AppManager for JAVA on UNIX and Linux Servers

Management Guide

June 2015

Legal Notice

Contents

About this Book and the Library

The NetIQ AppManager product (AppManager) is a comprehensive solution for managing, diagnosing, and analyzing performance, availability, and health for a broad spectrum of operating environments, applications, services, and server hardware.

AppManager provides system administrators with a central, easy-to-use console to view critical server and application resources across the enterprise. With AppManager, administrative staff can monitor computer and application resources, check for potential problems, initiate responsive actions, automate routine tasks, and gather performance data for real-time and historical reporting and analysis.

Intended Audience

This guide provides information for individuals responsible for installing an AppManager module and monitoring specific applications with AppManager.

Other Information in the Library

The library provides the following information resources:

Installation Guide for AppManager

Provides complete information about AppManager pre-installation requirements and step-by-step installation procedures for all AppManager components.

User Guide for AppManager Control Center

Provides complete information about managing groups of computers, including running jobs, responding to events, creating reports, and working with Control Center. A separate guide is available for the AppManager Operator Console.

Administrator Guide for AppManager

Provides information about maintaining an AppManager management site, managing security, using scripts to handle AppManager tasks, and leveraging advanced configuration options.

Upgrade and Migration Guide for AppManager

Provides complete information about how to upgrade from a previous version of AppManager.

Management guides

Provide information about installing and monitoring specific applications with AppManager.

Help

Provides context-sensitive information and step-by-step guidance for common tasks, as well as definitions for each field on each window.

The AppManager library is available in Adobe Acrobat (PDF) format from the AppManager Documentation page of the NetIQ website.

About NetIQ Corporation

We are a global, enterprise software company, with a focus on the three persistent challenges in your environment: Change, complexity and risk—and how we can help you control them.

Our Viewpoint

Adapting to change and managing complexity and risk are nothing new

In fact, of all the challenges you face, these are perhaps the most prominent variables that deny you the control you need to securely measure, monitor, and manage your physical, virtual, and cloud computing environments.

Enabling critical business services, better and faster

We believe that providing as much control as possible to IT organizations is the only way to enable timelier and cost effective delivery of services. Persistent pressures like change and complexity will only continue to increase as organizations continue to change and the technologies needed to manage them become inherently more complex.

Our Philosophy

Selling intelligent solutions, not just software

In order to provide reliable control, we first make sure we understand the real-world scenarios in which IT organizations like yours operate — day in and day out. That's the only way we can develop practical, intelligent IT solutions that successfully yield proven, measurable results. And that's so much more rewarding than simply selling software.

Driving your success is our passion

We place your success at the heart of how we do business. From product inception to deployment, we understand that you need IT solutions that work well and integrate seamlessly with your existing investments; you need ongoing support and training post-deployment; and you need someone that is truly easy to work with — for a change. Ultimately, when you succeed, we all succeed.

Our Solutions

- Identity & Access Governance
- Access Management
- Security Management
- Systems & Application Management
- Workload Management
- Service Management

Contacting Sales Support

For questions about products, pricing, and capabilities, contact your local partner. If you cannot contact your partner, contact our Sales Support team.

Worldwide:	www.netiq.com/about_netiq/officelocations.asp
United States and Canada:	1-888-323-6768
Email:	info@netiq.com
Website:	www.netiq.com

Contacting Technical Support

For specific product issues, contact our Technical Support team.

Worldwide:	www.netiq.com/support/contactinfo.asp
North and South America:	1-713-418-5555
Europe, Middle East, and Africa:	+353 (0) 91-782 677
Email:	support@netiq.com
Website:	www.netiq.com/support

Contacting Documentation Support

Our goal is to provide documentation that meets your needs. The documentation for this product is available on the NetIQ website in HTML and PDF formats on a page that does not require you to log in. If you have suggestions for documentation improvements, click **comment on this topic** at the bottom of any page in the HTML version of the documentation posted at www.netiq.com/documentation. You can also email Documentation-Feedback@netiq.com. We value your input and look forward to hearing from you.

Contacting the Online User Community

NetIQ Communities, the NetIQ online community, is a collaborative network connecting you to your peers and NetIQ experts. By providing more immediate information, useful links to helpful resources, and access to NetIQ experts, NetIQ Communities helps ensure you are mastering the knowledge you need to realize the full potential of IT investments upon which you rely. For more information, visit http://community.netiq.com.

1 Introducing AppManager for Java on UNIX and Linux Servers

AppManager for JAVA on UNIX and Linux Servers (AM JAVA UNIX) is a product for monitoring JAVA resource usage on remote UNIX, Linux, and Windows servers.

AM JAVA UNIX offers the following:

Classes Loaded and Unloaded

Monitor the loading and unloading of classes on a Java Virtual Machine (JVM) by collecting data for loaded and unloaded class counts.

Connector Statistics

Monitor connector request and error statistics for Apache Tomcat and JBoss servers. Set thresholds and raise an event when the request rate or error rate exceeds its threshold. Collect data for bytes sent per minute, request count per minute, and error count per minute.

CPU Utilization

Monitor JVM process CPU utilization and system CPU utilization. Set thresholds and raise an event when the JVM process CPU utilization or system CPU utilization exceeds its threshold. Collect data for JVM process CPU utilization and system CPU utilization.

Garbage Collector Statistics

Monitor JVM garbage collector statistics. Set thresholds for the following:

- Maximum time in milliseconds for a garbage collector run
- Garbage collector run count

Raise an event when either threshold is exceeded. Collect data for garbage collection duration and the number of garbage collector runs between Knowledge Script intervals.

Memory Heap Utilization

Monitor the memory heap utilization for a JVM. Set a threshold and raise an event when the used heap as a percentage of committed heap exceeds the threshold.Collect data for JVM heap and non-heap usage.

JBoss Transaction Statistics

Monitor transaction rates for a JBoss server. Set thresholds for the following counters:

- in-flight transactions
- committed transactions per minute
- transactions created per minute
- transactions aborted per minute
- transactions rolled back per minute due to application failure
- transactions rolled back per minute due to resource failure

Collect data for each counter. Raise an event when any of the counters exceeds a threshold you set.

