Argent for Java

Note: ArgSoft Intellectual Property Holdings Limited has created this document for informational purposes only. ArgSoft Intellectual Property Holdings Limited makes no warranties, express or implied, in this document. The information contained in this document is subject to change without notice. ArgSoft Intellectual Property Holdings Limited shall not be liable for any technical or editorial errors, or omissions contained in this document, nor for incidental, indirect or consequential damages resulting from the furnishing, performance, or use of the material contained in this document, or the document itself. All views expressed are opinions of ArgSoft Intellectual Property Holdings Limited. All trademarks are the property of their respective owners.

Table of Contents

Introduction

Designed and architected by technology visionary James Gosling in the mid-1990s, Java is a very popular programming language that has been adopted by many software developers. Programs written in Java are used today on millions of devices worldwide.

Java is the buzzword today everywhere.

Google’s Android phones are a key factor and, while Android dominates the smartphone industry, Java is increasingly used in home appliances like televisions, refrigerators, dishwashers, and security systems. Java is an important technology driving the explosive growth of the Internet, especially smart devices and the ‘Internet of Things’. Keep counting the number of computers and devices that use Java as the list is growing day by day.

Java’s rapid adoption and growth cannot be attributed only to Android. Java can be easily ported to a wide variety of operating systems and platforms. Java can be used in Web applications, mobile apps, and complex user interface (UI) programs. Java can be used to integrate divergent legacy applications in extract-transfer-load (ETL) scenarios.

With so many Java programs all around us, it is important to manage and monitor existing and emerging Java-based applications.

What is Argent for Java?

Argent for Java is a comprehensive tracking, monitoring, and management solution focused on Java-based applications across your enterprise.

Argent for Java provides a rule-based monitoring engine that collects statistics and metrics about your Java applications and reports and generates alerts for exception conditions.

Why Argent for Java?

Although Java has been ported and migrated to many operating systems and platforms, the language and run time performance is rarely optimized for each particular platform. Independent Software Providers (ISPs) and IT departments are often most concerned about delivering application features and functionality as a priority ahead of performance and reliability.

The performance and reliability of Java-based applications vary widely and are not consistent. A Java application developed for a UNIX platform may perform poorly on a Windows platform (and vice versa).

Tracking, monitoring, and managing your Java-based applications' performance and resource utilization is critically important, especially in non-stop environments such as customer-facing website applications.

Monitoring CPU and memory utilization can provide useful information but does not help you relate that information to your Java applications.

Argent for Java makes monitoring and managing your Java applications both easy and efficient. With Argent for Java you can forget about memory leaks, deadlocked and blocked threads, CPU overload, and more.

Argent for Java helps ensure your Java-based applications perform smoothly and reliably across your entire enterprise.

Argent for Java screens both JVMs and base servers via Java Management Extensions (JMX) technology.

The Java Virtual Machine (JVM)

Java incorporates a design architecture called the Java Virtual Machine (JVM). The JVM is a hypervisor that executes the Java programs written for it. An implementation of the JVM is part of making Java available on a given platform and operating system. Without an implementation of the JVM, Java programs cannot run. Java programs running within the JVM provide platform independence. This is because JVM converts the Java program's commands to the native language of the machine or device where the JVM and the Java program are installed.

When a Java application is launched, a JVM is instantiated for that application and persists in memory until the application completes. Internally, within a given operating system, there will be as many JVM's as there are Java applications launched.

Under Microsoft Windows, this is comparable to running a console-style application within a DOS command prompt window: There will be as many DOS command prompt windows as there are console-style command line programs active.

Since a JVM is a run-time environment for a Java application, each JVM will consume computer resources such as memory, CPU, and so on.

Argent for Java provides collects vital statistics for the JVM (and, correspondingly, the companion Java application).

Metrics and statistics collected include: Memory and CPU usage, CPU time consumed, thread and handle usage, thread counts, locks / deadlocks, and many more.

Argent for Java also keeps track of similar metrics for host machine where the JVM has been instantiated.

Memory Management

Within an instantiated JVM, memory must be allocated for new objects. Each JVM divides memory allocation into two categories: Heap Memory and Non-Heap Memory.

Heap Memory

Java heap memory from the Operating System is allocated by the JVM and it manages the heap for its Java Application. Every time the Java application creates a new object, the Java Virtual Machine gives out an adjacent space or an array of heap memory to store it within the heap memory already it took from the OS. “Live” objects that are frequently referenced by other objects are retained in the heap and those not referenced anymore are emptied from the heap or Garbage Collected by the JVM. This frees the heap memory.

The most newly created objects are referred to as “Young” generation by JVM algorithms and they become “Old” generation after they endure a few garbage collection processes. The young generation holds on to a small but extremely active segment of the heap where new objects are allocated. When the space allocated for Young generation gets full, a special garbage collection called ‘young collection’ frees up some of the young heaps by moving or promoting the oldest of the “Young” heaps to the “Old” heap. This frees up some space in Young heap which lets the JVM to allocate new objects again. ‘Old collection’ frees up space in the old heap by running a garbage collection in the old heap.

Young generation heap is again split into **Eden Space** and **Survivor Space**.

Eden Space is where new objects are actually allocated in the young heap. Many of these newly created objects will be dereferenced soon after they are created and become inaccessible. Objects that are not dereferenced are passed on to survivor space by the garbage collector first. In exceptional cases they get copied directly into the old generation heap.

Survivor Space is where the young generation objects that are not dereferenced are moved into by the garbage collector. In the survivor space, the surviving objects are shifted within the space to survive a few more GC passes after which only they move on to the “Old generation” heap. This is for optimal utilization of heap memory.

Old/Tenured space is the “Old generation” heap which is the largest memory pool to keep the objects that need to live for longer periods. Objects that leave survivor spaces are copied into tenured space.

Non-Heap Memory

Java Virtual Machine’s non-heap memory stores the runtime constant pool, field and function data and the program for functions and constructors for each class structure. It is a work area which is shared by all threads and the memory used for optimization of JVM’s internal processing.

Non-Heap memory is further split into **Permanent Generation (PERM)** and **Code Cache**.

Permanent Space is the pool that contains metadata of the virtual machine as such, like the class, method objects etc.

Code Cache contains the memory used for compilation and storage of native code by JVM.

The subsequent heap memory parameters are screened by Argent for Java:

Eden Space Usage: Keeps track of the space used (%, KB, MB and GB) by Eden space

Survivor Space Usage: Keeps track of the space used (%, KB, MB and GB) by Survivor space

Old/Tenured Space Usage: Keeps track of the space used (%, KB, MB and GB) by Old/Tenured space

Overall Heap Memory Usage: Keeps track of the space used (%, KB, MB and GB) by overall heap memory

The subsequent non-heap memory parameters are screened by Argent for Java:

Permanent Space Usage: Keeps track of the space consumed (%, KB, MB and GB) by Permanent space

Code Cache Usage: Keeps track of the space consumed (%, KB, MB and GB) by Code Cache

Garbage Collection

Garbage Collection (or GC) is a process that makes sure that unused memory is freed while an application is running. This process makes memory management and the application more efficient. When an application starts up each object is provided a memory space in the heap so that it can be referenced easily within the program or application. Garbage Collection identifies the objects that are never referenced and clears the space in memory booked for them. This makes space for new objects which are referenced in the heap.

Earlier programming languages like C required manual allocation and de-allocation of memory. Java comes with an in-built Garbage Collector which makes memory allocation more efficient.

Argent for Java tracks and monitors the following aspects of a Java Garbage Collector:

Time Spent For Garbage Collection: Estimated time taken for garbage collection

Number of Collections: Total rounds of garbage collection that have been processed or completed

Thread Statistics

One of the most important features of Java is that the JVM allows synchronized multi-threading with each thread executing its own function while JVM in itself remains a single process. A most commonly used example of such an application is an instant messenger which runs 2 threads – one waits for user's input while the other keeps checking the server for incoming posts. Another instance is a server application processing and executing different requests in different threads whereas certain requests in may have multiple threads running parallel.

Each thread or process utilizes part of the memory, CPU and storage are available to a JVM. Each JVM starts a new thread at the start or main() method of a program. Each process or thread within the program initiates a new path from it and stays independent. These threads can run concurrently on separate processors or in a single processor also. How the threads are prioritized processing on the same processor is controlled by the thread scheduler.

If we keep track of threads in a JVM, we can easily spot out the deadlocks by recognizing the code that takes up more CPU resources.

Argent for Java monitors the following types of JVM threads:

Live Threads: Number of live threads currently running

Daemon Threads: Number of daemon threads currently running

Total Threads Started: Total number of threads created and also started since the Java Virtual Machine started

Peak Threads: Peak live thread count since the Java Virtual Machine started or peak was reset

Thread's CPU Time: Total CPU time consumption of JVM threads

Deadlocked Threads: Number of threads that are in deadlock waiting to acquire object monitors

Class Loader Statistics

Java Runtime Environment or JRE loads the required Java classes automatically into the JVM with the help of the Java Class Loader. It is a part of JRE and makes life easier for Java runtime system which need not bother about the files and file systems.

The following aspects of a Java Class loader are closely tracked by Argent for Java:

Loaded Class Count: Keeps track of the number of classes loaded in JVM at any given time.

Total Loaded Class Count: Keeps track of the total number of classes loaded from the time when JVM started execution.

Unloaded Class Count: Keeps track of the number of classes unloaded from the JVM from the time when JVM started execution.

Java Just-In-Time (JIT) Compilation Statistics

The Java compiler converts Java code into bytecode which is easily understood by the hardware's processor. Bytecode does not depend on an operating system or a platform; rather it is understood by the device that runs the code. Java uses a Just-In-Time or JIT compiler that converts the bytecode into the device's native machine code. Since this compilation is done in runtime it is called a Just-In-Time or JIT compiler. Java's JIT can access dynamic runtime data and optimize in-line functions used repeatedly contrary to a standard compiler that cannot access runtime information.

Argent for Java tracks the time spent in JIT compilation.

CPU Statistics

Argent for Java tracks and monitors the subsequent CPU usage records of JVM:

CPU Time Consumption: Keeps track of the CPU time consumed by the processes on which the JVM is running.

CPU Usage: Keeps track of the "current CPU usage" for the JVM processes

Uptime Statistics

Argent for Java tracks and monitors the Uptime or time since the Java Virtual Machine process initiated

Host Memory Statistics

The memory details of the host machine that runs the JVMs are very important figures. Argent for Java tracks and monitors the subsequent memory information of a host machine.

Physical Memory Usage: Keeps track of the host machine's physical memory utilization

Swap Memory Usage: Keeps track of the host machine's swap space utilization

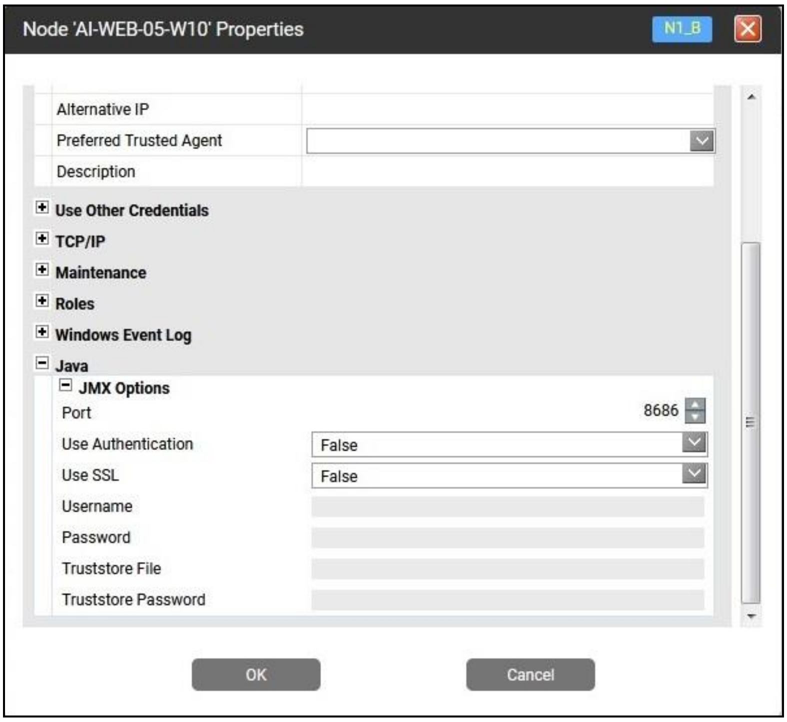
Argent for Java Prerequisites

Argent for Java requires:

1) An Argent server
    a. The Argent server can be 32-bit, if needed
2) A 32-bit version of JDK 1.7 (or above)
3) After installing the JDK, the **JAVA_HOME** environment variable must be configured
    a. Right click on the **Computer** icon and select **Properties** from the context menu
    b. Select **Advanced system settings** on the left portion of the window
    c. Click the **Environment Variables** button
        i. In the **System Variables** section, navigate to the **JAVA_HOME** environment variable, select it, and click **Edit**
        ii. Type the folder location where the JDK software was installed, e.g. **D:\Program Files (x86)\Java\jdk1.8.0_51**

Connectivity

Java Management Extensions (or JMX) technology inherent in the Java Virtual Machine is a mechanism that helps you to keep track of the JVM's efficiency. Argent for Java connects to and screens a remote JVM using JMX technology. The basic details such as machine name (where a JVM is operating), port number, and user details need to be provided. The following sections explain the configuration in more detail.



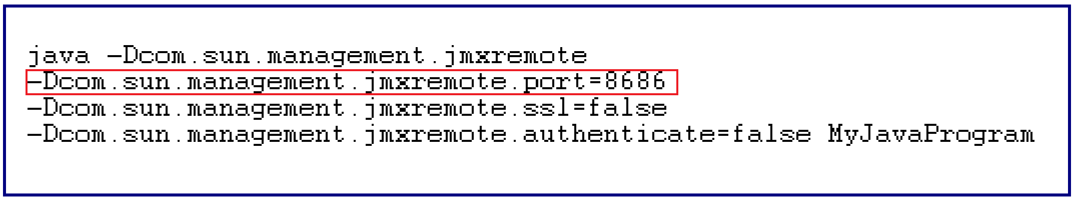
Node 'AI-WEB-05-W10' Properties

Alternative IP	
Preferred Trusted Agent	
Description	

- ⊞ Use Other Credentials
- ⊞ TCP/IP
- ⊞ Maintenance
- ⊞ Roles
- ⊞ Windows Event Log
- ⊟ Java
  - ⊟ JMX Options

Port	8686
Use Authentication	False
Use SSL	False
Username	
Password	
Truststore File	
Truststore Password	

When a Java application starts, several properties must be configured to enable the JMX manager to monitor the Java Virtual Machine. To explain further, the following command-line system properties should be set up to start a Java application named "MyJavaProgram":



```
java -Dcom.sun.management.jmxremote
-Dcom.sun.management.jmxremote.port=8686
-Dcom.sun.management.jmxremote.ssl=false
-Dcom.sun.management.jmxremote.authenticate=false MyJavaProgram
```

The **port** is the port number which enables the JMX connections

Argent for Java can connect to a remote JVM in 4 different ways using JMX.

