“Troubleshooting EWLM” on page 326

View this topic to obtain a list of possible errors and recovery methods. The errors are organized by EWLM component such as domain manager, managed server, or application errors.

## Working with service logs

System administrators can view the system log files that EWLM creates to help in problem determination.

The serviceability options within the EWLM Control Center allow you to define the trace level for EWLM activity. One of these trace files is the system log file. The system log file, System.log, is the only log file that is intended for you to use for problem determination. This log file contains information that may help with basic problem determination.

The system log file uses a fixed name of System.log. EWLM writes data to the file until it reaches the maximum size of 1 megabyte (MB). The next output after that causes the file to close, and the file is renamed to Previous.log. If a Previous.log file already exists, EWLM deletes it from the file system. Then, EWLM creates a new System.log file that will contain the next megabyte worth of trace information.

EWLM creates separate system log files for the domain manager, managed server, and firewall broker, if one exists.

To work with a system log, complete the following

1. Log on to the system that contains the EWLM component whose log file you want to view. For example, if you want to view the system log file for the EWLM domain manager, log on to the system defined as the domain manager. If you want to view the log file for the EWLM managed server, log on to system defined as the managed server.
2. Navigate to the following file system path:*EWLM User Data Directory/servers/configID* where the variables are defined as following:

Table 85. List of variables and their descriptions.

Variable	Description
<i>EWLM User Data Directory</i>	The file path specified when you installed the domain manager, managed server, or firewall broker, whichever applies.
<i>configID</i>	The configuration ID specified when you configured the domain manager, managed server, or firewall broker, whichever applies.

3. Examine the system.log file.
4. Examine the previous.log file, if one exists.

### Related concepts

“Troubleshooting EWLM” on page 326

View this topic to obtain a list of possible errors and recovery methods. The errors are organized by EWLM component such as domain manager, managed server, or application errors.

---

## Uninstalling EWLM

View this topic to learn how to uninstall EWLM.

When you installed the Virtualization Engine EWLM domain manager to your system, an uninstaller was also copied to the directory where the Virtualization Engine was installed and launched. You can use the Virtualization Engine uninstall wizard to remove the domain manager, the EWLM Control Center, the managed server installation images, and all of the required runtime components.



The wizard does not uninstall the managed servers or firewall brokers from the servers on which they are installed. To uninstall the managed servers or firewall brokers, you will need to run the uninstaller that copied to the endpoint system on which you installed the managed server or firewall broker.

If you want to uninstall a fix package, see Fixes and Updates in the Virtualization Engine Management collection topic. Ensure that you adhere to the uninstall considerations that are described in Fix package uninstallation considerations for EWLM.

To uninstall EWLM, complete the following:

- Uninstall domain manager and EWLM Control Center
- Uninstall managed servers and firewall brokers

## Domain manager and EWLM Control Center

To uninstall the domain manager and the EWLM Control Center, follow these steps:

1. Verify that the domain manager is stopped. For more information, see “Stopping EWLM” on page 194. If you completed the Virtualization Engine EWLM prerequisite uninstall tasks, you can skip to step 4.
2. Verify that the EWLM Control Center is stopped. For more information, see “Stopping the EWLM Control Center” on page 196.
3. Delete the domain manager instance and its configuration data. For more information, see “Deleting domain manager configuration data” on page 323
4. Uninstall the domain manager and EWLM Control Center:

### AIX, i5/OS, Linux, and Windows

Follow the step-by-step instructions for uninstalling the Virtualization Engine found in the Virtualization Engine topic collection. When you get to the panel that allows you to select which system services to uninstall, select the EWLM components you want to uninstall.

- z/OS Delete the Virtualization Engine WebSphere Application Server profile. For more information, see “Deleting a Virtualization Engine WebSphere Application Server profile on z/OS” on page 324.

## Managed servers and firewall brokers

To uninstall the managed servers or firewall brokers follow these steps:

1. Verify that the managed server and firewall broker are stopped, whichever apply. For more information, see “Stopping EWLM” on page 194.
2. Delete the managed server data, if applicable. For more information, see “Deleting managed server configuration data” on page 324
3. Delete the firewall broker data, if applicable. For more information, see “Deleting firewall broker configuration data” on page 324
4. Uninstall the managed server or firewall broker using either *interactive* mode or *silent* mode.

- **Interactive mode: AIX, HP-UX, Linux, Solaris, Windows**

To call the Uninstall wizard to uninstall the managed server or firewall broker in **interactive** mode, you will need to do the following on each platform:

- a. Navigate to the **\_uninst** subdirectory of the directory where the managed server or firewall broker was installed and launch the following program:
  - On AIX, HP-UX, Linux, and Solaris: `uninstall.bin`.
  - On Windows: `uninstall.exe`.
- b. Complete the Uninstall wizard to remove the managed server or firewall broker from the system.

- c. Repeat these steps on each server on which you want to uninstall a managed server or firewall broker

- **Silent mode: AIX, HP-UX, Linux, Solaris, Windows**

To uninstall in **silent** mode, navigate to the **\_uninst** subdirectory of the directory where the managed server or firewall broker was installed and run this command from a command line or batch file:

```
uninstaller -silent
-P managedServerBean.activeForUninstall=[true | false]
-P firewallBrokerBean.activeForUninstall=[true | false]
```

where each of the possible parameters is described as follows:

- *uninstaller* is the file name of the uninstaller for the appropriate platform; *uninstall.bin* for AIX, HP-UX, Linux, Solaris, and *uninstall.exe* for Windows.
- *managedServerBean.activeForUninstall* indicates whether to uninstall the managed server or not. Possible values are **true**, to uninstall the managed server, or **false**, to not uninstall the managed server. The default value is true.
- *firewallBrokerBean.activeForUninstall* indicates whether to uninstall the firewall broker or not. Possible values are **true**, to uninstall the firewall broker, or **false**, to not uninstall the firewall broker. The default value is true.

You must do this on each system from which you want to remove the managed server or firewall broker.

**i5/OS** To uninstall the managed server or firewall broker from a system running i5/OS, see “Uninstalling EWLM on i5/OS.”