Jetty Queued Thread Pool

Monitor the queued thread count for a Jetty server. Set a threshold for a percentage of the Jetty maximum thread count and raise an event when the queued thread count exceeds the threshold. Collect data for current thread count, maximum thread count, and idle thread count.

Managed Bean Utility

Query individual Java bean counters. Specify strings to find or not find in the query output. Set minimum and maximum values for numeric data in the query. Raise an event when the query result:

- Contains or does not contain the specified strings
- Contains numeric data that falls outside the minimum and maximum value thresholds

Collect data for each numeric counter. Raise an unconditional event with the text string value of the managed bean counter.

Memory Pool Utilization

Monitor memory pool utilization for a JVM. Set a threshold for the maximum percentage of pool utilization and raise an event when JVM memory pool utilization exceeds the threshold. Collect data for JVM memory pools utilization and maximum value.

Thread Utilization

Monitor Java process thread utilization. Collect data for JVM thread count, daemon thread count, and peak thread count.

Tomcat Application Sessions

Monitor application session statistics for Apache Tomcat servers. Set thresholds for the maximum number of sessions created per minute and the maximum number of sessions expired per minute and raise an event when session creations or session expirations exceeds the threshold. Collect data for active and expired sessions.

Tomcat Thread Pools

Monitor thread pool statistics for Apache Tomcat servers. Set maximum percentage thresholds for current and busy threads and raise an event when the current or busy thread count exceeds the threshold. Collect data for current thread count, maximum thread count, and busy thread count as a percentage of the current thread count.

2 Installing AppManager for JAVA on UNIX and Linux Servers

This chapter provides installation instructions and describes system requirements for AM JAVA UNIX.

This chapter assumes you have AppManager installed. For more information about installing AppManager or about AppManager system requirements, see the *Installation Guide for AppManager*, which is available on the AppManager Documentation page.

2.1 System Requirements

For the latest information about supported software versions and the availability of module updates, visit the AppManager Supported Products page. Unless noted otherwise, this module supports all updates, hotfixes, and service packs for released products.

AM JAVA UNIX has the following system requirements.

Item	Requirement
NetIQ AppManager installed on the AppManager repository (QDB) computers, on the UNIX agent proxy computers (AppManager agents), and on all console computers.	7.0.1 or later
NetIQ UNIX Agent	7.1 or later
Jolokia agent	1.2.3 or later
Operating system on the UNIX agent computers	One of the following: ◆ CentOS ◆ HP-UX ◆ IBM AIX ◆ Oracle Linux ◆ Oracle Solaris (SPARC) ◆ Red Hat Enterprise Linux ◆ SUSE Linux Enterprise Server

If you encounter problems using this module with a later version of your application, contact NetIQ Technical Support.

2.2 Prerequisites

This module includes a Jolokia web archive (WAR) agent that must be deployed to each JVM you want to monitor. For more information about Jolokia, see jolokia.org.

You can deploy the Jolokia WAR agent in a servlet container like other Java Enterprise Edition (JEE) web applications. Installation often consists of copying the Jolokia WAR agent to the Java deployment directory on the target server.

Some platforms require you to use a visual administrative application or command line tool to deploy the Jolokia WAR agent. Consult the documentation for each of your platforms for the correct deployment procedure.

This module has been tested on supported JVM products using the Jolokia WAR agent version 1.2.3.

2.3 Installing the Module

Run the module installer only once on any Windows computer. The module installer automatically identifies and updates all relevant AppManager components on a computer.

Access the `AM70-Java-8.x.x.0.msi` module installer from the `AM70_Java_8.x.x.0` self-extracting installation package on the AppManager Module Upgrades & Trials page.

You can install the Knowledge Scripts into local or remote AppManager repositories (QDBs). Install these components only once per QDB.

The module installer now installs Knowledge Scripts for each module directly into the QDB instead of to the `\AppManager\qdb\kp` folder as in previous releases of AppManager.

To install the module manually:

1. Double-click the module installer `.msi` file.
2. Accept the license agreement.
3. Run the module installer on all console computers to install the Help and console extensions.
4. To install the Knowledge Scripts into the QDB:
   - **4a** Select **Install Knowledge Scripts** to install the repository components, including the Knowledge Scripts.
   - **4b** Specify the SQL Server name of the server hosting the QDB, as well as the case-sensitive QDB name.
5. Run the module installer only for the primary QDB. Control Center automatically replicates this module to secondary QDBs.
6. (Conditional) If you have not discovered Java UNIX or Linux resources, run the Discovery_Java Knowledge Script on all UNIX or Linux agent computers where you installed the module. For more information, see Section 2.5, “Discovering AppManager for JAVA Resources,” on page 13.
7. To get the updates provided in this release, upgrade any running Knowledge Script jobs. For more information, see Section 2.6, “Upgrading Knowledge Script Jobs,” on page 14.

After the installation has completed, the `Java_Install.log` file, located in the `\NetIQ\Temp\NetIQ_Debug\<`*ServerName*`>` folder, lists any problems that occurred.

2.4 Silently Installing the Module

To silently (without user intervention) install a module using the default settings, run the following command from the folder in which you saved the module installer:

```
msiexec.exe /i "AM70-Java-8.x.x.0.msi" /qn
```

where *x.x* is the actual version number of the module installer.

To create a log file that describes the operations of the module installer, add the following flag to the command noted above:

```
/L* "AM70-Java-8.x.x.0.msi.log"
```

The log file is created in the directory in which you saved the module installer.

To silently install the module on a remote AppManager repository, you can use Windows authentication or SQL authentication.

Windows authentication:

```
AM70-Java-8.x.x.0.msi /qn MO_B_QDBINSTALL=1 MO_B_SQLSVR_WINAUTH=1
MO_SQLSVR_NAME=[SQL Server Name] MO_QDBNAME=[AM-Repository Name]
```

SQL authentication:

```
AM70-Java-8.x.x.0.msi /qn MO_B_QDBINSTALL=1 MO_B_SQLSVR_WINAUTH=0
MO_SQLSVR_USER=[SQL login] MO_SQLSVR_PWD=[SQL Login Password] MO_SQLSVR_NAME=[SQL
Server Name] MO_QDBNAME=[AM-Repository Name]
```

2.5 Discovering AppManager for JAVA Resources

Use the Discovery_JAVA Knowledge Script to discover Java resources installed on local or remote UNIX, Linux, and Windows servers. This Knowledge Script returns information about successful, failed, and partial discoveries, and it raises events to notify you of errors.