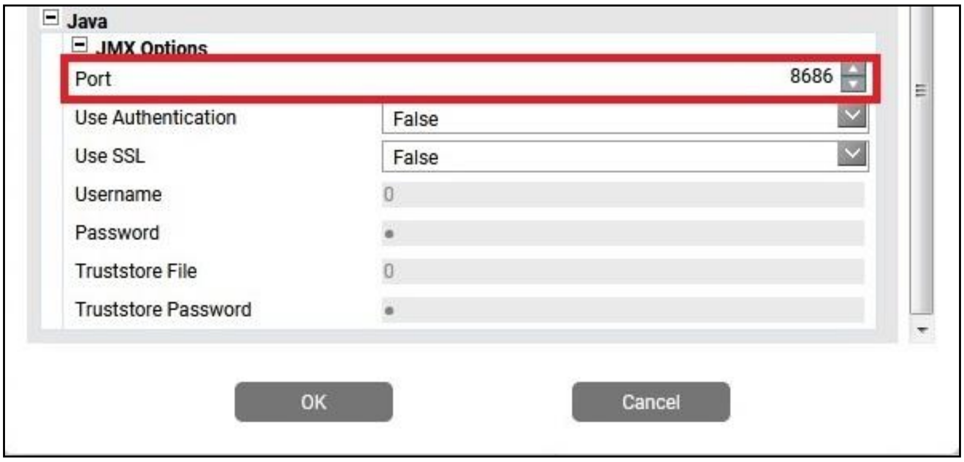
1. Connectivity that requires no user validation or SSL
2. Connectivity that requires user validation
3. Connectivity that requires user and SSL validations
4. Connectivity that requires SSL validation

Connectivity With No User Authentication Or SSL

In this method, no user or SSL verification is required to connect to a remote JVM. It can be set up by configuring the port number and setting all remote user verification properties to false in the command-line as follows:

```
Java -Dcom.sun.management.jmxremote
-Dcom.sun.management.jmxremote.port=8686
-Dcom.sun.management.jmxremote.ssl=false
-Dcom.sun.management.jmxremote.authenticate=false MyJavaProgram
```

Make sure that the same port number is referenced in the Argent for Java node properties dialog:



Java

JMX Options

Port	8686
Use Authentication	False
Use SSL	False
Username	0
Password	•
Truststore File	0
Truststore Password	•

Set the JMX options for **Use Authentication** and **Use SSL** to **False**.

Connectivity with User Authentication

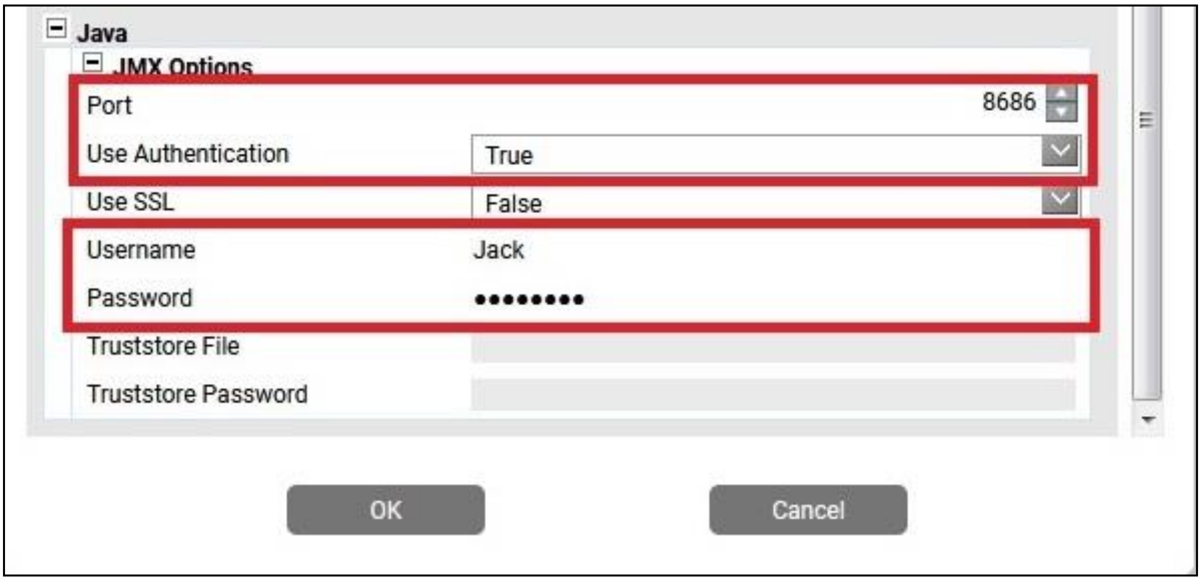
In this method, user verification is required to connect to a remote JVM. The subsequent basic command-line values must be provided when starting the Java application to be monitored:

```
java -Dcom.sun.management.jmxremote
-Dcom.sun.management.jmxremote.port=8686
-Dcom.sun.management.jmxremote.authenticate=true
-Dcom.sun.management.jmxremote.ssl=false
-Dcom.sun.management.jmxremote.access.file=C:\JMX\jmxremote.access
-Dcom.sun.management.jmxremote.password.file=C:\JMX\jmxremote.password  MyJavaProgram
```

A password file stores the username and password to authenticate.

Set the following properties in the Argent for Java node properties dialog:

- Port number of the remote machine to connect (which runs the JVM)
- Set Port Authentication to **true**
- Provide the user credentials such as username and password for verification



Java

JMX Options

Port	8686
Use Authentication	True
Use SSL	False
Username	Jack
Password	••••••••
Truststore File	
Truststore Password	

For this type of connection, set **Use SSL** to **False**.

Connectivity with User and SSL Authentication

In this method, user and SSL authentications are required to connect to a remote JVM. The basic command line values to be set when starting the Java application are as follows:

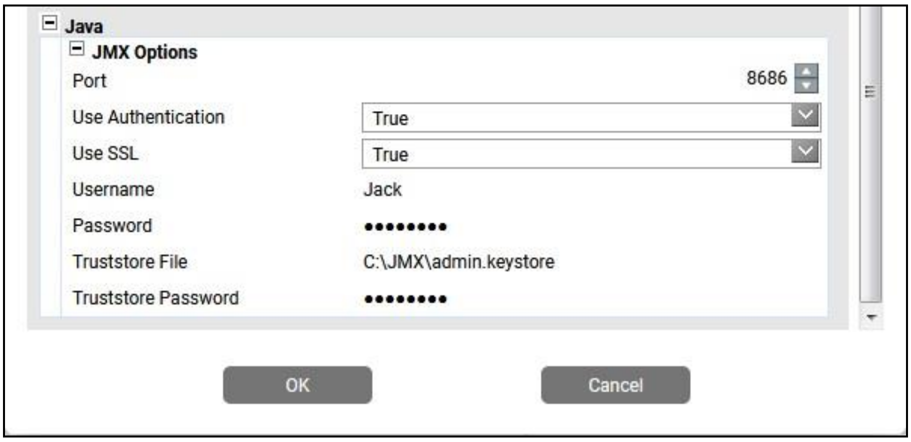
```
java -Dcom.sun.management.jmxremote
-Dcom.sun.management.jmxremote.port=8686
-Dcom.sun.management.jmxremote.authenticate=true
-Dcom.sun.management.jmxremote.ssl=true
-Dcom.sun.management.jmxremote.access.file=C:\JMX\jmxremote.access
-Dcom.sun.management.jmxremote.password.file=C:\JMX\jmxremote.password
-Djavax.net.ssl.keyStore=C:\JMX\admin.keystore
-Djavax.net.ssl.keyStorePassword=Plain text password
-Djava.security.manager MyJavaProgram
```

A password file stores the username and password to authenticate.

Specify the path of certificate keystore file and provide the keystore password for SSL authentication.

Make sure to set the following properties in the Argent for Java node properties dialog:

- Port number of the remote machine to connect (which runs the JVM)
- Set **Use Authentication** to **True**
- Set **Use SSL** to **True**
- Provide the user credentials such as username and password for verification
- Specify the path of SSL certificate keystore file
- Provide the SSL certificate keystore password



Java

JMX Options

Port	8686
Use Authentication	True
Use SSL	True
Username	Jack
Password	••••••••
Truststore File	C:\JMX\admin.keystore
Truststore Password	••••••••

Connectivity with SSL Authentication

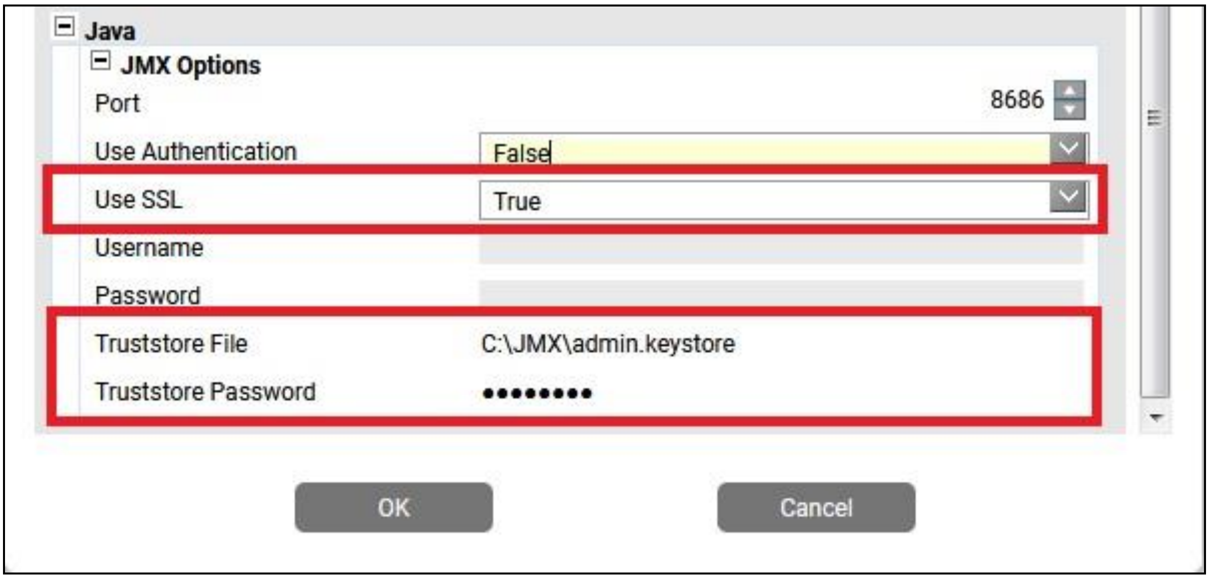
In this method, only SSL authentication is required to connect to a remote JVM. The basic command line values to be set when starting the Java application are as follows:

```
java -Dcom.sun.management.jmxremote
-Dcom.sun.management.jmxremote.port=8686
-Dcom.sun.management.jmxremote.authenticate=false
-Dcom.sun.management.jmxremote.ssl=true
-Djavax.net.ssl.keyStore=C:\JMX\admin.keystore
-Djavax.net.ssl.keyStorePassword=Plain text password
-Djava.security.manager MyJavaProgram
```

Specify the path of certificate keystore file and provide the keystore password for SSL authentication

Make sure you set the following properties in Argent for Java node properties dialog:

- Port number of the remote machine to connect (which runs the JVM)
- Set **Use SSL** to **True**
- Specify the path of SSL certificate keystore file
- Provide the SSL certificate keystore password

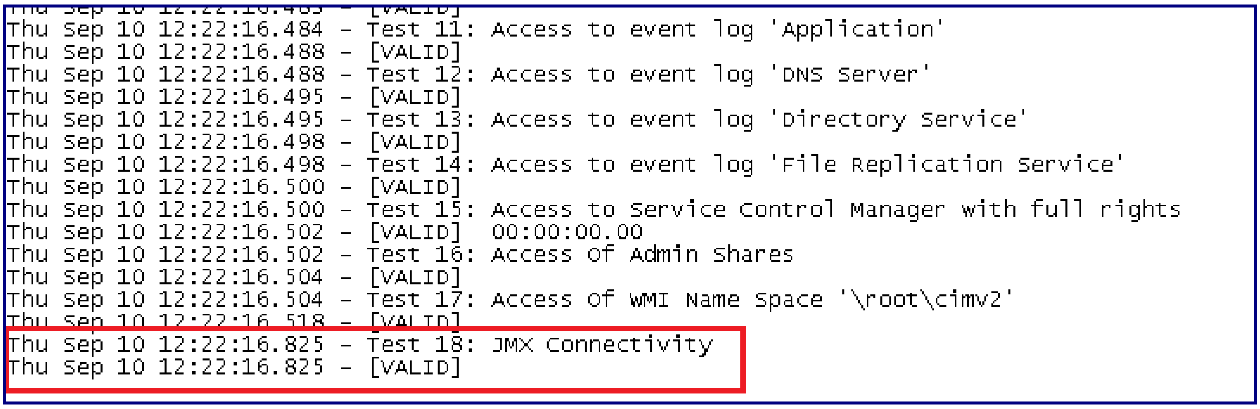


Java

JMX Options

Port	8686
Use Authentication	False
Use SSL	True
Username	
Password	
Truststore File	C:\JMX\admin.keystore
Truststore Password	•••••••••

After the attributes of JMX are set for the connection type you want to use, make sure to check that the JMX connectivity test returns **[VALID]**.



```
Thu Sep 10 12:22:16.483 - [VALID]
Thu Sep 10 12:22:16.484 - Test 11: Access to event log 'Application'
Thu Sep 10 12:22:16.488 - [VALID]
Thu Sep 10 12:22:16.488 - Test 12: Access to event log 'DNS Server'
Thu Sep 10 12:22:16.495 - [VALID]
Thu Sep 10 12:22:16.495 - Test 13: Access to event log 'Directory Service'
Thu Sep 10 12:22:16.498 - [VALID]
Thu Sep 10 12:22:16.498 - Test 14: Access to event log 'File Replication Service'
Thu Sep 10 12:22:16.500 - [VALID]
Thu Sep 10 12:22:16.500 - Test 15: Access to Service Control Manager with full rights
Thu Sep 10 12:22:16.502 - [VALID]  00:00:00.00
Thu Sep 10 12:22:16.502 - Test 16: Access Of Admin Shares
Thu Sep 10 12:22:16.504 - [VALID]
Thu Sep 10 12:22:16.504 - Test 17: Access Of WMI Name Space '\root\cimv2'
Thu Sep 10 12:22:16.518 - [VALID]
Thu Sep 10 12:22:16.825 - Test 18: JMX Connectivity
Thu Sep 10 12:22:16.825 - [VALID]
```

Argent for Java Rules

Argent for Java contains rules that track and monitor different statistics and metrics for a JVM. Threshold values are configured for each rule and values retrieved by Argent for Java during execution are checked against configured rules.