**z/OS** On z/OS, EWLM is installed using SMP/E. No further steps are recommended for uninstalling EWLM.

5. Repeat these steps on each system on which you want to uninstall a managed server or firewall broker.

#### **Related concepts**

“Stopping EWLM” on page 194

View this topic to learn how to stop the EWLM domain manager, managed server, and firewall broker. This task describes how to stop the EWLM function that runs on the system that contains the EWLM domain manager, managed server, or firewall broker. This task does not shutdown or stop the system.

## **Uninstalling EWLM on i5/OS**

View this topic to learn how to uninstall the i5/OS EWLM managed server from a remote system or locally.

To uninstall the EWLM managed server on i5/OS, use one of the following methods:

### **Uninstalling the EWLM managed server from a remote system**

View this topic to learn how to uninstall the i5/OS EWLM managed server from a remote Windows system.

Ensure that the Windows system can connect to the i5/OS system. For example, you can create a network drive that maps to the root directory of the Integrated File System (IFS) on the i5/OS target system. In addition, the JRE must exist on the Windows system.

To uninstall the EWLM managed servers, use one of the following methods:

Run one of the following commands from a Windows command prompt:

Option	Description
<b>Interactive mode</b>	<code>java -Dos400.share.drive -cp "M:\QIBM\ProdData\VE2\EWLMS\_uninst\uninstall.jar" run -os400</code>
<b>Console mode</b>	<code>java -Dos400.share.drive -cp "M:\QIBM\ProdData\VE2\EWLMS\_uninst\uninstall.jar" run -console -os400</code>
<b>Silent mode</b>	<code>java -Dos400.share.drive -cp "M:\QIBM\ProdData\VE2\EWLMS\_uninst\uninstall.jar" run -silent -os400</code>

where *M*: represents the network drive that points to the i5/OS target system.

## Uninstalling EWLM managed server on local system

View this topic to learn how to locally uninstall the i5/OS EWLM managed server.

Run one of the following commands from an i5/OS SHELL.

Option	Description
<b>Console mode</b>	<code>java -cp /QIBM/ProdData/VE2/EWLMS/_uninst/uninstall.jar run -console</code>
<b>Silent mode</b>	You must run the silent command from the path where the uninstall response file resides. <code>java -Dos400.awt.native=true -cp /QIBM/ProdData/VE2/EWLMS/_uninst/uninstall.jar run -silent -options ./EWLMSi50SUninst.rsp</code>

## Deleting domain manager configuration data

If uninstalling EWLM, you should delete the domain manager configuration data before you continue the EWLM uninstall. The Virtualization Engine uninstall wizard does not delete the domain manager instance or its associated configuration data.

**Note:** If you want to access your domain policy information, you must export your domain policies to a directory where they can be accessed after the uninstall is complete.

The steps for completing this task on the supported platforms follow:

Use the `deleteDM` command to delete the domain manager configuration data. This command will delete the contents of the EWLM directory that contains the configuration data for the WebSphere Application Server instance and the EWLM Control Center.

### Run deleteDM

```
deleteDM configID -adminUser userid -adminPW userpwd
```

where each parameter is defined as follows:

- *configID* is the EWLM configuration ID that was created when you ran the `createDM` command.
- *userid* is the WebSphere Application Server administrator user ID.
- *userpwd* is the password associated with *userid*.

For more information about issuing the command on your operating system, if supported, see, "Command usage considerations" on page 91.

After you run the `deleteDM` command, continue uninstalling EWLM.

## Deleting a Virtualization Engine WebSphere Application Server profile on z/OS

View this topic to learn how to delete a Virtualization Engine WebSphere Application Server profile on z/OS.

### Run deleteWAS

To delete the Virtualization Engine WebSphere Application Server profile on z/OS, run the following command, replacing the example values shown in italics with your own.

```
deleteWAS -cfg filename.cfg -sec path
```

For more information about issuing the command on your operating system, if supported, see, “Command usage considerations” on page 91.

The following table describes each of the parameters and provides a place for you to enter your specific values when you print this page for reference.

Parameter	Description	Value
-cfg	The name of the configuration file created by configWAS.	
-sec	The optional path of the shell command, <code>commands.sh</code> , that contains commands to define security. The default path is <code>/misc/RACFMSTR</code> under the directory specified through the command as the HFS root.	

## Deleting managed server configuration data

The uninstall wizard does not delete the managed server instance or its associated configuration data. Although not required, you should delete this data from your system before you uninstall the managed server. The steps for completing this task on the supported platforms follow:

Use the `deleteMS` command to delete the managed server configuration data. This command will delete the contents of the managed server directory where configuration data resides.

### Run deleteMS

```
deleteMS configID
```

where *configID* is the managed server configuration ID that was created when you ran the `createMS` command.

For more information on issuing the command on your operating system, if supported, see, “Command usage considerations” on page 91.

After you run the `deleteMS` command, continue uninstalling the managed server.

You can now continue uninstalling the managed server.

## Deleting firewall broker configuration data

The uninstall wizard does not delete the firewall broker instance or its associated configuration data. Although not required, you should delete this data from your system before you uninstall the firewall broker. The steps for completing this task on the supported platforms follow:

Use the `deleteFB` command to delete the firewall broker configuration data. This command will delete the contents of the firewall broker directory where configuration data resides.

## Run deleteFB

```
deleteFB configID
```

where *configID* is the firewall broker configuration ID that was created when you ran the createFB command.

For more information about issuing the command on your operating system, if supported, see, “Command usage considerations” on page 91.

After you run the deleteFB command, continue uninstalling the firewall broker.

You can now continue uninstalling the firewall broker.

## Fix package uninstallation considerations for EWLM

When uninstalling a fix package, whether it be a Virtualization Engine fix package for a systems service or a fix package for a product that a systems service is dependent on, there are some items to consider. These special considerations are specific to the Enterprise Workload Manager (EWLM) systems service.