Run the Discovery_JAVA script on AppManager UNIX agents.

By default, this script is scheduled to run once for each computer.

Set the following parameters as needed:

Parameter	How to Set It
JVM host settings	
Jolokia URL address	Specify the Jolokia agent URL address from where you want to discover Java resources. The default is a null string. **NOTE** ◆ AM JAVA UNIX only supports the HTTP protocol for the Jolokia URL ◆ URL redirection is not supported ◆ The URL address cannot exceed 113 characters in length
Timeout for AppManager to fetch metrics from JVM	Set the number of seconds, from 5 to 1000, that AppManager will wait for metrics to return from a Java virtual machine. The default is 600. **NOTE:** The other AM JAVA UNIX Knowledge Scripts use this timeout value to determine how long to wait when getting metrics.
Event Settings	
Event severity when module error or job fails	Specify the event severity, from 1 to 40, to indicate the importance of an event in which this module experiences an error or the job fails. The default is 5.
Raise event when discovery succeeds?	Set to `Yes` to raise an event when Java resource discovery succeeds. The default is no.

Parameter	How to Set It
Event severity	Specify the event severity, from 1 to 40, to indicate the importance of an event in which Java resource discovery succeeds. The default is 25.
Raise event when discovery partially succeeds?	Set to `Yes` to raise an event when Java resource discovery partially succeeds. The default is yes.
Event severity	Specify the event severity, from 1 to 40, to indicate the importance of an event in which Java resource discovery partially succeeds. The default is 15.
Discovery Settings	
Discover managed beans?	Set to `Yes` to discover managed Java beans. The default is yes.
Discover counters?	Set to `Yes` to discover Java bean counters. The default is no. **NOTE:** Discovery supports composite data types for complex counters. Other complex counter types, for example tabular and string array, are not supported. **CAUTION:** Enabling this option can cause a console performance issue as there may be a large number of objects to process and display.
Regular expression to include managed bean domain names	Specify a comma-separated list of managed bean domain names to include in discovery. You can use regular expressions in the names. The default is a null string. **NOTE:** The include filter takes precedence when both an include and exclude filter are specified.
Regular expression to exclude managed bean domain names	Specify a comma-separated list of managed bean domain names to exclude from discovery. You can use regular expressions in the names. The default is `jolokia|jmx4perl`.

2.6 Upgrading Knowledge Script Jobs

This release of AppManager for JAVA on UNIX and Linux Servers may contain updated Knowledge Scripts. You can push the changes for updated scripts to running Knowledge Script jobs in one of the following ways:

- Use the AMAdmin_UpgradeJobs Knowledge Script.
- Use the Properties Propagation feature.

2.6.1 Running AMAdmin_UpgradeJobs

The AMAdmin_UpgradeJobs Knowledge Script can push changes to running Knowledge Script jobs.

Upgrading jobs to use the most recent script version allows the jobs to take advantage of the latest script logic while maintaining existing parameter values for the job.

For more information, see the Help for the AMAdmin_UpgradeJobs Knowledge Script.

2.6.2 Propagating Knowledge Script Changes

You can propagate script changes to jobs that are running and to Knowledge Script Groups, including recommended Knowledge Script Groups and renamed Knowledge Scripts.

Before propagating script changes, verify that the script parameters are set to your specifications. Customized script parameters may have reverted to default parameters during the installation of the module. New parameters may need to be set appropriately for your environment or application.

You can choose to propagate only properties (specified in the Schedule and Values tabs), only the script (which is the logic of the Knowledge Script), or both. Unless you know specifically that changes affect only the script logic, you should propagate both properties and the script.

For more information about propagating Knowledge Script changes, see the “Running Monitoring Jobs” chapter of the *Operator Console User Guide for AppManager*.

Propagating Changes to Ad Hoc Jobs

You can propagate the properties and the logic (script) of a Knowledge Script to ad hoc jobs started by that Knowledge Script. Corresponding jobs are stopped and restarted with the Knowledge Script changes.

To propagate changes to ad hoc Knowledge Script jobs:

1. In the Knowledge Script view, select the Knowledge Script for which you want to propagate changes.
2. Click **Properties propagation** > **Ad Hoc Jobs**.
3. Select the components of the Knowledge Script that you want to propagate to associated ad hoc jobs:

Select	To propagate
Script	The logic of the Knowledge Script.
Properties	Values from the Knowledge Script Schedule and Values tabs, such as schedule, monitoring values, actions, and advanced options.

Propagating Changes to Knowledge Script Groups

You can propagate the properties and logic (script) of a Knowledge Script to corresponding Knowledge Script Group members.

After you propagate script changes to Knowledge Script Group members, you can propagate the updated Knowledge Script Group members to associated running jobs. For more information, see “Propagating Changes to Ad Hoc Jobs” on page 15.

To propagate Knowledge Script changes to Knowledge Script Groups:

1. In the Knowledge Script view, select the Knowledge Script Group for which you want to propagate changes.
2. On the **KS** menu, select **Properties propagation** > **Ad Hoc Jobs**.
3. (Condition) If you want to exclude a Knowledge Script member from properties propagation, deselect that member from the list in the Properties Propagation dialog box.

4 Select the components of the Knowledge Script that you want to propagate to associated Knowledge Script Groups:

Select	To propagate
Script	The logic of the Knowledge Script.
Properties	Values from the Knowledge Script Schedule and Values tabs, including the schedule, actions, and Advanced properties.

5 Click **OK**. Any monitoring jobs started by a Knowledge Script Group member are restarted with the job properties of the Knowledge Script Group member.

3 Java Knowledge Scripts

AppManager provides the following Knowledge Scripts for monitoring Java on UNIX and Linux servers.