Argent for Java generates alerts whenever the actual values diverge from the configured thresholds.

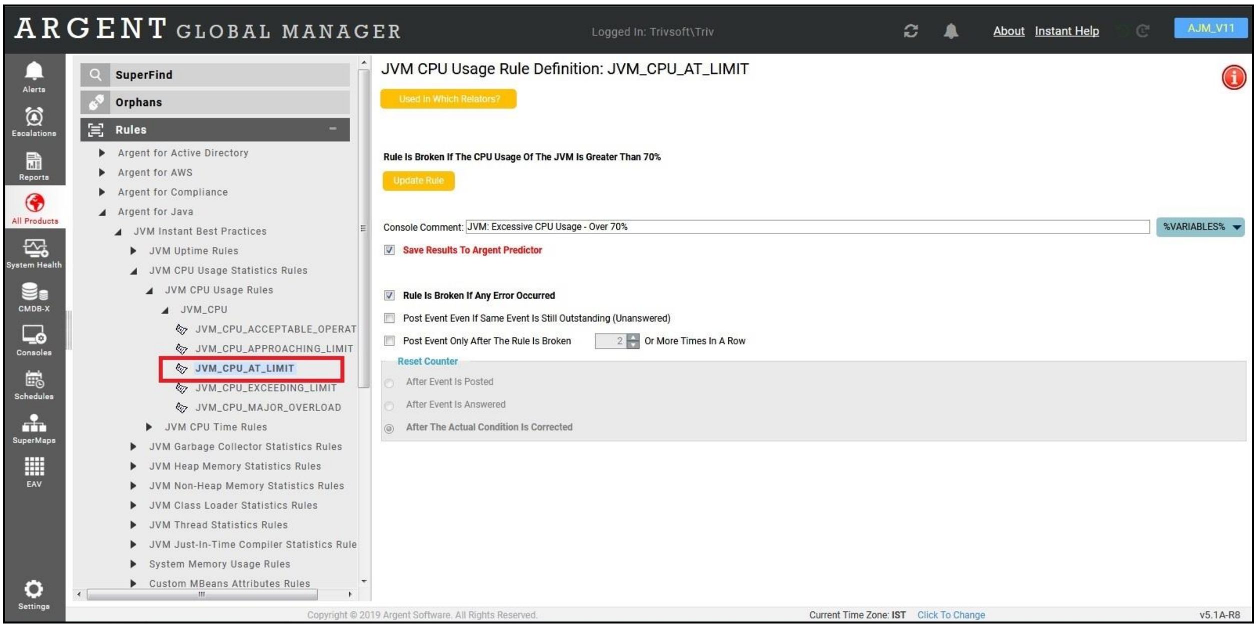
General Rules

Argent for Java provides support for general rules that apply to nearly every executing Java application:

1) CPU usage consumed
2) CPU time consumed
3) Garbage collection count
4) Garbage collection time

CPU Usage

This rule monitors CPU usage consumed by a JVM



ARGENT GLOBAL MANAGER
Logged In: Trivsoft\Triv
About
Instant Help
AJM_V11
Alerts
Escalations
Reports
All Products
System Health
CMDB-X
Consoles
Schedules
SuperMaps
EAV
Settings
SuperFind
Orphans
Rules
Argent for Active Directory
Argent for AWS
Argent for Compliance
Argent for Java
JVM Instant Best Practices
JVM Uptime Rules
JVM CPU Usage Statistics Rules
JVM CPU Usage Rules
JVM_CPU
JVM_CPU_ACCEPTABLE_OPERAT
JVM_CPU_APPROACHING_LIMIT
JVM_CPU_AT_LIMIT
JVM_CPU_EXCEEDING_LIMIT
JVM_CPU_MAJOR_OVERLOAD
JVM CPU Time Rules
JVM Garbage Collector Statistics Rules
JVM Heap Memory Statistics Rules
JVM Non-Heap Memory Statistics Rules
JVM Class Loader Statistics Rules
JVM Thread Statistics Rules
JVM Just-In-Time Compiler Statistics Rule
System Memory Usage Rules
Custom MBeans Attributes Rules
Used In Which Relators?
Update Rule
%VARIABLES%
Copyright © 2019 Argent Software. All Rights Reserved.
Current Time Zone: IST
Click To Change
v5.1A-R8

JVM CPU Usage Rule Definition: JVM_CPU_AT_LIMIT

Rule Is Broken If The CPU Usage Of The JVM Is Greater Than 70%

Console Comment: JVM: Excessive CPU Usage - Over 70%

☑ **Save Results To Argent Predictor**

☑ **Rule Is Broken If Any Error Occurred**

□ Post Event Even If Same Event Is Still Outstanding (Unanswered)

□ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

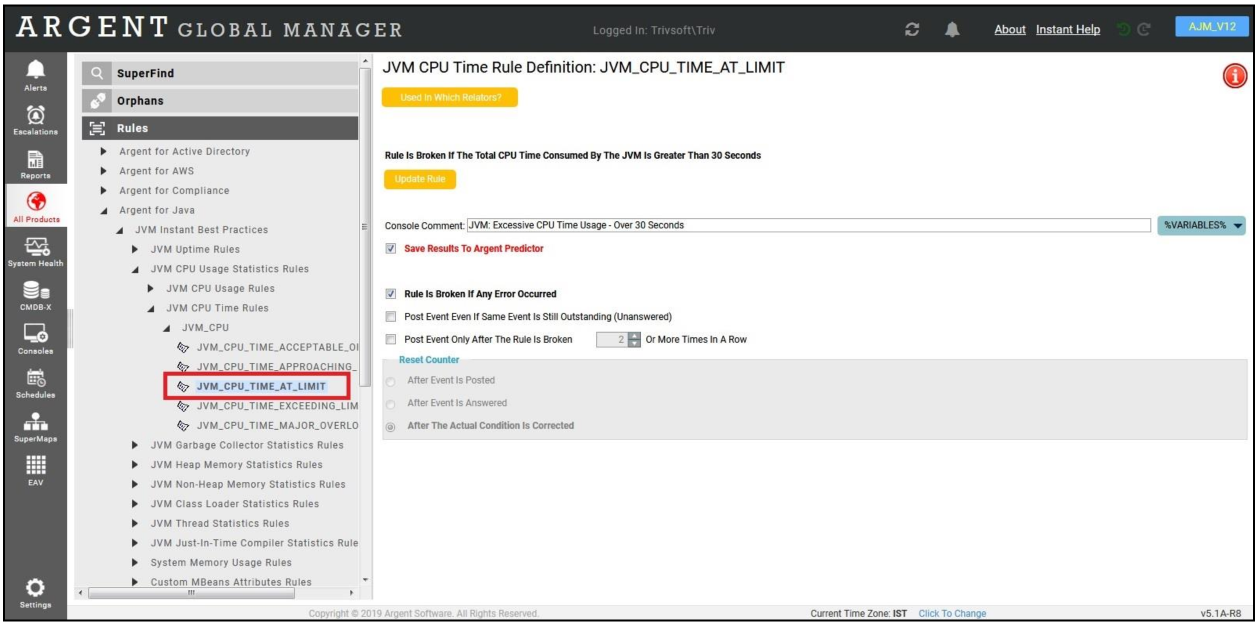
○ After Event Is Posted

○ After Event Is Answered

● After The Actual Condition Is Corrected

CPU Time

This rule monitors CPU time consumed by a JVM


ARGENT GLOBAL MANAGER
Logged In: Trivsoft\Triv
About
Instant Help
AJM_V12
Alerts
Escalations
Reports
All Products
System Health
CMDB-X
Consoles
Schedules
SuperMaps
EAV
Settings
SuperFind
Orphans
Rules
Argent for Active Directory
Argent for AWS
Argent for Compliance
Argent for Java
JVM Instant Best Practices
JVM Uptime Rules
JVM CPU Usage Statistics Rules
JVM CPU Usage Rules
JVM CPU Time Rules
JVM_CPU
JVM_CPU_TIME_ACCEPTABLE_OI
JVM_CPU_TIME_APPROACHING_
JVM_CPU_TIME_AT_LIMIT
JVM_CPU_TIME_EXCEEDING_LIM
JVM_CPU_TIME_MAJOR_OVERLO
JVM Garbage Collector Statistics Rules
JVM Heap Memory Statistics Rules
JVM Non-Heap Memory Statistics Rules
JVM Class Loader Statistics Rules
JVM Thread Statistics Rules
JVM Just-In-Time Compiler Statistics Rule
System Memory Usage Rules
Custom MBeans Attributes Rules
Used In Which Relators?
Update Rule
%VARIABLES%
Copyright © 2019 Argent Software. All Rights Reserved.
Current Time Zone: IST
Click To Change
v5.1A-R8

JVM CPU Time Rule Definition: JVM_CPU_TIME_AT_LIMIT

Rule Is Broken If The Total CPU Time Consumed By The JVM Is Greater Than 30 Seconds

Console Comment: JVM: Excessive CPU Time Usage - Over 30 Seconds

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

☐ Post Event Even If Same Event Is Still Outstanding (Unanswered)

☐ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

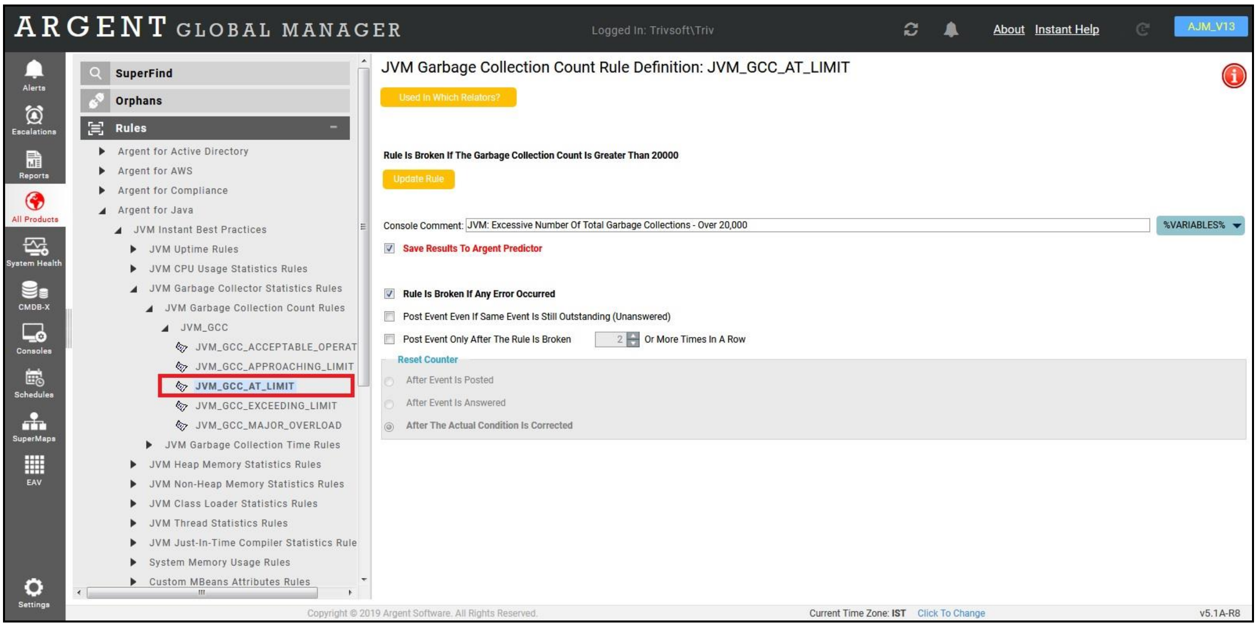
○ After Event Is Posted

○ After Event Is Answered

● After The Actual Condition Is Corrected

Garbage Collection Count

This rule monitors the garbage collection count within a JVM


ARGENT GLOBAL MANAGER
Logged In: Trivsoft\Triv
About
Instant Help
AJM_V13
Alerts
Escalations
Reports
All Products
System Health
CMDB-X
Consoles
Schedules
SuperMaps
EAV
Settings
SuperFind
Orphans
Rules
Argent for Active Directory
Argent for AWS
Argent for Compliance
Argent for Java
JVM Instant Best Practices
JVM Uptime Rules
JVM CPU Usage Statistics Rules
JVM Garbage Collector Statistics Rules
JVM Garbage Collection Count Rules
JVM_GCC
JVM_GCC_ACCEPTABLE_OPERAT
JVM_GCC_APPROACHING_LIMIT
JVM_GCC_AT_LIMIT
JVM_GCC_EXCEEDING_LIMIT
JVM_GCC_MAJOR_OVERLOAD
JVM Garbage Collection Time Rules
JVM Heap Memory Statistics Rules
JVM Non-Heap Memory Statistics Rules
JVM Class Loader Statistics Rules
JVM Thread Statistics Rules
JVM Just-In-Time Compiler Statistics Rule
System Memory Usage Rules
Custom MBeans Attributes Rules
Used In Which Relators?
Update Rule
%VARIABLES%
Copyright © 2019 Argent Software. All Rights Reserved.
Current Time Zone: IST
Click To Change
v5.1A-R8

JVM Garbage Collection Count Rule Definition: JVM_GCC_AT_LIMIT

Rule Is Broken If The Garbage Collection Count Is Greater Than 20000

Console Comment: JVM: Excessive Number Of Total Garbage Collections - Over 20,000

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

□ Post Event Even If Same Event Is Still Outstanding (Unanswered)

□ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

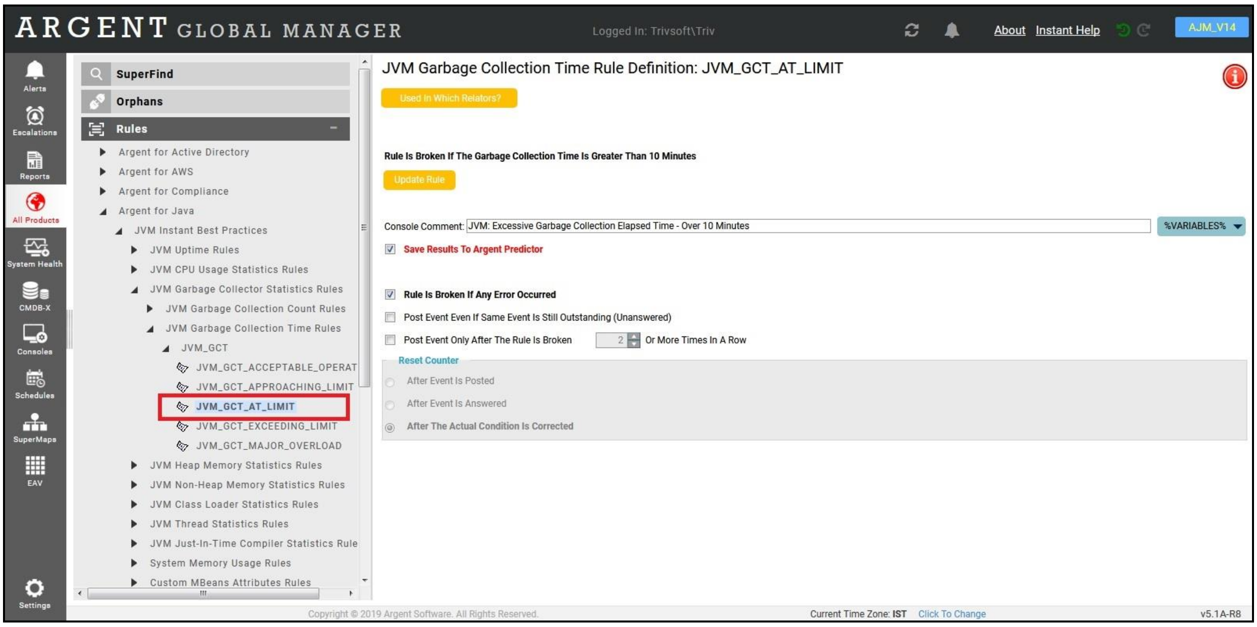
○ After Event Is Posted

○ After Event Is Answered

● After The Actual Condition Is Corrected

Garbage Collection Time

This rule monitors the total time consumed by garbage collection in a JVM


ARGENT GLOBAL MANAGER
Logged In: Trivsoft\Triv
About
Instant Help
AJM_V14
Alerts
Escalations
Reports
All Products
System Health
CMDB-X
Consoles
Schedules
SuperMaps
EAV
Settings
SuperFind
Orphans
Rules
Argent for Active Directory
Argent for AWS
Argent for Compliance
Argent for Java
JVM Instant Best Practices
JVM Uptime Rules
JVM CPU Usage Statistics Rules
JVM Garbage Collector Statistics Rules
JVM Garbage Collection Count Rules
JVM Garbage Collection Time Rules
JVM_GCT
JVM_GCT_ACCEPTABLE_OPERAT
JVM_GCT_APPROACHING_LIMIT
JVM_GCT_AT_LIMIT
JVM_GCT_EXCEEDING_LIMIT
JVM_GCT_MAJOR_OVERLOAD
JVM Heap Memory Statistics Rules
JVM Non-Heap Memory Statistics Rules
JVM Class Loader Statistics Rules
JVM Thread Statistics Rules
JVM Just-In-Time Compiler Statistics Rule
System Memory Usage Rules
Custom MBeans Attributes Rules
Used In Which Relators?
Update Rule
%VARIABLES%
Copyright © 2019 Argent Software. All Rights Reserved.
Current Time Zone: IST
Click To Change
v5.1A-R8

JVM Garbage Collection Time Rule Definition: JVM_GCT_AT_LIMIT

Rule Is Broken If The Garbage Collection Time Is Greater Than 10 Minutes

Console Comment: JVM: Excessive Garbage Collection Elapsed Time - Over 10 Minutes

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

□ Post Event Even If Same Event Is Still Outstanding (Unanswered)

□ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

○ After Event Is Posted

○ After Event Is Answered

● After The Actual Condition Is Corrected

Survivor Space Rules

A JVM allocates heap memory when it starts up. The JVM stores all runtime data in this heap. The JVM assigns memory for all objects and arrays and is typically used by all JVM threads. The heap memory is automatically deallocated by garbage collection when created objects are no longer referenced by a given Java program. Garbage collection automatically recycles the memory allocated for any object in the heap.

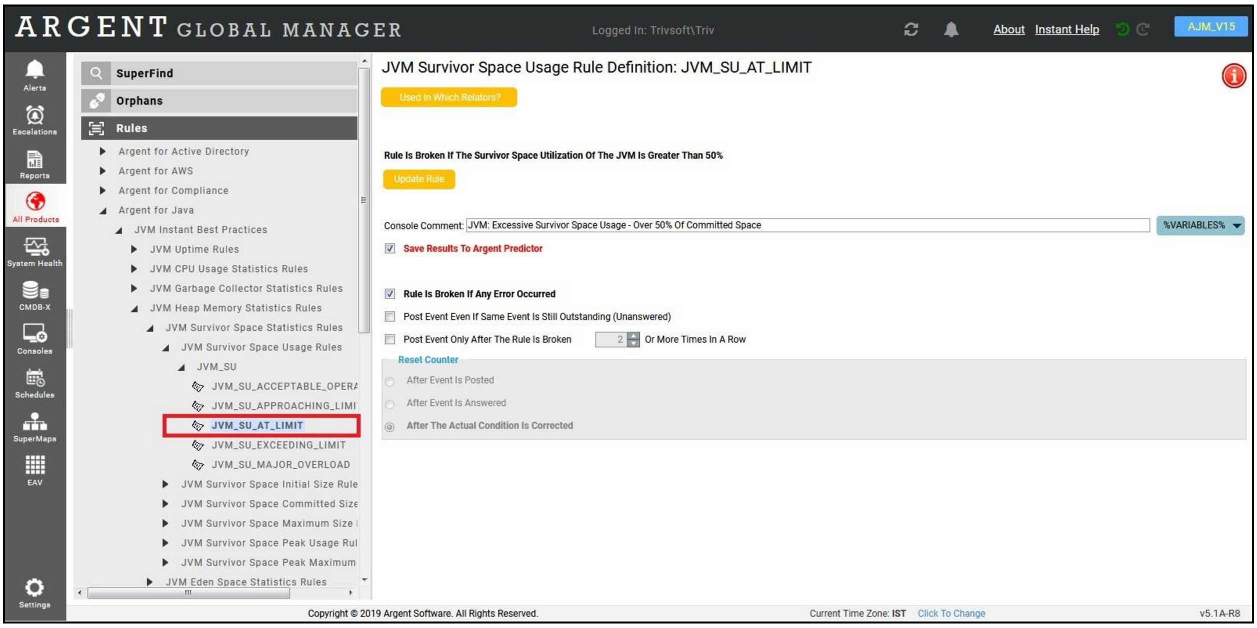
JVM heap memory is further segregated into young and old (or tenured) generations. The young generation space is segregated again into Eden space and Survivor space. All heap memory areas are monitored by Argent for Java.

Argent for Java provides support for the following survivor space rules:

1) Survivor Space Utilization
2) Survivor Space Initial Size
3) Survivor Space Committed Size
4) Survivor Space Maximum Size
5) Survivor Space Peak Usage
6) Survivor Space Peak Maximum Size

Survivor Space Utilization

Survivor space is used to store the surviving objects of a young generation over a few rounds of garbage collection. The total survivor space utilized by a JVM is monitored by Argent for Java.



ARGENT GLOBAL MANAGER
Logged In: Trivsoft\Triv
About
Instant Help
AJM_V15
Alerts
Escalations
Reports
All Products
System Health
CMDB-X
Consoles
Schedules
SuperMaps
EAV
Settings
SuperFind
Orphans
Rules
Argent for Active Directory
Argent for AWS
Argent for Compliance
Argent for Java
JVM Instant Best Practices
JVM Uptime Rules
JVM CPU Usage Statistics Rules
JVM Garbage Collector Statistics Rules
JVM Heap Memory Statistics Rules
JVM Survivor Space Statistics Rules
JVM Survivor Space Usage Rules
JVM_SU
JVM_SU_ACCEPTABLE_OPERA
JVM_SU_APPROACHING_LIMI
JVM_SU_AT_LIMIT
JVM_SU_EXCEEDING_LIMIT
JVM_SU_MAJOR_OVERLOAD
JVM Survivor Space Initial Size Rule
JVM Survivor Space Committed Size
JVM Survivor Space Maximum Size
JVM Survivor Space Peak Usage Rul
JVM Survivor Space Peak Maximum
JVM Eden Space Statistics Rules
Used In Which Relators?
Update Rule
%VARIABLES%

JVM Survivor Space Usage Rule Definition: JVM_SU_AT_LIMIT

Rule Is Broken If The Survivor Space Utilization Of The JVM Is Greater Than 50%

Console Comment: JVM: Excessive Survivor Space Usage - Over 50% Of Committed Space

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

□ Post Event Even If Same Event Is Still Outstanding (Unanswered)

□ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

○ After Event Is Posted

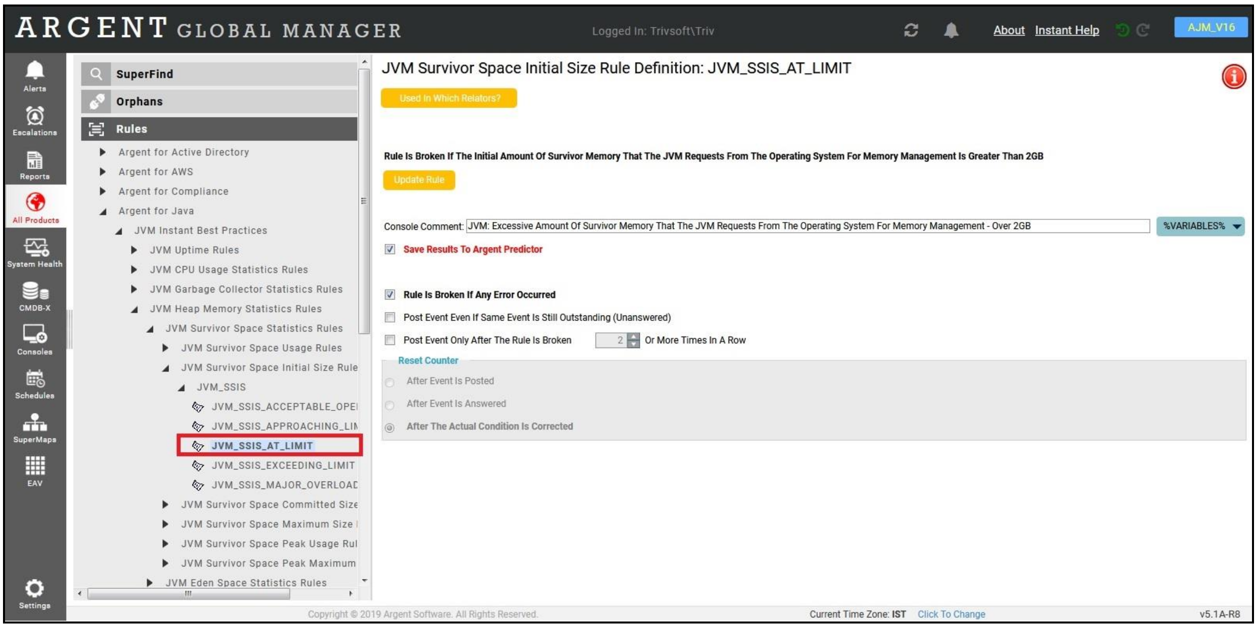
○ After Event Is Answered

● After The Actual Condition Is Corrected

Copyright © 2019 Argent Software. All Rights Reserved.   Current Time Zone: IST Click To Change   v5.1A-R8

Survivor Space Initial Size

This rule monitors the initial survivor memory space requested by a JVM


ARGENT GLOBAL MANAGER
Logged In: Trivsoft\Triv
About
Instant Help
AJM_V16
Alerts
Escalations
Reports
All Products
System Health
CMDB-X
Consoles
Schedules
SuperMaps
EAV
Settings
SuperFind
Orphans
Rules
Argent for Active Directory
Argent for AWS
Argent for Compliance
Argent for Java
JVM Instant Best Practices
JVM Uptime Rules
JVM CPU Usage Statistics Rules
JVM Garbage Collector Statistics Rules
JVM Heap Memory Statistics Rules
JVM Survivor Space Statistics Rules
JVM Survivor Space Usage Rules
JVM Survivor Space Initial Size Rule
JVM_SSIS
JVM_SSIS_ACCEPTABLE_OPEI
JVM_SSIS_APPROACHING_LIN
JVM_SSIS_AT_LIMIT
JVM_SSIS_EXCEEDING_LIMIT
JVM_SSIS_MAJOR_OVERLOAD
JVM Survivor Space Committed Size
JVM Survivor Space Maximum Size I
JVM Survivor Space Peak Usage Rul
JVM Survivor Space Peak Maximum
JVM Eden Space Statistics Rules
Used In Which Relators?
Update Rule
%VARIABLES%

JVM Survivor Space Initial Size Rule Definition: JVM_SSIS_AT_LIMIT

Rule Is Broken If The Initial Amount Of Survivor Memory That The JVM Requests From The Operating System For Memory Management Is Greater Than 2GB

Console Comment: JVM: Excessive Amount Of Survivor Memory That The JVM Requests From The Operating System For Memory Management - Over 2GB

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

□ Post Event Even If Same Event Is Still Outstanding (Unanswered)

□ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

○ After Event Is Posted

○ After Event Is Answered

● After The Actual Condition Is Corrected

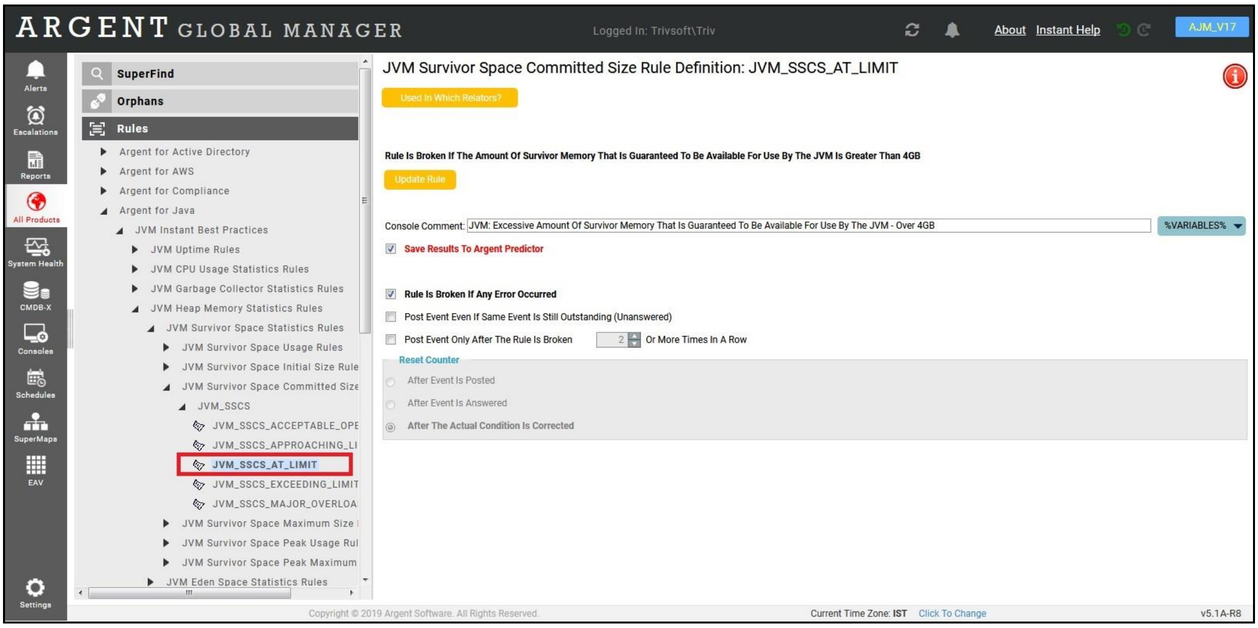
Copyright © 2019 Argent Software. All Rights Reserved.