### EWLM

When uninstalling a fix package for EWLM, consider the following:

- **Prerequisites**

1. If you want to access the domain policy information that you changed or created while using the fix package, you must save the domain policy. To save a domain policy, use the export function in the EWLM Control Center. Export the domain policy to a directory where it can be accessed after the fix package is uninstalled. For more information on how to export a domain policy, see “Managing a domain policy” on page 267.
2. If you want to keep any configuration changes made while using the fix package, such as adding users or setting up SSL, make note of the changes before uninstalling the fix package so that you can redo the changes later. Use the displayDM command to display the configuration values. For information about how to use this command, see “Displaying EWLM configuration properties” on page 265.
3. AIX, HP-UX, Solaris: Log on as a user with root authority and save the auth and limits files. Specify the following on the managed server:

```
| cp /etc/ewlm/auth /etc/EWLMSAVE/auth
| cp /etc/ewlm/limits /etc/EWLMSAVE/limits
```
4. If you are uninstalling a domain manager fix package, ensure that EWLM on the domain manager is stopped. For more information, see Stopping EWLM.
5. If you are uninstalling a domain manager fix package, ensure that the Control Center is stopped. For more information, see Stopping the EWLM Control Center.
6. If you are uninstalling a managed server fix package, ensure that EWLM on the managed servers and firewall brokers are stopped. For more information, see Stopping EWLM.
7. If you are uninstalling a managed server fix package, ensure that you stop the ARM-enabled applications that run on the system.

- **Uninstallation notes**

1. To uninstall an EWLM managed server fix package, complete the instructions as described in the Uninstall managed servers and firewall brokers topic. Do not complete step 2 that indicates to delete the managed server configuration data. When the Uninstall wizard opens, ensure that you select to *Uninstall last installed EWLM fix package* rather than *Uninstall entire EWLM product and last installed EWLM fix package*. In addition, see Fixes and Updates in the Virtualization Engine Management collection topic for uninstallation information.

2. To uninstall an EWLM domain manager fix package, complete the instructions as described in Fixes and Updates in the Virtualization Engine Management collection topic.

- **Post-uninstall notes**

1. AIX, HP-UX, Solaris: Restore the auth and limits files. Specify the following on the managed server:
 

```
cp /etc/EWLMSAVE/auth /etc/ewlm/auth
cp /etc/EWLMSAVE/limits /etc/ewlm/limits
```
2. If you configured a domain manager after installing a fix package, Delete the domain manager instance and its configuration data.
3. AIX, HP-UX, Solaris, and Windows: Restart the system. On Windows, the uninstall prompts you to restart after the uninstall completes.

---

## Troubleshooting EWLM

View this topic to obtain a list of possible errors and recovery methods. The errors are organized by EWLM component such as domain manager, managed server, or application errors.

Use the Troubleshooting topic to help you understand, isolate, and resolve problems that your EWLM experiences. Sometimes you will be able to resolve a problem on your own, other times you will need to gather information to help the service technicians resolve your problem in a timely manner. To effectively troubleshoot EWLM, see the following topics for specific troubleshooting information:

If you are unable to troubleshoot a problem with EWLM, contact your IBM service representative.

### Troubleshooting: Applications

Learn how to troubleshoot applications that you want EWLM to monitor.

The following is a list of possible errors and ways to recover:

Error	Resolution
An application is identified in the deployed domain policy, but EWLM is not monitoring the application.	<p>Ensure that the application uses the Application Response Measurement (ARM) standard APIs and that ARM is enabled.</p> <p>Use the following to verify which applications enable ARM.</p> <ul style="list-style-type: none"> <li>• <b>AIX, HP-UX, Linux, Solaris:</b> Use the lsarm command. For more details about this command, see “lsarm” on page 327.</li> <li>• <b>i5/OS:</b> Use the WRKARMJOB CL command.</li> <li>• <b>Windows</b> Use the ewlmWinADTool command. For more details about this command, see “ewlmWinADTool.exe” on page 328.</li> <li>• <b>z/OS:</b> Use the DISPLAY WLM command. For more information, see “Verifying the status of ARM services on z/OS” on page 189.</li> </ul>
HTTP server for iSeries (IBM Web Serving plug-in) loads and makes ARM calls but classification is not working	Ensure that your system meets the minimum operating system requirements and contains the latest fix packages. For more information, see Specified operating environments.

Error	Resolution
<p>EWLM will not monitor application-level or operating system-level work that runs on Linux. If you are attempting to monitor application-level work, the ARM library for the EWLM managed server fails with the following return code:</p> <p>-7022 ARM registration calls from applications that execute using LinuxThreads</p>	<p>The EWLM managed server for Linux on SUSE Linux Enterprise Server (SLES) 9 does not support applications, middleware, daemons or other deployed program solutions that run in a Linux Threads environment. Furthermore, the EWLM managed server for Linux process monitoring and performance metrics collection is unreliable for programs that run in this LinuxThreads environment and can lead to unexpected results in EWLM reporting and management. The EWLM managed server for Linux on SLES 9 does fully support applications, middleware, daemons or other deployed program solutions that execute in a Native POSIX Threading Library (NPTL) environment.</p>

## lsarm

The `lsarm` command displays information about ARM-instrumented applications registered with AIX, HP-UX, Linux, or Solaris. The `-a` option displays information about applications by application name. The `-p` option displays information about the applications used by a process.

### Syntax

```
lsarm -a [-g] [-t [-r]] [-u] [ApplicationName ...]
```

or

```
lsarm -p [-a [-g] [-t] [-r]] [processNumber ...]
```

### Flags

Parameter	Description
<code>-a</code>	Display application names
<code>-g</code>	Display group names associated with the application
<code>-t</code>	Display transaction class names associated with the application
<code>-r</code>	Display property names associated with the transaction class
<code>-u</code>	Display process numbers using the application
<code>-p</code>	Display the applications used by a process

### Examples

To display all application names, type:

```
lsarm -a
```

To display group names, transaction class names, property names, and process numbers using the database application, type:

```
lsarm -a -g -t -r -u database
```

To display the application names, group names, and transaction names used by process 25038, type:

```
lsarm -p -a -g -t 25038
```

## ewlmWinADTool.exe

The ewlmWinADTool tool allows you run a command that allows you to display ARM data. This command is supported only on Windows. You can access this command from the managed server default installation path: C:\ProgramFiles\IBM\VE2\EWLMMS\classes\ms.