From the Knowledge Script view of Control Center, you can access more information about any Knowledge Script by selecting it and clicking **Help**. In the Operator Console, select any Knowledge Script in the Knowledge Script pane and press **F1**.

Knowledge Script	What It Does
ClassLoadingStats	Monitors the loading and unloading of classes on a Java Virtual Machine (JVM)
ConnectorStats	Monitors connector statistics for Apache Tomcat and JBoss servers
CPUUtil	Monitors JVM process CPU utilization and system CPU utilization
GarbageCollectorStats	Monitors JVM garbage collector run statistics
HeapUtil	Monitors memory heap utilization for a JVM
JBossTransactionStats	Monitors transaction rates for a JBoss server
JettyQueuedThreadPool	Monitors the queued thread count for a Jetty server
ManagedBeanUtil	Queries individual managed bean counters
MemoryPools	Monitors memory pool utilization on a JVM
ThreadUtil	Monitors Java process thread utilization
TomcatAppSessions	Monitors application session statistics for Apache Tomcat servers
TomcatThreadPoolStats	Monitors thread pool statistics for Apache Tomcat servers

3.1 ClassLoadingStats

Use this Knowledge Script to monitor the Java loaded and unloaded class count.

3.1.1 Resource Objects

Java virtual machine object

3.1.2 Default Schedule

The default interval for this script is **Every 15 minutes**.

3.1.3 Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event settings	
Event severity when job fails	Specify the event severity, from 1 to 40, to indicate the importance of an event in which this job fails. The default is 5.
Raise event when AppManager fails to get metrics?	Set to `Yes` to raise an event if AppManager cannot collect any information about Java classes loaded and unloaded. The default is yes.
Event severity	Specify the event severity, from 1 to 40, to indicate the importance of an event in which AppManager cannot collect any information about Java classes loaded and unloaded. The default is 5.
Data collection options	
Collect data for loaded class count?	Set to `Yes` to collect data for the Java loaded class count. This is the number of classes on the JVM at the moment the metric was captured. The default is yes.
Collect data for unloaded class count?	Set to `Yes` to collect data for the Java unloaded class count. This is the cumulative number of classes unloaded by the JVM since it was last started. The default is yes.

3.2 ConnectorStats

Use this Knowledge Script to monitor statistics for all Global Request Processors in the Apache Tomcat server and each Web subsystem connector in the JBoss server. This script raises an event when the request count or error count exceeds the thresholds you set.

NOTE: This Knowledge Script cannot be used to monitor Mbeans on JBoss version 8.0.0 or later. For more information see developer.jboss.org/message/869273.

3.2.1 Resource Objects

Java virtual machine object

3.2.2 Default Schedule

The default interval for this script is **Every 15 minutes**.

3.2.3 Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event settings	

Parameter	How to Set It
Event severity when job fails	Specify the event severity, from 1 to 40, to indicate the importance of an event in which this job fails. The default is 5.
Raise event when AppManager fails to get metrics?	Set to `Yes` to raise an event if AppManager cannot collect connector statistics. The default is yes.
Event severity	Specify the event severity, from 1 to 40, to indicate the importance of an event in which AppManager cannot collect connector statistics. The default is 5.
Raise event when request rate in requests per minute exceeds threshold?	Set to `Yes` to raise an event if the connector requests per minute rate exceeds the threshold you set. The default is yes.
Event severity	Specify the event severity, from 1 to 40, to indicate the importance of an event in which the connector requests per minute rate exceeds the threshold you set. The default is 5.
Threshold - Request Rate	Enter the threshold value, from 0 to 2,147,483,647, for the connector requests per minute rate. AppManager raises an event if the connector requests per minute rate exceeds this threshold. The default is 80.
Raise event when error rate in errors per minute exceeds threshold?	Set to `Yes` to raise an event if the connector errors per minute rate exceeds the threshold you set. The default is yes.
Event severity	Specify the event severity, from 1 to 40, to indicate the importance of an event in which the connector errors per minute rate exceeds the threshold you set. The default is 5.
Threshold - Error Rate	Enter the threshold value, from 0 to 2,147,483,647, for the connector errors per minute rate. AppManager raises an event if the connector errors per minute rate exceeds this threshold. The default is 5.
Filter settings	
Exclude filter for Connectors	Enter the names of connectors to exclude from connector statistics monitoring. You can use regular expressions to specify connector names.The default is a null string.
Data collection options	
Collect data for bytes sent rate?	Set to `Yes` to collect data for connector bytes sent per minute. The default is no.
Collect data for request count rate?	Set to `Yes` to collect data for the connector request count per minute. The default is no.
Collect data for error count rate?	Set to `Yes` to collect data for the connector error count per minute. The default is no.

3.3 CPUUtil

Use this Knowledge Script to monitor JVM CPU utilization and system CPU utilization. This script raises an event when JVM CPU utilization or system CPU utilization exceeds the threshold you set.

NOTE

- This Knowledge Script supports JVM version 1.7 and later
- This Knowledge Script ignores any invalid CPU utilization values provided by the JVM

3.3.1 Resource Objects

Java virtual machine object

3.3.2 Default Schedule

The default interval for this script is **Every 5 minutes**.

3.3.3 Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event settings	
Event severity when job fails	Specify the event severity, from 1 to 40, to indicate the importance of an event in which this job fails. The default is 5.
Raise event when AppManager fails to get metrics?	Set to `Yes` to raise an event if AppManager cannot collect any information about CPU utilization. The default is yes.
Event severity	Specify the event severity, from 1 to 40, to indicate the importance of an event in which AppManager cannot collect any information about CPU utilization. The default is 5.
Raise event when JVM process CPU utilization exceeds threshold?	Set to `Yes` to raise an event when the JVM process exceeds the process CPU utilization threshold you set. The default is yes.
Event severity	Specify the event severity, from 1 to 40, to indicate the importance of an event in which the JVM process CPU utilization exceeds the threshold you set. The default is 5.
Threshold in percentage	Enter the threshold value, from 0 to 100, for the JVM process CPU utilization percentage. AppManager raises an event if the JVM process CPU utilization exceeds this threshold. The default is 80%.
Raise event when system CPU utilization exceeds threshold?	Set to `Yes` to raise an event in which the system CPU utilization exceeds the threshold you set. The default is yes.
Event severity	Specify the event severity, from 1 to 40, to indicate the importance of an event in which the system CPU utilization exceeds the threshold you set. The default is 5.