Current Time Zone: IST Click To Change

v5.1A-R8

Survivor Space Committed Size

This rule monitors the committed size of survivor memory for a JVM



ARGENT GLOBAL MANAGER
Logged In: Trivsoft\Triv
About
Instant Help
AJM_V17
Alerts
Escalations
Reports
All Products
System Health
CMDB-X
Consoles
Schedules
SuperMaps
EAV
Settings
SuperFind
Orphans
Rules
Argent for Active Directory
Argent for AWS
Argent for Compliance
Argent for Java
JVM Instant Best Practices
JVM Uptime Rules
JVM CPU Usage Statistics Rules
JVM Garbage Collector Statistics Rules
JVM Heap Memory Statistics Rules
JVM Survivor Space Statistics Rules
JVM Survivor Space Usage Rules
JVM Survivor Space Initial Size Rule
JVM Survivor Space Committed Size
JVM_SSCS
JVM_SSCS_ACCEPTABLE_OPE
JVM_SSCS_APPROACHING_LI
JVM_SSCS_AT_LIMIT
JVM_SSCS_EXCEEDING_LIMIT
JVM_SSCS_MAJOR_OVERLOA
JVM Survivor Space Maximum Size
JVM Survivor Space Peak Usage Rul
JVM Survivor Space Peak Maximum
JVM Eden Space Statistics Rules
Copyright © 2019 Argent Software. All Rights Reserved.
Current Time Zone: IST
Click To Change
v5.1A-R8

JVM Survivor Space Committed Size Rule Definition: JVM_SSCS_AT_LIMIT

Used In Which Relators?

Rule Is Broken If The Amount Of Survivor Memory That Is Guaranteed To Be Available For Use By The JVM Is Greater Than 4GB

Update Rule

Console Comment: JVM: Excessive Amount Of Survivor Memory That Is Guaranteed To Be Available For Use By The JVM - Over 4GB %VARIABLES%

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

□ Post Event Even If Same Event Is Still Outstanding (Unanswered)

□ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

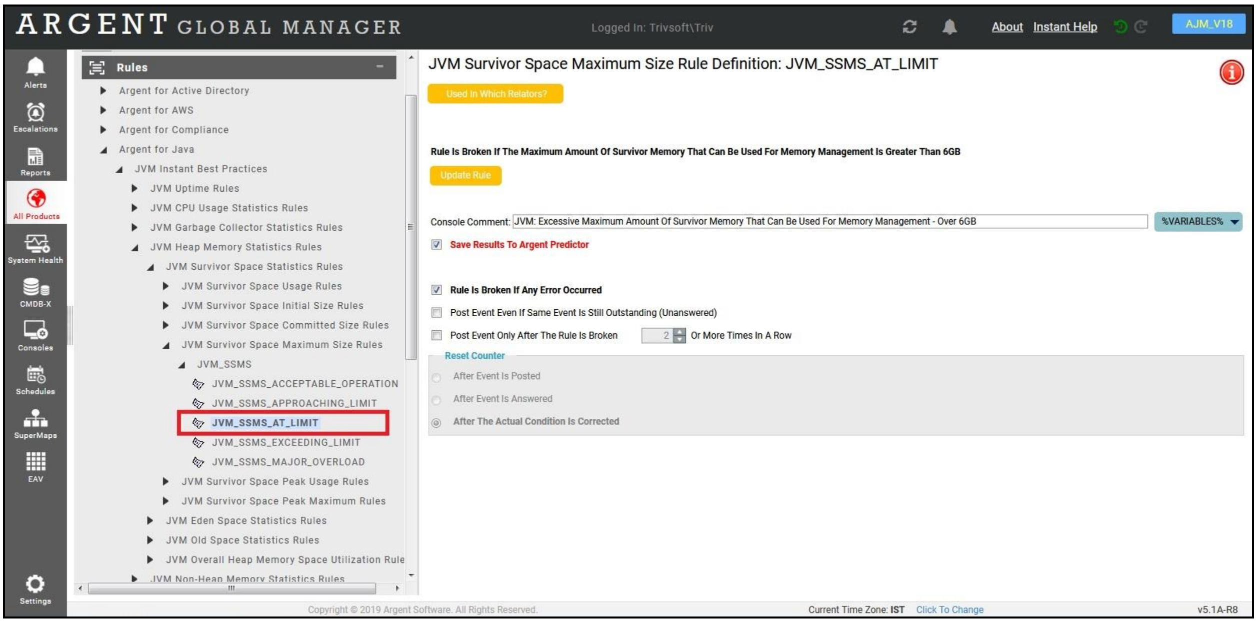
○ After Event Is Posted

○ After Event Is Answered

● After The Actual Condition Is Corrected

Survivor Space Maximum Size

This rule monitors the maximum size of survivor memory in a JVM


ARGENT GLOBAL MANAGER
Logged In: Trivsoft\Triv
About
Instant Help
AJM_V18
Alerts
Escalations
Reports
All Products
System Health
CMDB-X
Consoles
Schedules
SuperMaps
EAV
Settings
Rules
Argent for Active Directory
Argent for AWS
Argent for Compliance
Argent for Java
JVM Instant Best Practices
JVM Uptime Rules
JVM CPU Usage Statistics Rules
JVM Garbage Collector Statistics Rules
JVM Heap Memory Statistics Rules
JVM Survivor Space Statistics Rules
JVM Survivor Space Usage Rules
JVM Survivor Space Initial Size Rules
JVM Survivor Space Committed Size Rules
JVM Survivor Space Maximum Size Rules
JVM_SSMS
JVM_SSMS_ACCEPTABLE_OPERATION
JVM_SSMS_APPROACHING_LIMIT
JVM_SSMS_AT_LIMIT
JVM_SSMS_EXCEEDING_LIMIT
JVM_SSMS_MAJOR_OVERLOAD
JVM Survivor Space Peak Usage Rules
JVM Survivor Space Peak Maximum Rules
JVM Eden Space Statistics Rules
JVM Old Space Statistics Rules
JVM Overall Heap Memory Space Utilization Rule
JVM Non-Heap Memory Statistics Rules
Used In Which Relators?
Update Rule
%VARIABLES%
Copyright © 2019 Argent Software. All Rights Reserved.
Current Time Zone: IST
Click To Change
v5.1A-R8

JVM Survivor Space Maximum Size Rule Definition: JVM_SSMS_AT_LIMIT

Rule Is Broken If The Maximum Amount Of Survivor Memory That Can Be Used For Memory Management Is Greater Than 6GB

Console Comment: JVM: Excessive Maximum Amount Of Survivor Memory That Can Be Used For Memory Management - Over 6GB

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

□ Post Event Even If Same Event Is Still Outstanding (Unanswered)

□ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

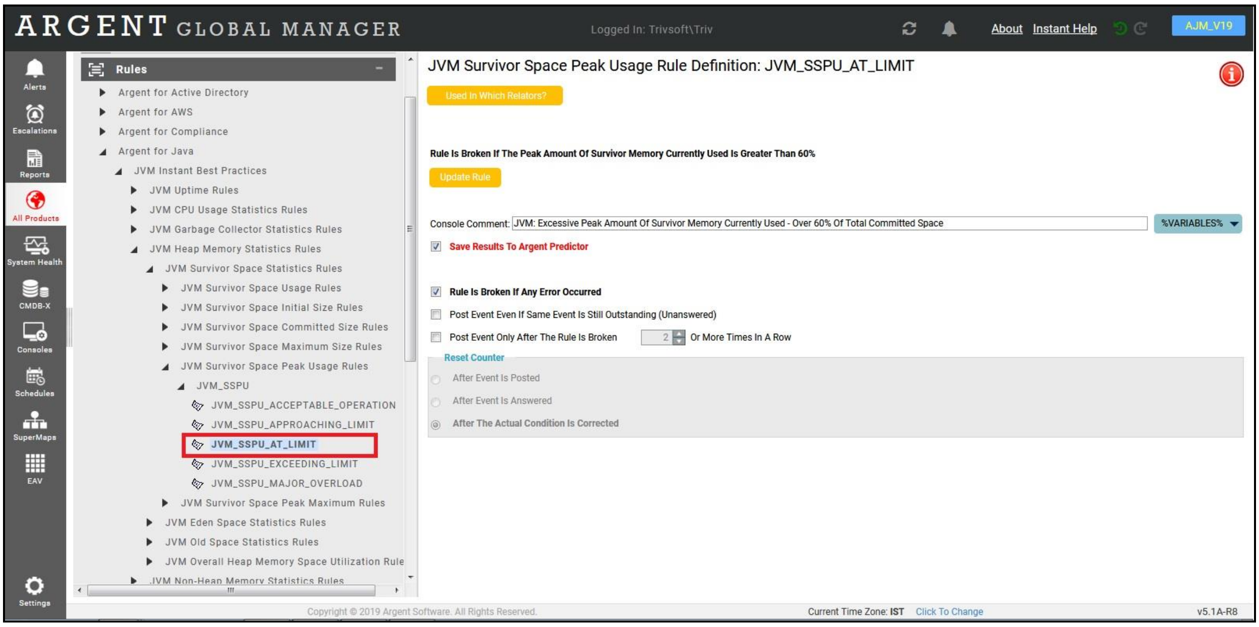
○ After Event Is Posted

○ After Event Is Answered

● After The Actual Condition Is Corrected

Survivor Space Peak Usage

This rule tracks the maximum usage of survivor memory at any given point of time as a percentage of usage.



ARGENT GLOBAL MANAGER
Logged In: Trivsoft\Triv
About
Instant Help
AJM_V19
Alerts
Escalations
Reports
All Products
System Health
CMDB-X
Consoles
Schedules
SuperMaps
EAV
Settings
Rules
Argent for Active Directory
Argent for AWS
Argent for Compliance
Argent for Java
JVM Instant Best Practices
JVM Uptime Rules
JVM CPU Usage Statistics Rules
JVM Garbage Collector Statistics Rules
JVM Heap Memory Statistics Rules
JVM Survivor Space Statistics Rules
JVM Survivor Space Usage Rules
JVM Survivor Space Initial Size Rules
JVM Survivor Space Committed Size Rules
JVM Survivor Space Maximum Size Rules
JVM Survivor Space Peak Usage Rules
JVM_SSPU
JVM_SSPU_ACCEPTABLE_OPERATION
JVM_SSPU_APPROACHING_LIMIT
JVM_SSPU_AT_LIMIT
JVM_SSPU_EXCEEDING_LIMIT
JVM_SSPU_MAJOR_OVERLOAD
JVM Survivor Space Peak Maximum Rules
JVM Eden Space Statistics Rules
JVM Old Space Statistics Rules
JVM Overall Heap Memory Space Utilization Rule
JVM Non-Heap Memory Statistics Rules
Used In Which Relators?
Update Rule
%VARIABLES%
Copyright © 2019 Argent Software. All Rights Reserved.
Current Time Zone: IST
Click To Change
v5.1A-R8

JVM Survivor Space Peak Usage Rule Definition: JVM_SSPU_AT_LIMIT

Rule Is Broken If The Peak Amount Of Survivor Memory Currently Used Is Greater Than 60%

Console Comment: JVM: Excessive Peak Amount Of Survivor Memory Currently Used - Over 60% Of Total Committed Space

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

□ Post Event Even If Same Event Is Still Outstanding (Unanswered)

□ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

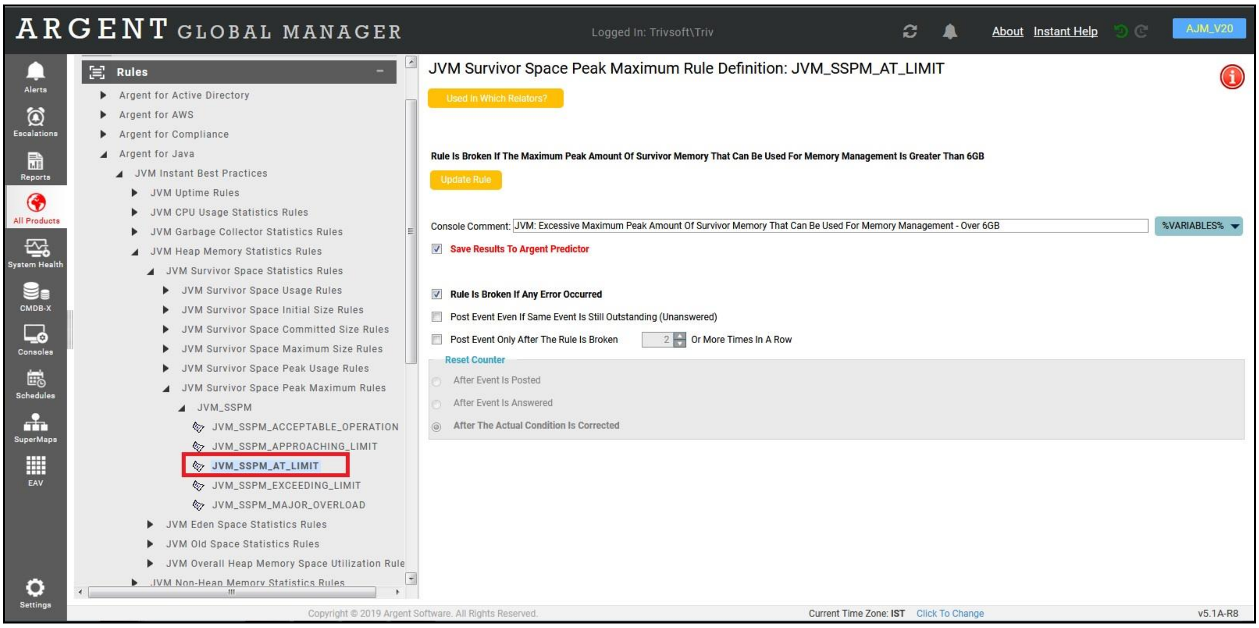
○ After Event Is Posted

○ After Event Is Answered

● After The Actual Condition Is Corrected

Survivor Space Peak Maximum Size

This rule tracks the maximum space utilized by survivor memory at any given point in time


ARGENT GLOBAL MANAGER
Logged In: Trivsoft\Triv
About
Instant Help
AJM_V20
Alerts
Escalations
Reports
All Products
System Health
CMDB-X
Consoles
Schedules
SuperMaps
EAV
Settings
Rules
Argent for Active Directory
Argent for AWS
Argent for Compliance
Argent for Java
JVM Instant Best Practices
JVM Uptime Rules
JVM CPU Usage Statistics Rules
JVM Garbage Collector Statistics Rules
JVM Heap Memory Statistics Rules
JVM Survivor Space Statistics Rules
JVM Survivor Space Usage Rules
JVM Survivor Space Initial Size Rules
JVM Survivor Space Committed Size Rules
JVM Survivor Space Maximum Size Rules
JVM Survivor Space Peak Usage Rules
JVM Survivor Space Peak Maximum Rules
JVM_SSPM
JVM_SSPM_ACCEPTABLE_OPERATION
JVM_SSPM_APPROACHING_LIMIT
JVM_SSPM_AT_LIMIT
JVM_SSPM_EXCEEDING_LIMIT
JVM_SSPM_MAJOR_OVERLOAD
JVM Eden Space Statistics Rules
JVM Old Space Statistics Rules
JVM Overall Heap Memory Space Utilization Rule
JVM Non-Heap Memory Statistics Rules
Used In Which Relators?
Update Rule
%VARIABLES%
Copyright © 2019 Argent Software. All Rights Reserved.
Current Time Zone: IST
Click To Change
v5.1A-R8

JVM Survivor Space Peak Maximum Rule Definition: JVM_SSPM_AT_LIMIT

Rule Is Broken If The Maximum Peak Amount Of Survivor Memory That Can Be Used For Memory Management Is Greater Than 6GB

Console Comment: JVM: Excessive Maximum Peak Amount Of Survivor Memory That Can Be Used For Memory Management - Over 6GB

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

□ Post Event Even If Same Event Is Still Outstanding (Unanswered)

□ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

○ After Event Is Posted

○ After Event Is Answered

● After The Actual Condition Is Corrected

Eden Space Rules

Newly created objects are allocated in Eden Space within a JVM.   Many newly created objects become de-referenced and are inaccessible soon after their creation. Garbage collection moves some of these not-yet-dereferenced objects into Survivor Space.