### Syntax

```
ewlmWinADTool arm4psinfo [{pid|*} [list:armTagList] [filter:armTag=name]]
```

where each of the parameters are defined as follows:

Parameter	Description
<i>pid</i>	Denoted in decimals, the pid is the process identifier (PID) for a process as shown in the Task Manager. An asterisk (*) indicates to query for all processes that are currently doing ARM 4.0 instrumentation.
<i>armTag</i>	Can be AR (application), TR (transaction), AI (application instance) or AG (application group). To query a particular set of ARM 4.0 data, use the subcommand <i>filter:armTag=name</i>
<i>name</i>	The string to filter ARM 4.0 data through, for a particular category of data. Use quotation marks (") for names that have spaces. If you do not specify a name, no filtering occurs.
<i>armTagList</i>	<p><i>list:armTagList</i> defines which category of ARM 4.0 data to display to the console. Specify the <i>armTagList</i> in the form of "<i>armTag+armTag+armTag</i>".</p> <p>If <i>armTagList</i> is empty with the <i>list:</i> subcommand present, only the summary information of the target process(es) will be displayed.</p> <p>To display all the ARM 4.0 data, do not specify the <i>list:</i> subcommand.</p> <p><b>Note:</b> The group name and the instance name of an application instance are always displayed together, when AI, AG or AI+AG is specified.</p> <p>Not all combinations of display lists with filters are applicable. Filters are ignored when not applicable. For example, with a display list of AR (application) and a filter of TR (transaction), data of all the registered applications will be displayed.</p>

### Examples

To display all ARM 4.0 data for all the processes that are doing ARM instrumentation. This may take a while on a system that is busy doing heavy ARM processing.

```
ewlmWinADTool arm4psinfo
```

To display only the summary ARM 4.0 information of all the processes doing ARM instrumentation.

```
ewlmWinADTool arm4psinfo * list:
```

To display all the ARM 4.0 data that is associated with an application named, "*sample*" of all the processes that are doing ARM instrumentation.

```
ewlmWinADTool arm4psinfo * filter:AR=sample
```



To display the transaction registration data associated with the application named, "sample", with 1508 being its process identifier (PID).

```
ewlWinADTool arm4psinfo 1508 list:TR filter:AR=sample
```

For more details use:

```
ewlWinADTool -h arm4psinfo
```

## Verifying the status of ARM services on z/OS

View this topic to learn how to verify the status of ARM services on z/OS.

To verify the status of ARM services on z/OS, run the following command from an operator console:

```
DISPLAY WLM,AM
```

The following is an example of a response to the command, showing that ARM services are enabled.

```
IWM075I 17.54.21 WLM DISPLAY 512
EWLM ARM SERVICES ARE ENABLED
EWLM MANAGED SERVER JOBNAME=EWLMS ASID=0032
EWLM POLICY NAME=BOSCH DAYTIME POLICY
NUMBER OF REGISTERED PROCESSES=3, APPLICATIONS=1
```

## Troubleshooting: EWLM Control Center

Learn how to troubleshoot errors in the EWLM Control Center.

If you are experiencing problems with the EWLM Control Center, the problem might be a result of an incorrect Web browser requirement or WebSphere Application Server setting. For the EWLM Control Center to function properly, both the Web browser and WebSphere Application Server need to meet specific requirements.

The following is a list of possible errors and ways to recover:

*Table 86. EWLM Control Center errors and resolutions.*

Error	Resolution
<i>Logging in and out</i>	

Table 86. EWLM Control Center errors and resolutions. (continued)


Error	Resolution
<p>Inactive user automatically timed out.</p>	<p>If a user is logged into the Control Center and is inactive for a specified amount of time, the user is automatically logged out. If you want to disable or adjust the time interval before a user is automatically logged out, complete the following:</p> <ol style="list-style-type: none"> <li>1. Log on to the WebSphere Application Server Administration console.</li> <li>2. Change the session timeout interval for the server:               <ol style="list-style-type: none"> <li>a. Select <b>Servers</b> → <b>Application Servers</b> → <i>Server1</i> → <b>Web Container</b> → <b>Session Management</b>. <b>Note:</b> For i5/OS, replace <i>server1</i> with <i>ewlmServer</i>.</li> <li>b. In the <b>Session timeout</b> field, select <b>Set timeout to specify a time interval</b> or select <b>No timeout to disable the timeout interval</b>.</li> </ol> </li> <li>3. Change the session time out interval for the application:               <ol style="list-style-type: none"> <li>a. Select <b>Applications</b> → <b>Enterprise Applications</b> → <i>wlmwebui</i> → <b>Session Management</b>.</li> <li>b. In the <b>Session timeout</b> field, select <b>Set timeout to specify a time interval</b> or select <b>No timeout to disable the timeout interval</b>.</li> </ol> </li> <li>4. For this change to take effect, restart the EWLM Control Center by completing the following:               <ol style="list-style-type: none"> <li>a. Stop the EWLM Control Center. For more information, see “Stopping EWLM” on page 194.</li> <li>b. Start the EWLM Control Center. For more information, see “Starting the EWLM Control Center” on page 192.</li> </ol> </li> <li>5. Log in to the EWLM Control Center.</li> </ol> <p><b>Note:</b> This changes the timeout interval for all sessions running in this WebSphere Application Server.</p>
<p>When you log on to the EWLM Control Center, you encounter the following error: You cannot log in when global security is not enabled. Enable global security and log in.</p>	<p>Enable WebSphere Application Server global security.</p>
<p>After you log off the EWLM Control Center on a z/OS domain manager, the monitor views that are displayed in pop-up windows may reload the EWLM Control Center home page on which you are not logged off.</p>	<p>Ensure that you close all monitor views when you log off the EWLM Control Center. This ensures that no EWLM Web pages remain open after you log off.</p>
<p><b>Displaying of Web pages</b></p>	
<p>EWLM Control Center is not displaying the Web pages properly. For example:</p> <ul style="list-style-type: none"> <li>• Control Center pages do not open.</li> <li>• When  is clicked, the overview help opens rather than the help specific to the Web page.</li> </ul>	<p>Install and use Internet Explorer 6.0 with Service Pack 1 to access the EWLM Control Center.</p>
<p>Application or server topologies do not display.</p>	<p>Install Java Runtime Environment 1.4.2 with Service Release 2.</p>