Parameter	How to Set It
Threshold in percentage	Enter the threshold value, from 0 to 100, for the percentage of system CPU utilization. AppManager raises an event if the system CPU utilization exceeds this threshold. The default is 80%.
Data collection options	
Collect data for JVM process CPU utilization in percent?	Set to `Yes` to collect data for JVM process CPU utilization. The default is no.
Collect data for system CPU utilization in percent?	Set to `Yes` to collect data for system CPU utilization. The default is no.

3.4 GarbageCollectorStats

Use this Knowledge Script to monitor Java Virtual Machine (JVM) garbage collection statistics. This script raises an event when:

- **Garbage collection takes too long:** A JVM garbage collection run takes longer than the millisecond threshold you set
- **Garbage collection occurs too often:** The number of JVM garbage collection runs that occur between Knowledge Script iterations exceeds the threshold you set

3.4.1 Resource Objects

Java virtual machine object

3.4.2 Default Schedule

The default interval for this script is **Every 15 minutes**.

3.4.3 Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event settings	
Event severity when job fails	Specify the event severity, from 1 to 40, to indicate the importance of an event in which this job fails. The default is 5.
Raise event when AppManager fails to get metrics?	Set to `Yes` to raise an event if AppManager cannot collect any information about JVM garbage collection. The default is yes.
Event severity	Specify the event severity, from 1 to 40, to indicate the importance of an event in which AppManager cannot collect any information about JVM garbage collection. The default is 5.
Raise event when garbage collector duration exceeds threshold?	Set to `Yes` to raise an event when the JVM garbage collector duration exceeds the threshold you set. The default is yes.

Parameter	How to Set It
Event severity	Specify the event severity, from 1 to 40, to indicate the importance of an event in which the JVM garbage collection duration exceeds the threshold you set. The default is 5.
Threshold for garbage collector duration	Enter the threshold value in milliseconds (ms), from 0 to 2,147,483,647, for the JVM garbage collection duration. AppManager raises an event if the JVM garbage collection duration exceeds this threshold. The default is 100.
Raise event when number of garbage collector runs between knowledge script interval exceeds threshold?	Set to `Yes` to raise an event when the number of garbage collector runs between Knowledge Script intervals exceeds the threshold you set. The default is yes.
Event severity	Specify the event severity, from 1 to 100, to indicate the importance of an event in which the number of garbage collector runs between Knowledge Script intervals exceeds the threshold you set. The default is 5.
Threshold for garbage collector collection run count	Enter the threshold value, from 0 to 2,147,483,647, for the maximum garbage collector runs between Knowledge Script intervals. AppManager raises an event if the number of garbage collector runs exceeds this count. The default is 10.
Filter settings	
Exclude filter for garbage collector names	Enter the names of garbage collectors to exclude from garbage collector monitoring. You can use regular expressions to specify garbage collector names. The default is a null string.
Data collection options	
Collect data for garbage collection duration?	Set to `Yes` to collect data for garbage collector duration. The default is no.
Collect data for number of garbage collector runs between Knowledge Script intervals?	Set to `Yes` to collect data for the number of garbage collector runs that occur each interval between Knowledge Script executions. The default is no.

3.5 HeapUtil

Use this Knowledge Script to monitor Java heap utilization. This script raises an event when the percentage of JVM heap used exceeds the threshold you set.

3.5.1 Resource Objects

Java virtual machine object

3.5.2 Default Schedule

The default interval for this script is **Every 5 minutes**.

3.5.3 Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event settings	
Event severity when job fails	Specify the event severity, from 1 to 40, to indicate the importance of an event in which this job fails. The default is 5.
Raise event when AppManager fails to get metrics?	Set to `Yes` to raise an event if AppManager cannot collect any information about JVM heap utilization. The default is yes.
Event severity	Specify the event severity, from 1 to 40, to indicate the importance of an event in which AppManager cannot collect any information about JVM heap utilization. The default is 5.
Raise event when JVM heap memory usage exceeds threshold?	Set to `Yes` to raise an event if used heap exceeds the threshold you set. The default is yes.
Event severity	Specify the event severity, from 1 to 40, to indicate the importance of an event in which used heap exceeds the threshold you set. The default is 5.
Threshold - JVM Used Heap in Committed Heap	Enter the threshold value, from 1 to 100, for used heap as a percentage of the committed heap. AppManager raises an event if the used heap as a percentage of committed heap exceeds this threshold. The default is 80%.
Data collection options	
Collect data for JVM heap usage?	Set to `Yes` to collect data for heap memory usage. The default is no.
Collect data for JVM non-heap usage?	Set to `Yes` to collect data for non-heap memory usage. The default is no.

3.6 JBossTransactionStats

Use this Knowledge Script to monitor transaction statistics for JBoss servers. This script can raise events when any of the following exceeds the thresholds you set:

- created transactions
- in-flight transactions
- committed transactions
- aborted transactions
- transactions rolled back at application request
- transactions rolled back due to resource failure

NOTE: This Knowledge Script requires that the JBoss transaction statistics are enabled.

3.6.1 Resource Objects

Java virtual machine object

3.6.2 Default Schedule

The default interval for this script is **Every 15 minutes**.