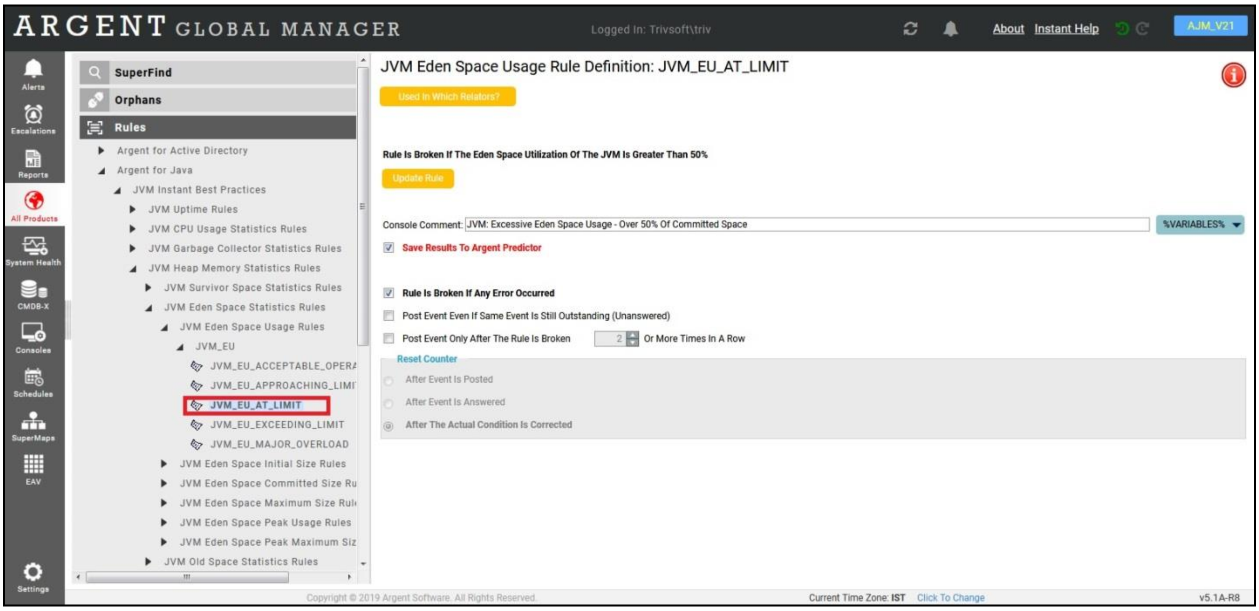
Argent for Java monitors the Eden Space consumption in a JVM.

Argent for Java provides support for the following Eden space rules:

1) Eden Space Usage Rule
2) Eden Space Initial Size
3) Eden Space Committed Size
4) Eden Space Maximum Size
5) Eden Space Peak Usage
6) Eden Space Peak Maximum Size

Eden Space Usage Rule

This rule monitors Java Virtual Machine's Eden Space utilization


ARGENT GLOBAL MANAGER
Logged In: Trivsoft\triv
About
Instant Help
AJM_V21
Alerts
Escalations
Reports
All Products
System Health
CMDB-X
Consoles
Schedules
SuperMaps
EAV
Settings
SuperFind
Orphans
Rules
Argent for Active Directory
Argent for Java
JVM Instant Best Practices
JVM Uptime Rules
JVM CPU Usage Statistics Rules
JVM Garbage Collector Statistics Rules
JVM Heap Memory Statistics Rules
JVM Survivor Space Statistics Rules
JVM Eden Space Statistics Rules
JVM Eden Space Usage Rules
JVM_EU
JVM_EU_ACCEPTABLE_OPERA
JVM_EU_APPROACHING_LIMI
JVM_EU_AT_LIMIT
JVM_EU_EXCEEDING_LIMIT
JVM_EU_MAJOR_OVERLOAD
JVM Eden Space Initial Size Rules
JVM Eden Space Committed Size Ru
JVM Eden Space Maximum Size Rul
JVM Eden Space Peak Usage Rules
JVM Eden Space Peak Maximum Siz
JVM Old Space Statistics Rules
Used In Which Relators?
Update Rule
%VARIABLES%
Copyright © 2019 Argent Software. All Rights Reserved.
Current Time Zone: IST
Click To Change
v5.1A-R8

JVM Eden Space Usage Rule Definition: JVM_EU_AT_LIMIT

Rule Is Broken If The Eden Space Utilization Of The JVM Is Greater Than 50%

Console Comment: JVM: Excessive Eden Space Usage - Over 50% Of Committed Space

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

□ Post Event Even If Same Event Is Still Outstanding (Unanswered)

□ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

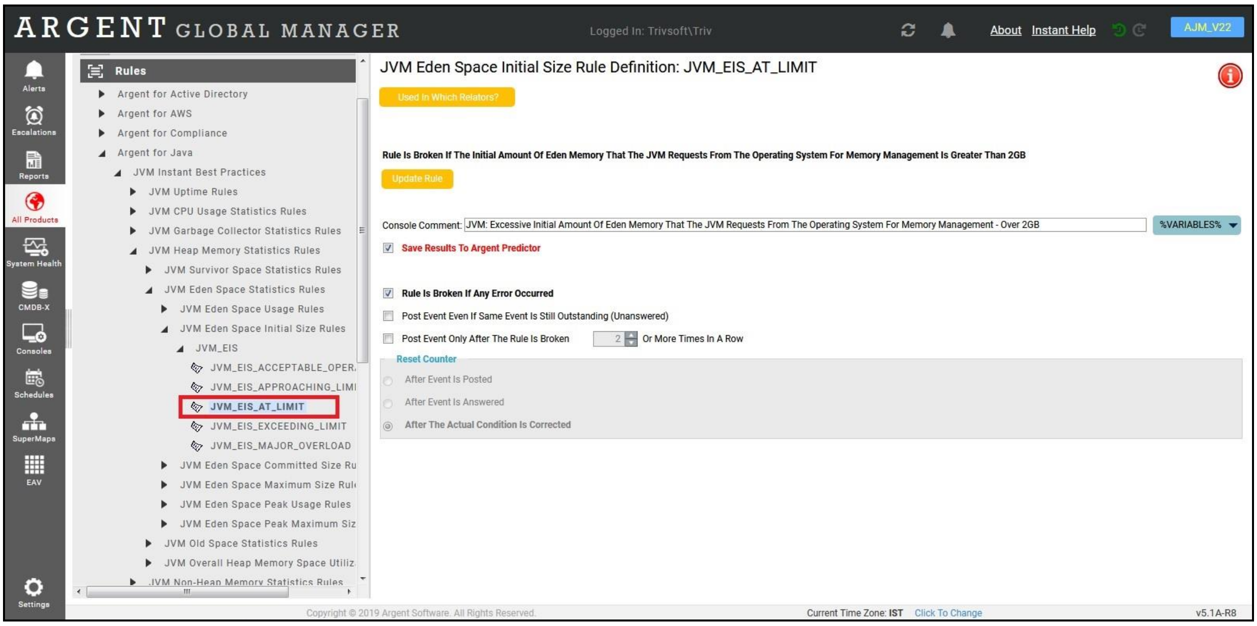
○ After Event Is Posted

○ After Event Is Answered

● After The Actual Condition Is Corrected

Eden Space Initial Size

This rule monitors the initial eden space requested by a JVM


ARGENT GLOBAL MANAGER
Logged In: Trivsoft\Triv
About
Instant Help
AJM_V22
Alerts
Escalations
Reports
All Products
System Health
CMDB-X
Consoles
Schedules
SuperMaps
EAV
Settings
Rules
Argent for Active Directory
Argent for AWS
Argent for Compliance
Argent for Java
JVM Instant Best Practices
JVM Uptime Rules
JVM CPU Usage Statistics Rules
JVM Garbage Collector Statistics Rules
JVM Heap Memory Statistics Rules
JVM Survivor Space Statistics Rules
JVM Eden Space Statistics Rules
JVM Eden Space Usage Rules
JVM Eden Space Initial Size Rules
JVM_EIS
JVM_EIS_ACCEPTABLE_OPER.
JVM_EIS_APPROACHING_LIMI
JVM_EIS_AT_LIMIT
JVM_EIS_EXCEEDING_LIMIT
JVM_EIS_MAJOR_OVERLOAD
JVM Eden Space Committed Size Ru
JVM Eden Space Maximum Size Rul
JVM Eden Space Peak Usage Rules
JVM Eden Space Peak Maximum Siz
JVM Old Space Statistics Rules
JVM Overall Heap Memory Space Utiliz
JVM Non-Heap Memory Statistics Rules
Used In Which Relators?
Update Rule
%VARIABLES%
Copyright © 2019 Argent Software. All Rights Reserved.
Current Time Zone: IST
Click To Change
v5.1A-R8

JVM Eden Space Initial Size Rule Definition: JVM_EIS_AT_LIMIT

Rule Is Broken If The Initial Amount Of Eden Memory That The JVM Requests From The Operating System For Memory Management Is Greater Than 2GB

Console Comment: JVM: Excessive Initial Amount Of Eden Memory That The JVM Requests From The Operating System For Memory Management - Over 2GB

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

□ Post Event Even If Same Event Is Still Outstanding (Unanswered)

□ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

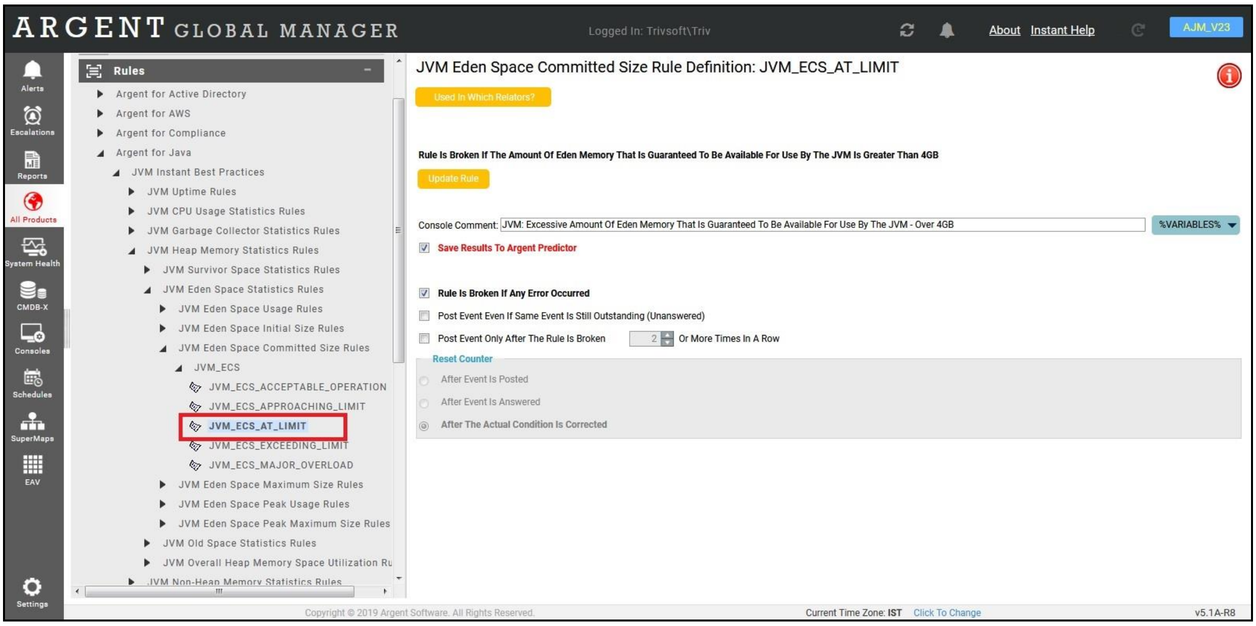
○ After Event Is Posted

○ After Event Is Answered

● After The Actual Condition Is Corrected

Eden Space Committed Size

This rule monitors the committed size of eden space for a JVM


ARGENT GLOBAL MANAGER
Logged In: Trivsoft\Triv
About
Instant Help
AJM_V23
Alerts
Escalations
Reports
All Products
System Health
CMDB-X
Consoles
Schedules
SuperMaps
EAV
Settings
Rules
Argent for Active Directory
Argent for AWS
Argent for Compliance
Argent for Java
JVM Instant Best Practices
JVM Uptime Rules
JVM CPU Usage Statistics Rules
JVM Garbage Collector Statistics Rules
JVM Heap Memory Statistics Rules
JVM Survivor Space Statistics Rules
JVM Eden Space Statistics Rules
JVM Eden Space Usage Rules
JVM Eden Space Initial Size Rules
JVM Eden Space Committed Size Rules
JVM_ECS
JVM_ECS_ACCEPTABLE_OPERATION
JVM_ECS_APPROACHING_LIMIT
JVM_ECS_AT_LIMIT
JVM_ECS_EXCEEDING_LIMIT
JVM_ECS_MAJOR_OVERLOAD
JVM Eden Space Maximum Size Rules
JVM Eden Space Peak Usage Rules
JVM Eden Space Peak Maximum Size Rules
JVM Old Space Statistics Rules
JVM Overall Heap Memory Space Utilization Ru
JVM Non-Heap Memory Statistics Rules
Used In Which Relators?
Update Rule
%VARIABLES%
Copyright © 2019 Argent Software. All Rights Reserved.
Current Time Zone: IST
Click To Change
v5.1A-R8