Table 86. EWLM Control Center errors and resolutions. (continued)


Error	Resolution
Secondary windows do not open. For example, the monitors, detail reports, and topology views do not open.	Ensure that your Web browser allows pop-up windows. Also, ensure that any pop-up blocker programs are turned off.
Monitor views do not display properly.	<p>Install the Adobe SVG viewer on the system from which you access the EWLM Control Center. The first time you access a monitor view and EWLM recognizes the program is not installed, EWLM will ask you if you want to install it. If you select to not install it, the monitor views will not display.</p> <p>To install the Adobe SVG viewer, see Adobe  (http://www.adobe.com/svg/viewer/install/). Install the language and version of the Adobe SVG Viewer that corresponds to the language that the EWLM Control Center displays.</p>
Left pane of EWLM Control Center does not display the options that you want to work with. Possible options include any one or more of the following: Set up, Manage, and Monitor.	<p>Set up, Manage, and Monitor are the main views in the EWLM Control Center. To access each view, you must have the proper authority. EWLM displays only the views that you have authority to access.</p> <p>To assign a user IDs to EWLM roles, see “Assigning users to EWLM Control Center roles” on page 117.</p>
Security Alert	<p>To avoid the security alert warnings when using the EWLM Control Center, complete one of the following:</p> <ul style="list-style-type: none"> <li>• Click <b>OK</b> to ignore the warning and continue using the EWLM default certificate.</li> <li>• Use your own certificate rather than the EWLM default certificate. Ensure that the host name of the certificate matches the host name used in the URL to access the EWLM Control Center.</li> </ul>
<b>Storing data</b>	
WebSphere Application Server out of memory error.	<p>The default memory size for WebSphere Application Server is 256 megabytes. If the data in the Control Center’s reports exceeds this limit, an error occurs. To change the memory size of the WebSphere instance, complete the following:</p> <ol style="list-style-type: none"> <li>1. Log on to the WebSphere Application Server Administration console.</li> <li>2. Select <b>Servers</b> → <b>Application servers</b> → <i>Server1</i> → <b>Process Definition</b> → <b>Java Virtual Machine</b>. <b>Note:</b> For i5/OS, replace <i>server1</i> with <i>ewlmServer</i>.</li> <li>3. In the <b>Maximum Heap Size</b> field, specify a size in megabytes.</li> <li>4. Restart the Control Center WebSphere Application Server instance.</li> </ol>
<b>Downloading dependent applications</b>	
Web browser error indicating that EWLM cannot install JRE 1.4.2 with Service Pack 2 automatically.	EWLM attempted to download JRE 1.4.2 because it was not found on your system. JRE 1.4.2 with Service Release 2 is required to view the topologies in the EWLM Control Center. You must manually download JRE 1.4.2 with Service Release 2 onto your system.
<b>Accuracy of data</b>	

Table 86. EWLM Control Center errors and resolutions. (continued)



Error	Resolution
<p>On an x86 virtualized platform such as VMware 2.5.x any time sensitive application running within a virtual machine, including the guest operating system's performance counters, can generate anomalous time readings. A virtual machine cannot exactly duplicate the timing behavior of an x86 physical machine.</p>	<p>EWLM depends on the operating system timer accuracy and therefore is susceptible to anomalous time measurements. For more information about VMware virtual machine time keeping and how to improve timekeeping in virtual machines, see Timekeeping in VMware Virtual Machines  (<a href="http://www.vmware.com/pdf/vmware_timekeeping.pdf">http://www.vmware.com/pdf/vmware_timekeeping.pdf</a>). <b>Note:</b> You need Adobe Reader installed on your system to view or print this PDF. You can download a free copy from the Adobe Web site (<a href="http://www.adobe.com/products/acrobat/readstep.html">www.adobe.com/products/acrobat/readstep.html</a>) .</p>
<p>Transaction counts in server or application topology do not match transaction counts that EWLM displays in the service class or transaction class details.</p>	<p>This error only occurs immediately following a service policy activation.</p> <p>After a service policy activation, EWLM must collect data for a full interval duration to accurately display the transaction counts for each application or server included in the topology view.</p> <p>Note that the transaction counts in the service class or transaction class do not require a full time interval to complete before the counts are accurate. These transaction counts do not require the topology level data, which is only collected after each data collection interval completes.</p>
<p>When you recover an i5/OS system that includes Cryptographic Access Provider (5722-AC3), the Cryptographic Access Provider product fails when you use it.</p>	<p>EWLM uses the Cryptographic Access Provider (5722-AC3) product on i5/OS to provide secure communications. To recover from this error, follow the procedures in <i>iSeries Backup and Recovery</i> to ensure this product is properly set up.</p>
<p><b>Performance</b> or <b>Performance Index</b> report a value of <b>None</b>.</p>	<p>If the service class specifies a velocity goal and the work that uses that goal completes in less than 250 milliseconds, EWLM might not be aware of this work because it processes so quickly. For velocity performance goals, EWLM collects performance data four times each second, that is, every 250 milliseconds. Therefore, if work completes in less than 250 milliseconds, EWLM might not be aware that the work exists. For work that completes this quickly, you should use an Average response time or Percentile response time goal. This allows you to view the actual amount of time that the work took to process.</p>
<p><b>Transaction</b> field reports a negative number.</p>	<p>If the managed server collects a value transaction count that is not valid, EWLM is not able to display accurate transaction data. This problem can also indicate that other performance calculations are not accurate. Currently, there is no recovery solution.</p>
<p><b>Partition workload group</b> reports a value of <b>None</b>.</p>	<p>If a partition workload group moves from one partition workload group to another, the partition workload group might report a value of <b>None</b> for a short period of time. After EWLM detects the partition workload group change, the Partition workload group will be updated accordingly.</p>

Table 86. EWLM Control Center errors and resolutions. (continued)

Error	Resolution
Time value is incorrect	The EWLM Control Center's time values are based on the time zone that the Web browser uses. If the domain manager is using a time zone that differs from the time zone that the Web browser uses, the time values displayed in the EWLM Control Center will be updated to reflect the time zone of the domain manager rather than the Web browser's time zone.