3.6.3 Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event settings	
Event severity when job fails	Specify the event severity, from 1 to 40, to indicate the importance of an event in which this job fails. The default is 5.
Raise event when AppManager fails to get metrics?	Set to `Yes` to raise an event if AppManager cannot collect any information about JBoss transactions. The default is yes.
Event severity	Specify the event severity, from 1 to 40, to indicate the importance of an event in which AppManager cannot collect any information about JBoss transactions. The default is 5.
Raise event when transactions creation rate exceeds threshold?	Set to `Yes` to raise an event when the number of transactions created per minute exceeds the threshold you set.The default is yes.
Event severity	Specify the event severity, from 1 to 40, to indicate the importance of an event in which the number of transactions created per minute exceeds the threshold you set. The default is 5.
Threshold - Transactions creation rate	Enter the threshold value, from 0 to 2,147,483,647, for the number of transactions created per minute. AppManager raises an event when the number of transactions created per minute exceeds this threshold. The default is 100.
Raise event when In-flight transactions count exceeds threshold?	Set to `Yes` to raise an event when the in-flight transaction count exceeds the threshold you set. The default is yes.
Event severity	Specify the event severity, from 1 to 40, to indicate the importance of an event in which the in-flight transaction count exceeds the threshold you set. The default is 5.
Threshold - In-flight transactions count	Enter the threshold value, from 0 to 2,147,483,647, for the in-flight transaction count. AppManager raises an event if the in-flight transaction count exceeds this threshold. The default is 1000.
Raise event when committed transactions rate exceeds threshold?	Set to `Yes` to raise an event when the number of committed transactions per minute exceeds the threshold you set. The default is yes.
Event severity	Specify the event severity, from 1 to 40, to indicate the importance of an event in which the number of committed transaction per minute exceeds the threshold you set. The default is 5.
Threshold - Committed transactions rate	Enter the threshold value, from 0 to 2,147,483,647, for the number of committed transactions per minute. AppManager raises an event if the number of committed transaction per minutes exceeds this threshold. The default is 100 transactions per minute.
Raise event when aborted transactions rate exceeds threshold?	Set to `Yes` to raise an event when the number of transactions aborted per minute exceeds the threshold you set. The default is yes.

Parameter	How to Set It
Event severity	Specify the event severity, from 1 to 40, to indicate the importance of an event in which the number of transactions aborted per minute exceeds the threshold you set. The default is 5.
Threshold - Aborted transaction rate	Enter the threshold value, from 0 to 2,147,483,647, for the number of transactions aborted per minute. AppManager raises an event if the number of transactions aborted per minute exceeds this threshold. The default is 30.
Raise event when rollback transaction rate, due to application request, exceeds threshold?	Set to `Yes` to raise an event when the rollback transaction rate per minute, due to application requests, exceeds the threshold you set. The default is yes.
Event severity	Specify the event severity, from 1 to 40, to indicate the importance of an event in which the number of application requested rolled back transactions per minute exceeds the threshold you set. The default is 5.
Threshold - Application requested rollback transaction rate	Enter the threshold value, from 0 to 2,147,483,647, for the number of application requested rolled back transactions per minute. AppManager raises an event if the number of application requested rolled back transactions per minute exceeds this threshold. The default is 10.
Raise event when rollback transaction rate, due to resource failure, exceeds threshold?	Set to `Yes` to raise an event when the number of transactions rolled back per minute due to resource failure exceeds the threshold you set. The default is yes.
Event severity	Specify the event severity, from 1 to 40, to indicate the importance of an event in which the number of transactions rolled back per minute due to resource failure exceeds the threshold you set. The default is 5.
Threshold - Rollback transaction rate due to resource failure	Enter the threshold value, from 0 to 2,147,483,647, for the number of transactions rolled back per minute due to resource failure. AppManager raises an event if the number of transactions rolled back per minute due to resource failure exceeds this threshold. The default is 10.
Data collection options	
Collect data for transaction creation rate?	Set to `Yes` to collect data for the transactions created per minute rate. The default is no.
Collect data for In-flight transaction count?	Set to `Yes` to collect data for the in-flight transaction count. The default is no.
Collect data for committed transaction rate?	Set to `Yes` to collect data for the committed transactions per minute rate. The default is no.
Collect data for aborted transaction rate?	Set to `Yes` to collect data for the aborted transactions per minute rate. The default is no.
Collect data for transactions rolled back rate due to application request?	Set to `Yes` to collect data for the per minute rate of transactions rolled back due to application request. The default is no.
Collect data for transactions rolled back rate due to resource failure?	Set to `Yes` to collect data for the per minute rate of transactions rolled back due to resource failure. The default is no.

3.7 JettyQueuedThreadPool

Use this Knowledge Script to monitor the queued thread pool for Jetty servers. This script can raise an event when the thread count exceeds a threshold you set.

NOTE: This Knowledge Script requires that Jetty-specific managed beans be enabled through Java Management Extensions (JMX).

3.7.1 Resource Objects

Java virtual machine object

3.7.2 Default Schedule

The default interval for this script is **Every 5 minutes**.

3.7.3 Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event settings	
Event severity when job fails	Specify the event severity, from 1 to 40, to indicate the importance of an event in which this job fails. The default is 5.
Raise event when AppManager fails to get metrics?	Set to `Yes` to raise an event if AppManager cannot collect any information about Jetty queued thread pool usage. The default is yes.
Event severity	Specify the event severity, from 1 to 40, to indicate the importance of an event in which AppManager cannot collect any information about Jetty queued thread pool usage. The default is 5.
Raise event when current thread count exceeds threshold?	Set to `Yes` to raise an event if thread count exceeds the threshold you set. The default is yes.
Event severity	Specify the event severity, from 1 to 40, to indicate the importance of an event in which thread utilization exceeds the threshold you set. The default is 5.
Threshold - current thread count to maximum threads	Enter the threshold value, from 0 to 100, for current thread count as a percentage of the maximum thread count. AppManager raises an event when the thread utilization exceeds this threshold. The default is 80%.
Filter settings	
Filter to exclude thread pool id(s)	Type the thread pool identifiers, as regular expressions, that you want to exclude from monitoring. The default is a null string. **NOTE:** The Knowledge Script interprets the text in this field as a regular expression. If you type a comma-separated list of explicit identifiers, they are interpreted as a regular expression and the identifiers are not excluded. You must enter all identifiers in regular expression form.
Data collection options	

Parameter	How to Set It
Collect data for current thread count?	Set to `Yes` to collect data for the current thread count. The default is no.
Collect data for maximum thread count?	Set to `Yes` to collect data for the maximum thread count. The default is no.
Collect data for idle thread count?	Set to `Yes` to collect data for idle thread count. the default is no.