JVM Eden Space Committed Size Rule Definition: JVM_ECS_AT_LIMIT

Rule Is Broken If The Amount Of Eden Memory That Is Guaranteed To Be Available For Use By The JVM Is Greater Than 4GB

Console Comment: JVM: Excessive Amount Of Eden Memory That Is Guaranteed To Be Available For Use By The JVM - Over 4GB

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

□ Post Event Even If Same Event Is Still Outstanding (Unanswered)

□ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

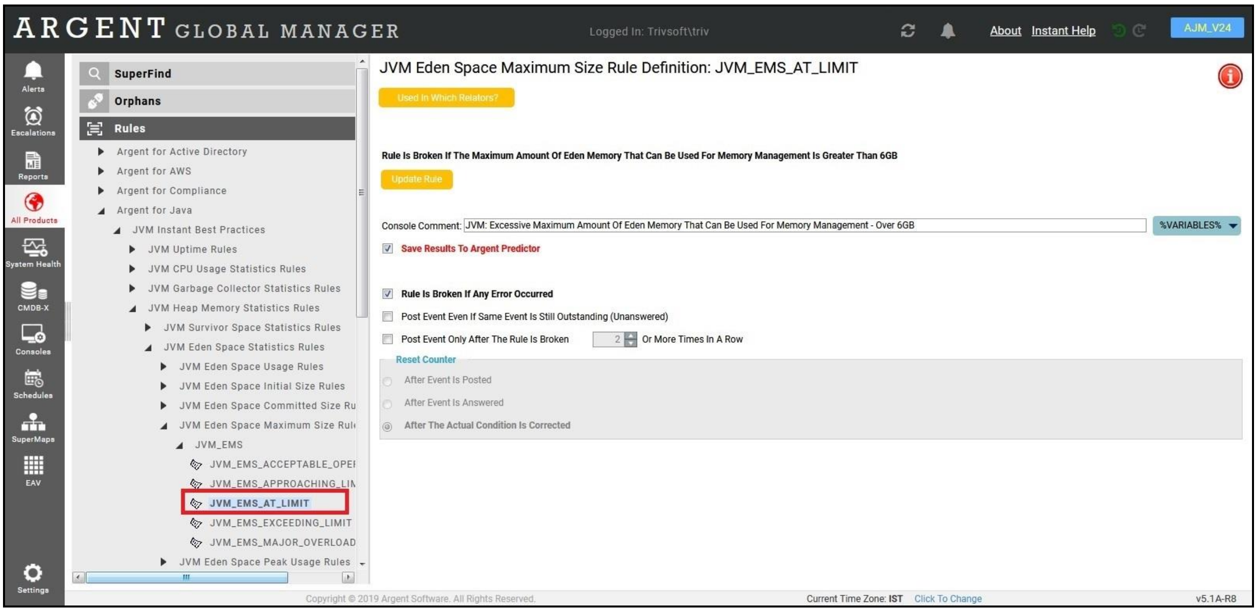
○ After Event Is Posted

○ After Event Is Answered

● After The Actual Condition Is Corrected

Eden Space Maximum Size

This rule monitors the maximum size of eden memory in a JVM



JVM Eden Space Maximum Size Rule Definition: JVM_EMS_AT_LIMIT

Rule Is Broken If The Maximum Amount Of Eden Memory That Can Be Used For Memory Management Is Greater Than 6GB

Console Comment: JVM: Excessive Maximum Amount Of Eden Memory That Can Be Used For Memory Management - Over 6GB

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

☐ Post Event Even If Same Event Is Still Outstanding (Unanswered)

☐ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

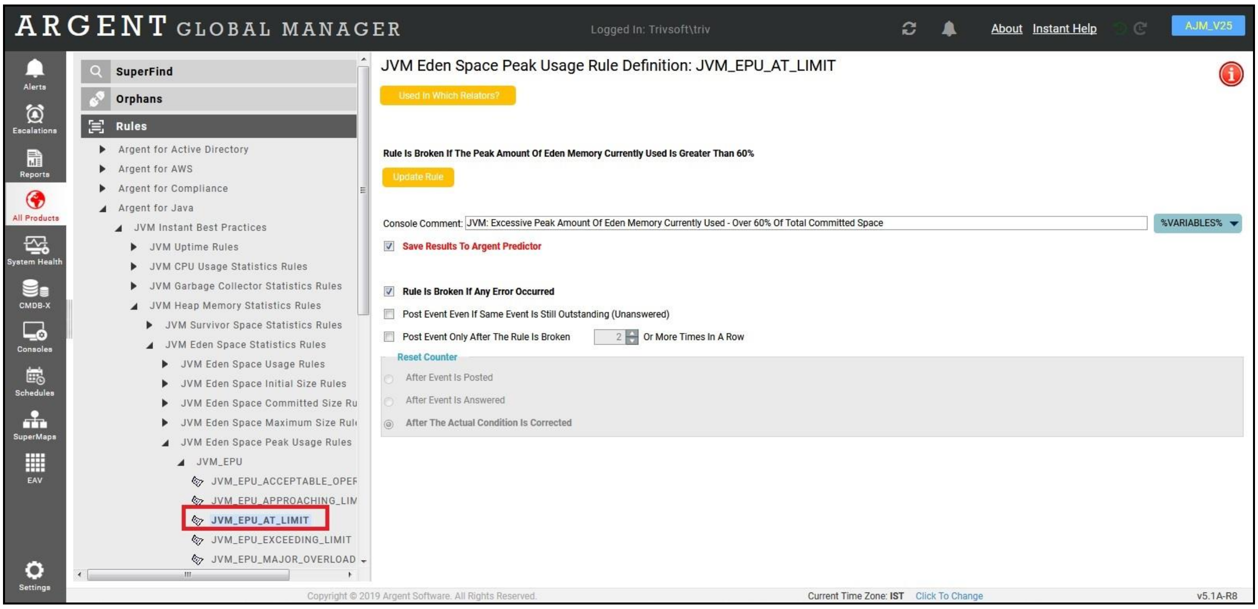
○ After Event Is Posted

○ After Event Is Answered

● After The Actual Condition Is Corrected

Eden Space Peak Usage

This rule monitors the peak usage of eden memory



JVM Eden Space Peak Usage Rule Definition: JVM_EPU_AT_LIMIT

Rule Is Broken If The Peak Amount Of Eden Memory Currently Used Is Greater Than 60%

Console Comment: JVM: Excessive Peak Amount Of Eden Memory Currently Used - Over 60% Of Total Committed Space

☑ **Save Results To Argent Predictor**

☑ **Rule Is Broken If Any Error Occurred**

☐ Post Event Even If Same Event Is Still Outstanding (Unanswered)

☐ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

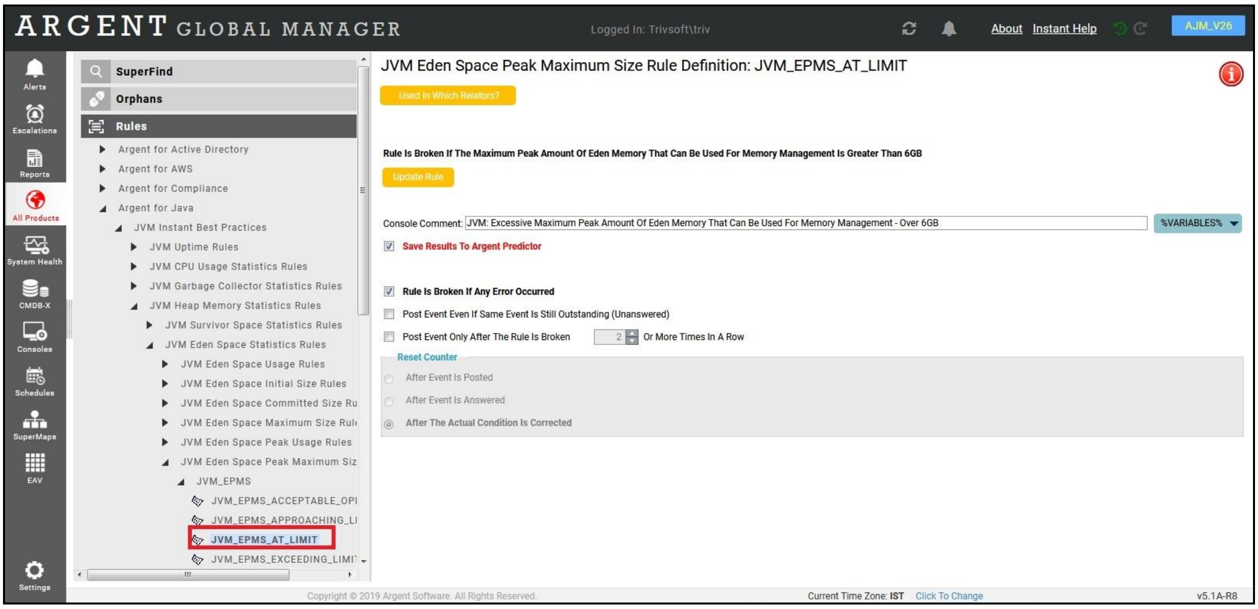
○ After Event Is Posted

○ After Event Is Answered

● After The Actual Condition Is Corrected

Eden Space Peak Maximum Size

This rule monitors the peak maximum size of eden memory


ARGENT GLOBAL MANAGER
Logged In: Trivsoft\triv
About
Instant Help
AJM_V26
Alerts
Escalations
Reports
All Products
System Health
CMDB-X
Consoles
Schedules
SuperMaps
EAV
Settings
SuperFind
Orphans
Rules
Argent for Active Directory
Argent for AWS
Argent for Compliance
Argent for Java
JVM Instant Best Practices
JVM Uptime Rules
JVM CPU Usage Statistics Rules
JVM Garbage Collector Statistics Rules
JVM Heap Memory Statistics Rules
JVM Survivor Space Statistics Rules
JVM Eden Space Statistics Rules
JVM Eden Space Usage Rules
JVM Eden Space Initial Size Rules
JVM Eden Space Committed Size Ru
JVM Eden Space Maximum Size Rul
JVM Eden Space Peak Usage Rules
JVM Eden Space Peak Maximum Siz
JVM_EPMS
JVM_EPMS_ACCEPTABLE_OPI
JVM_EPMS_APPROACHING_LI
JVM_EPMS_AT_LIMIT
JVM_EPMS_EXCEEDING_LIMIT
Used In Which Relators?
Update Rule
%VARIABLES%
Copyright © 2019 Argent Software. All Rights Reserved.
Current Time Zone: IST
Click To Change
v5.1A-R8

JVM Eden Space Peak Maximum Size Rule Definition: JVM_EPMS_AT_LIMIT

Rule Is Broken If The Maximum Peak Amount Of Eden Memory That Can Be Used For Memory Management Is Greater Than 6GB

Console Comment: JVM: Excessive Maximum Peak Amount Of Eden Memory That Can Be Used For Memory Management - Over 6GB

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

☐ Post Event Even If Same Event Is Still Outstanding (Unanswered)

☐ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

○ After Event Is Posted

○ After Event Is Answered

● After The Actual Condition Is Corrected

Old Space Rules

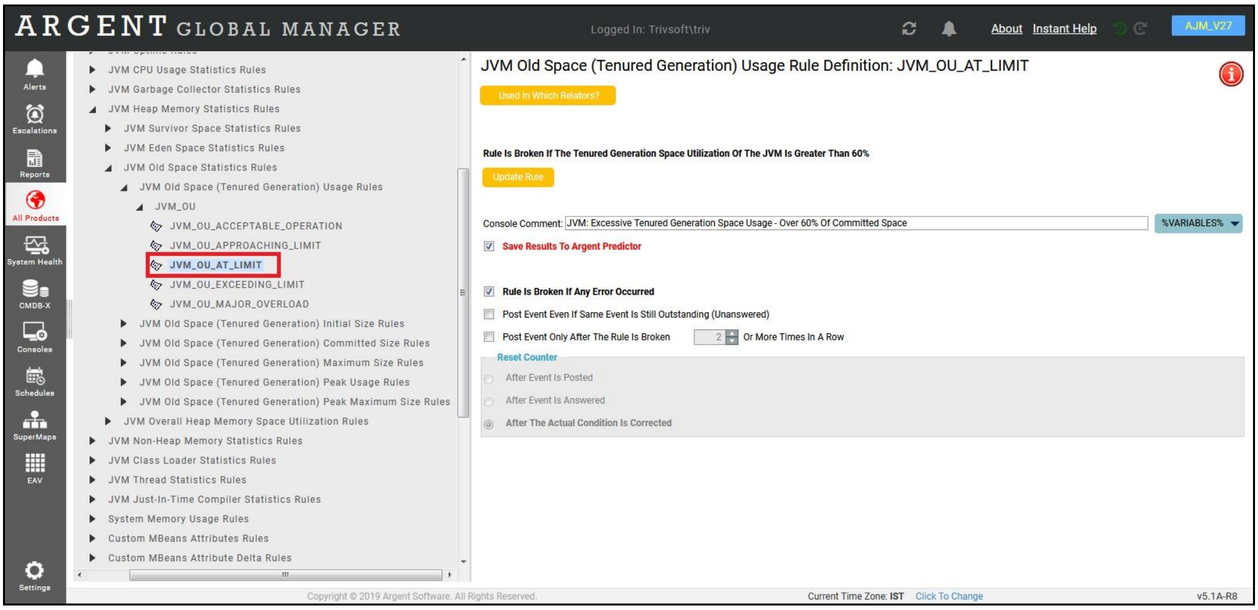
Java virtual machines (JVMs) use the old (or tenured) space in heap memory to store objects that are required for long durations.