#### Related information

iSeries Backup and Recovery PDF




## Troubleshooting: Managed server, domain manager, and firewall broker


Learn how to troubleshoot errors that you might encounter while working with the EWLM managed server, domain manager, or firewall broker.

The following is a list of possible errors and ways to recover.

Error	Resolution
<b>Installation</b>	
Insufficient memory to complete the managed server installation.	<p>During the managed server installation, ensure that the TEMP directory specifies a location that has at least 100 MB of memory as described in the worksheet for the managed server in "Managed server planning worksheet" on page 86.</p> <p>If you perform a silent install, EWLM logs these error messages in the installation log file. These messages are not displayed directly to you.</p> <p>On AIX, you can bypass this problem by directing the installation to another directory if the /temp directory does not have sufficient space. To do this, complete the following:</p> <ol style="list-style-type: none"> <li>1. Identify an alternate space to extract temporary files.</li> <li>2. Run the following command, where <i>alt</i> is your identified alternate space:  <code>-is:tempdir /alt</code></li> </ol>
After you install a domain manager fix package, the EWLM Control Center contains errors or does not contain the information as described in the fix package.	After you install a domain manager fix package, you must run the changeCCapp command as documented in "Fix package installation considerations for EWLM" on page 102.
Linux managed server install hangs.	<p>If the managed server install completes successfully on Linux or if the installation hangs on the panel that states "Choose Next to begin EWLM post-install setup", the installation may not complete successfully. You can view the startMS log file for errors. An example error that indicates this condition is 'Cannot find library ewljplat.'</p> <p>To recover from this error, cancel or close the installation request. Uninstall the EWLM managed server. Then, attempt the installation request again.</p>

Error	Resolution
<p>EWLM domain manager installation errors</p>	<p>Windows: Only one instance of the EWLM domain manager can be installed on a system. For example, you cannot install multiple instances of the EWLM domain manager onto the same system by specifying different installation paths. You can, however, create multiple configuration IDs for that domain manager.</p> <p>If you install more than one instance of the EWLM domain manager on the same system, the EWLM domain manager configuration will become corrupt and unusable. If this occurs, uninstall the EWLM domain manager. Then, attempt the install again.</p>
<p><i>Starting and stopping EWLM function</i></p>	
<p>Communication error between domain manager and managed server.</p> <p>This condition is documented in the diagnostics.xml log file as the following: eWLM Platform Service return code: -325</p>	<p>HP-UX and Solaris: If you start EWLM on an HP-UX or Solaris managed server, the start request appears successfully. However, if you encounter a communication error between the domain manager and managed server, you must install the LSOF utility on the managed server. This is a known problem.</p> <p>To recover from this error, you must install the LSOF utility on the managed server even if you do not plan on using the EWLM load balancing function. For more information on how to install this utility, see the following topics:</p> <ul style="list-style-type: none"> <li>• Installing LSOF utility for load balancing support on Solaris</li> <li>• Installing LSOF utility for load balancing support on HP-UX</li> </ul>
<p>Access denied when starting a managed server on a system running Windows.</p>	<p>If you attempt to start a managed server on a system running Windows and your user ID is not assigned to the proper authority, you receive an error message indicating that access is denied.</p> <p>To recover, you must log on with an user ID that has the proper authority for the platform on which you are working. To learn more about the proper authority for each platform, see user authorities.</p>
<p>Windows 2000 managed server does appear started after installing the managed server function or you receive either of the following conditions exist:</p> <p>Remote users receive error -6311 when starting an ARM application or</p> <p>Remote users receive error -6811 when starting the EWLM managed server agent</p>	<p>If the Windows 2000 server has "Terminal Services" running, it requires connected users of a remote session (for example, users connect to the system through Microsoft's Remote Desktop Connection or Terminal Service Client) to log off and log back in to actually bring the EWLM managed server in effect into the new remote session. It is particularly so, when the system does not prompt for a restart of the system. Restarting the server can also bring the EWLM managed server into effect.</p>