3.8 ManagedBeanUtil

Use this Knowledge Script to query a managed bean server for a specific Java managed bean counter. This script can raise events when the Java bean query:

- Returns or does not return specific strings you identified
- Returns numeric data that falls outside a minimum and maximum threshold you set

This Knowledge Script can also raise an unconditional event with the text string value of the managed bean counter.

3.8.1 Resource Objects

JAVA managed bean counter object

3.8.2 Default Schedule

The default interval for this script is **Every 5 minutes**.

3.8.3 Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event settings	
Event severity when job fails	Specify the event severity, from 1 to 40, to indicate the importance of an event in which this job fails. The default is 5.
Raise event when AppManager fails to get metrics?	Set to `Yes` to raise an event if AppManager cannot collect Java bean query output. The default is yes.
Event severity	Specify the event severity, from 1 to 40, to indicate the importance of an event in which AppManager cannot collect Java bean query output. The default is 5.
Raise event with output result?	Set to `Yes` to raise an event containing the result of the Java bean query. The default is no.
Event severity	Specify the event severity, from 1 to 40, to indicate the importance of an event that contains the result of the Java bean query. The default is 25.

Parameter	How to Set It
Raise event if output contains specific strings?	Set to `Yes` to raise an event when the Java bean query output contains one or more of the strings you specified. The default is no.
String list (comma-separated)	Enter one or more comma-separated strings to search for in the Java bean query output. The Knowledge Script raises an event if one or more of these strings is found. The default is a null string.
Event severity	Specify the event severity, from 1 to 40, to indicate the importance of an event in which one or more of the strings you specified is found in the Java bean query output. The default is 5.
Raise event if output doesn't contain specific strings?	Set to `Yes` to raise an event when the Java bean query does not contain one or more of the strings you specified. The default is no.
String list (comma-separated)	Enter one or more comma-separated strings to search for in the Java bean query output. The Knowledge Script raises an event if one or more of these strings is not found.The default is a null string.
Event severity	Specify the event severity, from 1 to 40, to indicate the importance of an event in which one or more of the strings you specified is not found in the Java bean query output. The default is 5.
Raise event if extracted numeric data exceed thresholds?	Set to Yes to raise an event when numeric data extracted from the Java bean query output falls outside the minimum and maximum thresholds you set. The default is no.
Event severity	Specify the event severity, from 1 to 40, to indicate the importance of an event in which numeric data extracted from the Java bean query output falls outside the minimum and maximum thresholds you set. The default is 5.
Threshold - maximum:minimum (comma-separated if composite attribute)	Enter maximum and minimum thresholds separated by a colon to define thresholds for numeric data extracted from the Java bean query output. Separate multiple maximum and minimum thresholds with commas. For example, the string `17:7,90:10,50:5` defines three maximum and minimum thresholds: maximum17 and minimum 7, maximum 90 and minimum 10, and maximum 50 and minimum 5. The default is a null string. This Knowledge Script raises an event when either threshold is crossed. For example, if you specify threshold `17:7`, this script raises an event when the extracted numeric data is greater than 17 or less than 7. Thresholds do not require both maximum and minimum values. For example, `30` or `30:` define a maximum of 30 and no minimum, while `:20` defines a minimum of 20 and no maximum. In the first case, this script only raises an event if the numeric data is greater than 30. In the second case, this script only raises an event when the numeric data is less than 20.
Managed Bean Settings	

Parameter	How to Set It
Inner members of composite data types (comma-separated list)	Enter one or more comma-separated managed bean inner attribute members for composite data types you want to monitor. The default is a null string. This parameter is only for composite attributes. If dropped on a simple attribute, leave this parameter as a null string. This parameter does not require the **Threshold - maximum:minimum** parameter. If you want to monitor inner attribute members but not raise events, you can specify the inner attribute members here and leave **Threshold - maximum:minimum** as a null string. But if you want to raise events based on thresholds for inner attribute members, complete this parameter along with the **Threshold - maximum:minimum** parameter to identify the inner attribute members you want to monitor and their event thresholds. For example, if you type `used,init` for this parameter and `20:17,80:18` for the **Threshold - maximum:minimum** parameter, this script maps the following: ◆ thresholds `20:17` to inner member `used` ◆ thresholds `80:18` to inner member `init` This script raises an event for either of the following conditions: ◆ inner member `used` value is greater than 20 or less than 17 ◆ inner member `init` value is greater than 80 or less that 18
Data Collection	
Collect numeric data?	Set to Yes to collect numeric data from the Java bean query. The default is no.

3.9 MemoryPools

Use this Knowledge Script to monitor Java Virtual Machine (JVM) memory pools. This script can raise an event when memory pool utilization exceeds the threshold you set.

3.9.1 Resource Objects

Java virtual machine object

3.9.2 Default Schedule

The default interval for this script is **Every 5 minutes**.

3.9.3 Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event settings	

Parameter	How to Set It
Event severity when job fails	Specify the event severity, from 1 to 40, to indicate the importance of an event in which this job fails. The default is 5.
Raise event when AppManager fails to get metrics?	Set to `Yes` to raise an event if AppManager cannot collect any information about JVM memory pool usage. The default is yes.
Event severity	Specify the event severity, from 1 to 40, to indicate the importance of an event in which AppManager cannot collect any information about JVM memory pool usage. The default is 5.
Raise event when JVM memory pools utilization exceeds threshold?	Set to `Yes` to raise an event when JVM memory pools utilization with respect to its maximum utilization value exceeds the threshold you set. The default is yes.
Event severity	Specify the event severity, from 1 to 40, to indicate the importance of an event in which JVM memory pools utilization exceeds the threshold you set. The default is 5.
Threshold in percentage	Enter the threshold value, from 1 to 100, for JVM memory pools utilization. AppManager raises an event if the JVM memory pools utilization exceeds this threshold. The default is 80%.
Filter settings	
Filter to exclude memory pools	Type the name of a memory pool to exclude from data collection. You can use regular expressions to specify the memory pool names.The default is `Perm Gen(.*)|Eden Space|Code Cache|Survivor Space`.
Data collection options	
Collect data for JVM memory pools utilization?	Set to `Yes` to collect data for JVM memory pools utilization. The default is no.
Collect data for JVM memory pools utilization in percent?	Set to `Yes` to collect JVM memory pools utilization in percent. The default is no.
Collect data for JVM memory pools maximum value?	Set to `Yes` to collect data for JVM memory pools maximum utilization value. The default is no.