Error	Resolution
<p>The z/OS platform code is not found when starting the EWLM managed server on z/OS. EWLM displays the following error message for this condition:</p> <pre>ENTRY: ewlMConnect CEE3512S An HFS load of module ewlMplat.so failed. The system return code was 0000000157; the reason code was 0BDF03B2 From entry point Java_com_ibm_wlm_ns_StateServicesImplementationNative_ewlMConnect at compile unit offset +00000106 at entry offset +00000106 at address 0E89E536. 03B2 JRMVSPgmNotFound</pre>	<p>Add SYS1.SIEALNKE to the linklist concatenation.</p> <p>For more information, see z/OS MVS Initialization and Tuning Guide  (<a href="http://publibz.boulder.ibm.com/epubs/pdf/iea2e120.pdf">http://publibz.boulder.ibm.com/epubs/pdf/iea2e120.pdf</a>).</p> <p><b>Note:</b> You need Adobe Reader installed on your system to view or print this PDF. You can download a free copy from the Adobe Web site (<a href="http://www.adobe.com/products/acrobat/readstep.html">www.adobe.com/products/acrobat/readstep.html</a>) .</p>
<p>Problem starting EWLM on z/OS due to the SYS1.SIEALNKE not cataloged. The following error is displayed:</p> <pre>PROBLEM starting EWLM: CEE3512S An HFS load of module ewlMplat.so failed. The system return code was 0000000157; the reason code was 0BDF03B2. From entry point Java_com_ibm_wlm_ns_StateServicesImplementationNative_ewlMConnect at compile unit offset +000000CE at entry offset +000000CE at address 0ED66BCE. PROCESSING COMPLETE</pre>	<p>Ensure that SYS1.SIEALNKE is cataloged.</p> <p>For more information, see z/OS Language Environment Run-Time Messages  (<a href="http://publibz.boulder.ibm.com/epubs/pdf/ceea9150.pdf">http://publibz.boulder.ibm.com/epubs/pdf/ceea9150.pdf</a>)</p>
<p>Problem starting the EWLM managed server on z/OS. The following is an example error that you might encounter:</p> <pre>CEE3512S An HFS load of module ewlMplat.so failed. The system return code was 00 00000157; the reason code was 0BDF03B2. From entry point Java_com_ibm_wlm_ns_StateServicesImplementationNative_ewlMConnect at compile unit offset +000000CE at entry offset +000000CE at address 0ED66BCE. PROCESSING COMPLETE</pre>	<p>Then /usr/lib may not be in the EWLMMMS LIBPATH. Ensure that the LIBPATH variable in /etc/profile includes /usr/lib.</p> <p>For other general errors when starting EWLM on z/OS, you can review the following logs in /var/adm/ewlm/ewlmms/ for the specific failure information:</p> <ul style="list-style-type: none"> <li>ewlmmsErr.log</li> <li>ewlmmsOut.log</li> <li>createMS.log</li> </ul> <p>You may also need to review the job log for the started procedure to identify how to recover from the error.</p>
<p><b>Configuring EWLM</b></p>	
<p>Errors when configuring the managed server on z/OS such as the following:</p> <pre>EWLM customization error: loading customized properties ERROR WLMConfig error occurred There is only one file (db2j.log) in the /eWLMDData directory. The db2j.log file has many 'db2j... (Unknown Source)' entries along with an 'EDC8114I Address family not supported.' message, indicating the required UNIX Sockets (address family UDS) are not defined.</pre>	<p>The z/OS UNIX domain sockets are not defined. You should evaluate the values in your BPXPRMxx member of PARMLIB to ensure that they identify this type of file system. For more information, see “Updating the BPXPRMxx member in PARMLIB” on page 168.</p>



Error	Resolution
<p>Failure when you run an EWLM script in the z/OS UNIX shell environment. The following is the error message for this condition:</p> <pre>Warning: Failed to initialize Just-in-Time Compiler. Will use interpreter. JVMDG217: Dump Handler is Processing a Signal -  Please Wait. JVMHP002: JVM requesting System Transaction Dump JVMHP012: System Transaction Dump written to OEDFLTU.SYSTDUMP.D040812.T101233 JVMDG305: Java core not written, unable to allocate memory for print buffer.  JVMDG215: Dump Handler has Processed Exception Signal 11. CEE3204S The system detected a protection exception (System Completion Code=0C4) From compile unit /u/sovbl1d/cm1411/cm1411-200309 30/src/jit/pfm/s390/register_committed_code_md.c at entry point AtomicTestAndSet at statement 30 at compile unit offset +00000018 at entry offset +00000018 at address 7C4FB6B8.  PROCESSING COMPLETE</pre>	<p>The TSO user under which the script is run does not have a large enough region size.</p> <p>Log onto the z/OS system with a larger region size. For more information, see <a href="http://publibz.boulder.ibm.com/epubs/pdf/ikj4b450.pdf">z/OS TSO/E Customization </a> (<a href="http://publibz.boulder.ibm.com/epubs/pdf/ikj4b450.pdf">http://publibz.boulder.ibm.com/epubs/pdf/ikj4b450.pdf</a>).</p>
<p>General configuration error when using the EWLM configuration wizard.</p>	<p>Attempt the configuration again using the configuration script rather than the wizard. For more information on how to use the configuration scripts, also referred to as commands, see the task topic that is specific to the task that you want to perform. For example, see “Configuring a domain manager” on page 108 or “Configuring a managed server” on page 111.</p>
<p>Cannot close the help window for the EWLM configuration wizard.</p>	<p>If your window manager does not support an X button to close the help window, you may be able to close the window with a keyboard shortcut supported by your window manager, such as Alt +F4. Or, you may be able to minimize the window. If you are not able to close or minimize the help window, you may want to switch to a window manager that provides the required function.</p>

## Related information for EWLM





Listed here are the IBM Redbooks™ (in PDF format), manuals (in PDF format), Web sites, and information center topics that relate to the Enterprise Workload Manager topic. You can view or print any of the PDFs.




## IBM Redbooks

- IBM Enterprise Workload Manager V2.1   
View this book to learn more about IBM Enterprise Workload Manager and its partition management capabilities. This PDF includes a scenario that addresses installing, planning, setting up, configuring, and managing EWLM in an IT environment.
- IBM Enterprise Workload Manager  (3.8 MB)  
View this book to learn more about IBM Enterprise Workload Manager. This PDF includes a scenario that addresses installing, planning, setting up, configuring, and managing EWLM in an IT environment.

## Web sites

- IBM HTTP Server   
(<http://www-306.ibm.com/software/webservers/httpservers/library/>)  
View this site to learn more about the IBM HTTP Server which is an IBM-instrumented application that EWLM can monitor.
- WebSphere Application Server v5.1 Information Center   
(<http://publib.boulder.ibm.com/infocenter/ws51help/>)  
View this site to learn more about the WebSphere Application Server, which is an ARM-instrumented application that EWLM can monitor.
- WebSphere Application Server v6.0 Information Center   
(<http://publib.boulder.ibm.com/infocenter/ws60help/>)  
View this site to learn more about WebSphere Application Server 6.0, which is an ARM-instrumented application that EWLM can monitor.
- Virtualization Engine Systems Services Fixes Web site   
(<http://www14.software.ibm.com/webapp/set2/sas/f/VirtualizationEngine/>)  
View this site ensure that you have installed the required fixes for the Virtualization Engine system services. The fixes include operating system fixes that are required for each system service as well as application-level fixes that are required for each system service to function properly.

## Education

- IBM Training - IT Education Services   
(<http://ibm.com/training>)  
View this site to find information about educational activities related to EWLM. One course in particular is Enterprise Workload Manager Planning and Implementation (Course Code OZ200).

## Other information

- Virtualization Engine
- Fixes and updates
- EWLM APIs
- IBM Virtualization Engine Suite Performance Reference

## | Technical Support


- | • Contact your EWLM sales representative for customer support. For IBM support, call 1-800-IBM-SERV.

## Saving PDF files

To save a PDF on your workstation for viewing or printing:

1. Right-click the PDF in your browser (right-click a link above).
2. Click the option that saves the PDF locally.
3. Navigate to the directory in which you want to save the PDF.
4. Click **Save**.

## Downloading Adobe Acrobat Reader

You need Adobe Acrobat Reader installed on your system to view or print these PDFs. You can download a copy from the Adobe Web site ([www.adobe.com/products/acrobat/readstep.html](http://www.adobe.com/products/acrobat/readstep.html)) .

## Accessibility features for EWLM

Accessibility features help users who have a physical disability, such as restricted mobility or limited vision, to use information technology products successfully. IBM strives to provide products with usable access for everyone, regardless of age or ability.

### EWLM Control Center: Accessibility features

The following list includes the major accessibility features in the EWLM Control Center. These features support:

- You can use screen-reader software and a digital speech synthesizer to hear what is displayed on the monitor.
- You can operate all features using the keyboard instead of the mouse.
- You can export the real-time performance monitors and topology charts to a text version of the data.

**Tip:** The IBM Systems Software Information Center and its related publications are accessibility-enabled for the IBM Home Page Reader. You can operate all features using the keyboard instead of the mouse.

### EWLM Control Center: Keyboard navigation

The EWLM Control Center uses your Web browser's keyboard shortcuts. For a list of keyboard shortcuts, see your browser's accessibility information.

### EWLM Control Center: Interface information

EWLMs real-time performance monitors and topology charts provide data using a graphical format. For accessibility purposes, you can download a text version of this data that screen-reader software can read. EWLM saves the data in a Microsoft Excel Comma Separated Values (CSV) file. This file contains the x- and y- coordinates for each value on the graph.

To download a text version of graphical data, click **Export** in the EWLM Control Center.

### EWLM Configuration Wizard: Accessibility features

The following list includes the major accessibility features in the EWLM Control Center. These features support:

- You can use screen-reader software and a digital speech synthesizer to hear what is displayed on the monitor.
- You can operate all features using the keyboard instead of the mouse.

**Tip:** The IBM Systems Software Information Center and its related publications are accessibility-enabled for the IBM Home Page Reader. You can operate all features using the keyboard instead of the mouse.

In addition, the EWLM Configuration Wizard supports the operating system settings for the wizard's size, font, and color.

**Note:** The EWLM Configuration Wizard help is not accessible.

## EWLM Control Center: Keyboard navigation

The following table describes the keyboard shortcuts for the EWLM Configuration Wizard.

*Table 87. EWLM Configuration Wizard keyboard shortcuts*

Action	Shortcut
Move to the next option	Tab
Move to the previous option	Shift+Tab
Select the current option	Space
Navigate out of a table to the next component	Navigate out of a table to the next component
Navigate out of a table to the next component	Navigate out of a table to the next component
Navigate out of a table the previous component	Shift+Ctrl+Tab
Activate the Next or Finish buttons	Enter
Activate the Back, Cancel, or Help buttons	Navigate to the button by pressing Tab or Shift+Tab, then press Space.
Move among options in a selection list	Up arrow key or Down arrow key
Move among tabbed sections in a page	Left arrow key or Right arrow key

## EWLM Control Center: Interface information

The EWLM Configuration Wizard provides you with a graphical interface to configure the EWLM. As an alternative to the configuration wizard, you can use the EWLM commands. The help for the configuration commands can be read by screen-reader software. For more information on the configuration commands, see the Configure EWLM information in the EWLM topic collection.

## Vendor software

EWLM includes certain vendor software that is not covered under the IBM license agreement. IBM makes no representation about the accessibility features of these products. Contact the vendor for the accessibility information about its products.

## IBM and accessibility

See the *IBM Accessibility Center* for more information about the commitment that IBM has to accessibility.



---

## Appendix. Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

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

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation  
Licensing  
2-31 Roppongi 3-chome, Minato-ku  
Tokyo 106-0032, Japan

**The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:** INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation

Software Interoperability Coordinator, Department YBWA  
3605 Highway 52 N  
Rochester, MN 55901  
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this information and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, IBM License Agreement for Machine Code, or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

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 and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

#### COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. \_enter the year or years\_. All rights reserved.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

---

## Programming Interface Information

This Enterprise Workload Manager publication documents intended Programming Interfaces that allow the customer to write programs to obtain the services of the specified products.

---

## Trademarks

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

AIX  
AIX 5L  
DB2  
DB2 Universal Database  
eServer  
i5/OS  
iSeries  
IBM  
IBM (logo)  
iSeries  
Language Environment  
MVS  
OpenPower  
POWER  
pSeries  
Virtualization Engine  
WebSphere  
xSeries  
z/OS  
zSeries

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

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

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

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

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

---

## Terms and conditions

Permissions for the use of these publications is granted subject to the following terms and conditions.

**Personal Use:** You may reproduce these publications for your personal, noncommercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative works of these publications, or any portion thereof, without the express consent of IBM.

**Commercial Use:** You may reproduce, distribute and display these publications solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of these publications, or reproduce, distribute or display these publications or any portion thereof outside your enterprise, without the express consent of IBM.

Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the publications or any information, data, software or other intellectual property contained therein.

IBM reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the publications is detrimental to its interest or, as determined by IBM, the above instructions are not being properly followed.

You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations.

IBM MAKES NO GUARANTEE ABOUT THE CONTENT OF THESE PUBLICATIONS. THE PUBLICATIONS ARE PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.







Printed in USA