3.10 ThreadUtil

Use this Knowledge Script to monitor Java thread utilization. This script can collect data for process thread count, daemon thread count, and peak thread count.

3.10.1 Resource Objects

Java virtual machine object

3.10.2 Default Schedule

The default interval for this script is **Every 5 minutes**.

3.10.3 Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event settings	
Event severity when job fails	Specify the event severity, from 1 to 40, to indicate the importance of an event in which this job fails. The default is 5.
Raise event when AppManager fails to get metrics?	Set to `Yes` to raise an event if AppManager cannot collect any information about the Java thread utilization. The default is yes.
Event severity	Specify the event severity, from 1 to 40, to indicate the importance of an event in which AppManager cannot collect any information about Java thread utilization. The default is 5.
Data collection options	
Collect data for JVM thread count?	Set to `Yes` to collect data for the JVM process thread count. The default is yes.
Collect data for JVM daemon thread count?	Set to `Yes` to collect data for the JVM process daemon thread count. The default is yes.
Collect data for JVM peak thread count?	Set to `Yes` to collect data for the JVM process peak thread count. The default is no.

3.11 TomcatAppSessions

Use this Knowledge Script to monitor application session statistics for Apache Tomcat servers. This script can raise an event when the rate of session creation or the rate of session expiration exceeds the thresholds that you set.

3.11.1 Resource Objects

Java virtual machine object

3.11.2 Default Schedule

The default interval for this script is **Every 5 minutes**.

3.11.3 Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event settings	
Event severity when job fails	Specify the event severity, from 1 to 40, to idicate the importance of an event in which this job fails. The default is 5.

Parameter	How to Set It
Raise event when AppManager fails to get metrics?	Set to `Yes` to raise an event if AppManager cannot collect any information about Tomcat application session statistics. The default is yes.
Event severity	Specify the event severity, from 1 to 40, to indicate the importance of an event in which AppManager cannot collect any information about Tomcat application statistics. The default is 5.
Raise event when Session create rate exceeds threshold?	Set to `Yes` to raise an event if the number of sessions created per minute exceeds the threshold you set. The default is yes.
Event severity	Specify the event severity, from 1 to 40, to indicate the importance of an event in which the number of sessions created per minute exceeds the threshold you set. The default is 5.
Threshold	Enter the threshold value, from 0 to 2,147,483,647, for the number of sessions created per minute. AppManager raises an event if the number of sessions created per minute exceeds this threshold. The default is 100.
Raise event when Session expire rate exceeds threshold?	Set to `Yes` to raise an event if the number of sessions expiring per minute exceeds the threshold you set. The default is yes.
Event severity	Specify the event severity, from 1 to 40, to indicate the importance of an event in which the number of sessions expiring per minute exceeds the threshold you set. The default is 5.
Threshold	Enter the threshold value, from 0 to 2,147,483,647, for the number of sessions expiring per minute. AppManager raises an event if the number of sessions expiring per minute exceeds this threshold. The default is 100.
Filter settings	
Exclude filter for Application Contexts	Enter an application context to be excluded from Tomcat application session monitoring. You can use regular expressions to specify the application contexts. The default is a null string.
Data collection options	
Collect data for Active Sessions?	Set to `Yes` to collect data for active Tomcat sessions. The default is no.
Collect data for Expired Sessions?	Set to `Yes` to collect data for expired Tomcat sessions. The default is no.

3.12 TomcatThreadPoolStats

Use this Knowledge Script to monitor thread pool statistics for Apache Tomcat servers. This script can raise an event when:

- The current thread count exceeds a percentage of the maximum thread count you set
- The busy thread count exceeds a percentage of the current thread count you set

3.12.1 Resource Objects

Java virtual machine object

3.12.2 Default Schedule

The default interval for this script is **Every 5 minutes**.

3.12.3 Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event settings	
Event severity when job fails	Specify the event severity, from 1 to 40, to indicate the importance of an event in which this job fails. The default is 5.
Raise event when AppManager fails to get metrics?	Set to `Yes` to raise an event if AppManager cannot collect any information about Apache Tomcat thread pool utilization. The default is yes.
Event severity	Specify the event severity, from 1 to 40, to indicate the importance of an event in which AppManager cannot collect any information about Apache Tomcat thread pool utilization. The default is 5.
Raise event when current thread count exceeds threshold?	Set to `Yes` to raise an event if the current thread count exceeds the threshold you set. The default is yes.
Event severity	Specify the event severity, from 1 to 40, to indicate the importance of an event in which the current thread count exceeds the threshold you set. The default is 5.
Threshold - Current thread count to maximum threads	Enter the threshold value, from 0 to 100, for the current thread count as a percentage of the maximum thread count. AppManager raises an event when the thread count exceeds this threshold. The default is 80%.
Raise event when current thread busy count exceeds threshold?	Set to `Yes` to raise an event if the busy thread count exceeds a threshold you set. The default is yes.
Event severity	Specify the event severity, from 1 to 40, to indicate the importance of an event in which the busy thread count exceeds a threshold you set. The default is 5.
Threshold - Current thread busy count to current thread count	Enter the threshold value, from 0 to 100, for the busy thread count as a percentage of the current thread count. AppManager raises an event when the busy thread count exceeds this threshold. The default is 80%.
Filter settings	
Exclude filter for Connectors	Enter a regular expression string to match and exclude connectors from the thread pool statistics. The default is a null string.
Data collection options	
Collect data for current thread count?	Set to `Yes` to collect data for the current thread count. The default is no.
Collect data for maximum thread count?	Set to `Yes` to collect data for the maximum thread count. The default is no.
Collect data for current thread busy count?	Set to `Yes` to collect data for the current thread busy count. The default is no.

Parameter	How to Set It
Collect data for current thread busy count in percent of current thread count?	Set to `Yes` to collect data for the current thread busy count as a percentage of the current thread count. The default is no.