Argent for Java provides support for the following old space rules:

1) Old Space Utilization
2) Old Space Initial Size
3) Old Space Committed Size
4) Old Space Maximum Size
5) Old Space Peak Usage
6) Old Space Peak Maximum Size
7) Overall Heap Memory Utilization

Old Space Utilization

This rule monitors old space utilization in a JVM


ARGENT GLOBAL MANAGER
Logged In: Trivsoft\triv
About
Instant Help
AJM_V27
Alerts
Escalations
Reports
All Products
System Health
CMDB-X
Consoles
Schedules
SuperMaps
EAV
Settings
JVM CPU Usage Statistics Rules
JVM Garbage Collector Statistics Rules
JVM Heap Memory Statistics Rules
JVM Survivor Space Statistics Rules
JVM Eden Space Statistics Rules
JVM Old Space Statistics Rules
JVM Old Space (Tenured Generation) Usage Rules
JVM_OU
JVM_OU_ACCEPTABLE_OPERATION
JVM_OU_APPROACHING_LIMIT
JVM_OU_AT_LIMIT
JVM_OU_EXCEEDING_LIMIT
JVM_OU_MAJOR_OVERLOAD
JVM Old Space (Tenured Generation) Initial Size Rules
JVM Old Space (Tenured Generation) Committed Size Rules
JVM Old Space (Tenured Generation) Maximum Size Rules
JVM Old Space (Tenured Generation) Peak Usage Rules
JVM Old Space (Tenured Generation) Peak Maximum Size Rules
JVM Overall Heap Memory Space Utilization Rules
JVM Non-Heap Memory Statistics Rules
JVM Class Loader Statistics Rules
JVM Thread Statistics Rules
JVM Just-In-Time Compiler Statistics Rules
System Memory Usage Rules
Custom MBeans Attributes Rules
Custom MBeans Attribute Delta Rules
Used In Which Relators?
Update Rule
%VARIABLES%
Copyright © 2019 Argent Software. All Rights Reserved.
Current Time Zone: IST
Click To Change
v5.1A-R8

JVM Old Space (Tenured Generation) Usage Rule Definition: JVM_OU_AT_LIMIT

Rule Is Broken If The Tenured Generation Space Utilization Of The JVM Is Greater Than 60%

Console Comment: JVM: Excessive Tenured Generation Space Usage - Over 60% Of Committed Space

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

□ Post Event Even If Same Event Is Still Outstanding (Unanswered)

□ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

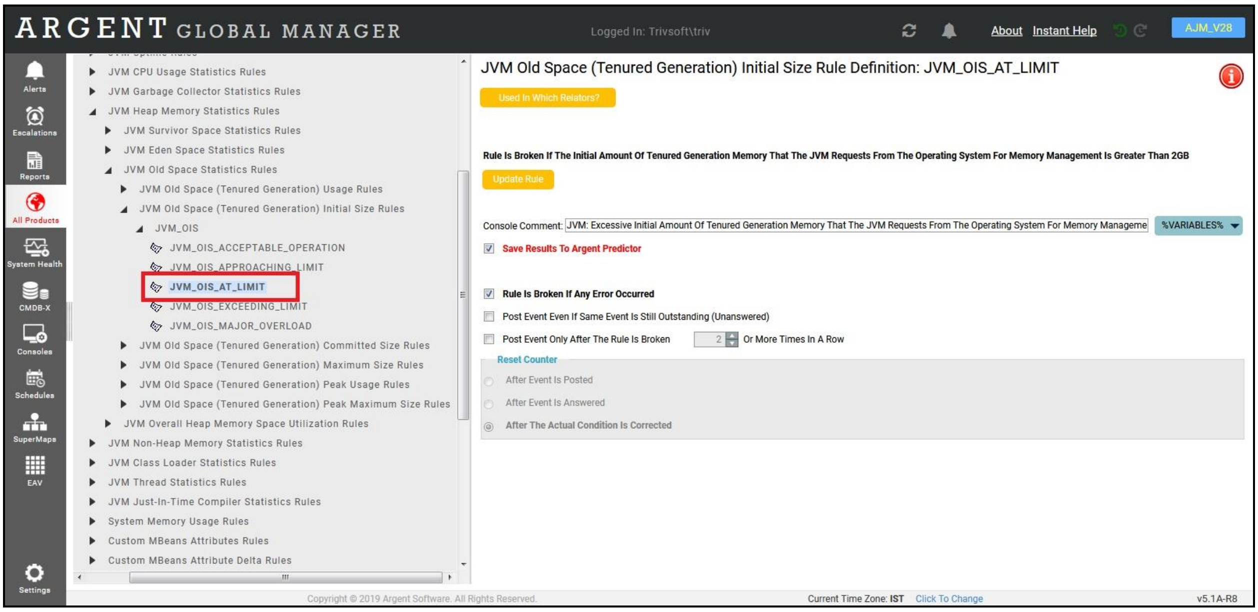
○ After Event Is Posted

○ After Event Is Answered

● After The Actual Condition Is Corrected

Old Space Initial Size

This rule monitors the initial size of the old or tenured memory requested by a JVM


ARGENT GLOBAL MANAGER
Logged In: Trivsoft\triv
About
Instant Help
AJM_V28
Alerts
Escalations
Reports
All Products
System Health
CMDB-X
Consoles
Schedules
SuperMaps
EAV
Settings
JVM CPU Usage Statistics Rules
JVM Garbage Collector Statistics Rules
JVM Heap Memory Statistics Rules
JVM Survivor Space Statistics Rules
JVM Eden Space Statistics Rules
JVM Old Space Statistics Rules
JVM Old Space (Tenured Generation) Usage Rules
JVM Old Space (Tenured Generation) Initial Size Rules
JVM_OIS
JVM_OIS_ACCEPTABLE_OPERATION
JVM_OIS_APPROACHING_LIMIT
JVM_OIS_AT_LIMIT
JVM_OIS_EXCEEDING_LIMIT
JVM_OIS_MAJOR_OVERLOAD
JVM Old Space (Tenured Generation) Committed Size Rules
JVM Old Space (Tenured Generation) Maximum Size Rules
JVM Old Space (Tenured Generation) Peak Usage Rules
JVM Old Space (Tenured Generation) Peak Maximum Size Rules
JVM Overall Heap Memory Space Utilization Rules
JVM Non-Heap Memory Statistics Rules
JVM Class Loader Statistics Rules
JVM Thread Statistics Rules
JVM Just-In-Time Compiler Statistics Rules
System Memory Usage Rules
Custom MBeans Attributes Rules
Custom MBeans Attribute Delta Rules
Used In Which Relators?
Update Rule
%VARIABLES%

JVM Old Space (Tenured Generation) Initial Size Rule Definition: JVM_OIS_AT_LIMIT

Rule Is Broken If The Initial Amount Of Tenured Generation Memory That The JVM Requests From The Operating System For Memory Management Is Greater Than 2GB

Console Comment: JVM: Excessive Initial Amount Of Tenured Generation Memory That The JVM Requests From The Operating System For Memory Manageme

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

□ Post Event Even If Same Event Is Still Outstanding (Unanswered)

□ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

○ After Event Is Posted

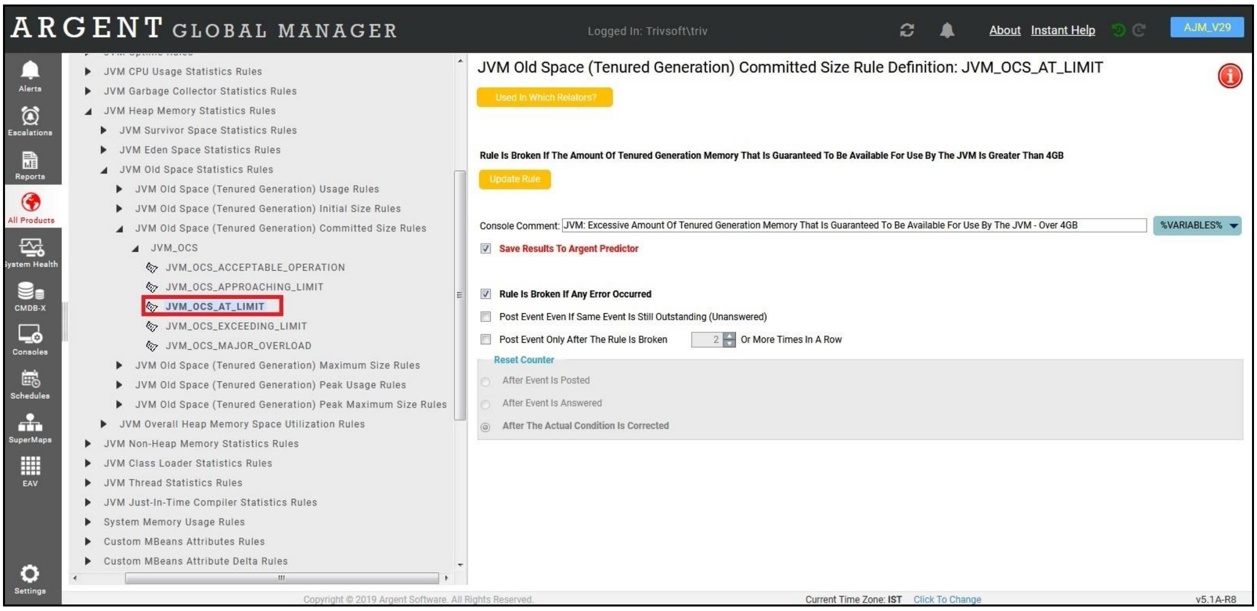
○ After Event Is Answered

● After The Actual Condition Is Corrected

Copyright © 2019 Argent Software. All Rights Reserved. Current Time Zone: IST Click To Change v5.1A-R8

Old Space Committed Size

This rule monitors the commited size of the old or tenured memory available for use by a JVM



- JVM CPU Usage Statistics Rules
- JVM Garbage Collector Statistics Rules
- JVM Heap Memory Statistics Rules
  - JVM Survivor Space Statistics Rules
  - JVM Eden Space Statistics Rules
  - JVM Old Space Statistics Rules
    - JVM Old Space (Tenured Generation) Usage Rules
    - JVM Old Space (Tenured Generation) Initial Size Rules
    - JVM Old Space (Tenured Generation) Committed Size Rules
      - JVM_OCS
        - JVM_OCS_ACCEPTABLE_OPERATION
        - JVM_OCS_APPROACHING_LIMIT
        - JVM_OCS_AT_LIMIT
        - JVM_OCS_EXCEEDING_LIMIT
        - JVM_OCS_MAJOR_OVERLOAD
    - JVM Old Space (Tenured Generation) Maximum Size Rules
    - JVM Old Space (Tenured Generation) Peak Usage Rules
    - JVM Old Space (Tenured Generation) Peak Maximum Size Rules
  - JVM Overall Heap Memory Space Utilization Rules
- JVM Non-Heap Memory Statistics Rules
- JVM Class Loader Statistics Rules
- JVM Thread Statistics Rules
- JVM Just-In-Time Compiler Statistics Rules
- System Memory Usage Rules
- Custom MBeans Attributes Rules
- Custom MBeans Attribute Delta Rules

JVM Old Space (Tenured Generation) Committed Size Rule Definition: JVM_OCS_AT_LIMIT

Used In Which Relators?

Rule Is Broken If The Amount Of Tenured Generation Memory That Is Guaranteed To Be Available For Use By The JVM Is Greater Than 4GB

Update Rule

Console Comment: JVM: Excessive Amount Of Tenured Generation Memory That Is Guaranteed To Be Available For Use By The JVM - Over 4GB %VARIABLES%

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

☐ Post Event Even If Same Event Is Still Outstanding (Unanswered)

☐ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

○ After Event Is Posted

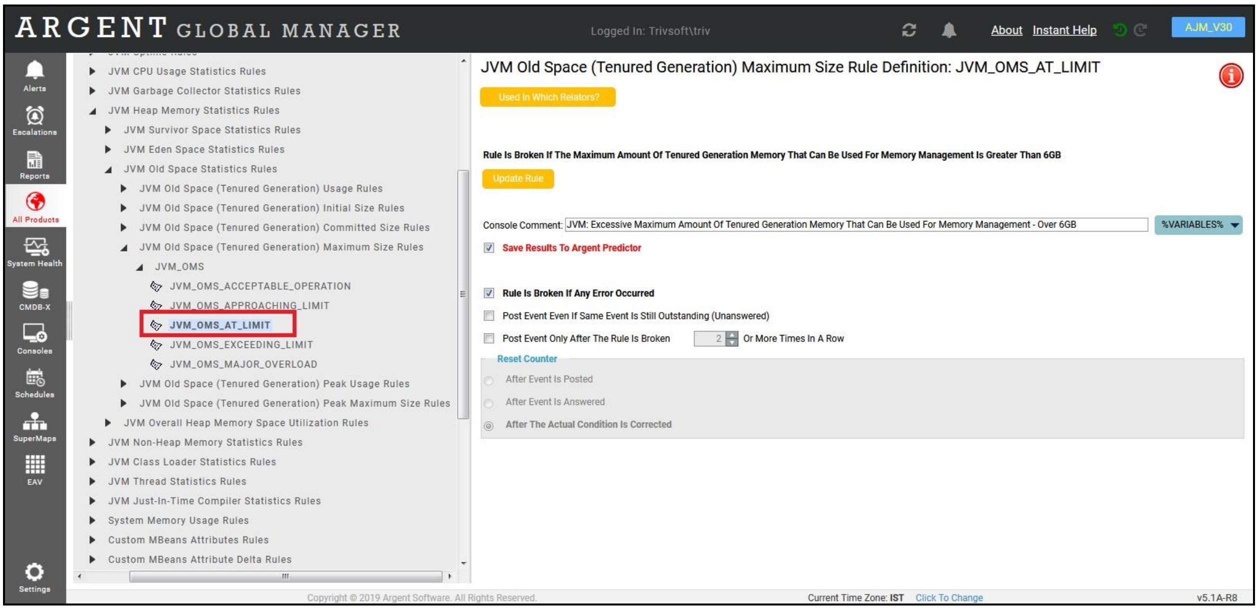
○ After Event Is Answered

● After The Actual Condition Is Corrected

Copyright © 2019 Argent Software. All Rights Reserved. Current Time Zone: IST Click To Change v5.1A-R8

Old Space Maximum Size

This rule monitors the maximum size of the old or tenured memory in a JVM


ARGENT GLOBAL MANAGER
Logged In: Trivsoft\triv
About
Instant Help
AJM_V30
Alerts
Escalations
Reports
All Products
System Health
CMDB-X
Consoles
Schedules
SuperMaps
EAV
Settings
JVM CPU Usage Statistics Rules
JVM Garbage Collector Statistics Rules
JVM Heap Memory Statistics Rules
JVM Survivor Space Statistics Rules
JVM Eden Space Statistics Rules
JVM Old Space Statistics Rules
JVM Old Space (Tenured Generation) Usage Rules
JVM Old Space (Tenured Generation) Initial Size Rules
JVM Old Space (Tenured Generation) Committed Size Rules
JVM Old Space (Tenured Generation) Maximum Size Rules
JVM_OMS
JVM_OMS_ACCEPTABLE_OPERATION
JVM_OMS_APPROACHING_LIMIT
JVM_OMS_AT_LIMIT
JVM_OMS_EXCEEDING_LIMIT
JVM_OMS_MAJOR_OVERLOAD
JVM Old Space (Tenured Generation) Peak Usage Rules
JVM Old Space (Tenured Generation) Peak Maximum Size Rules
JVM Overall Heap Memory Space Utilization Rules
JVM Non-Heap Memory Statistics Rules
JVM Class Loader Statistics Rules
JVM Thread Statistics Rules
JVM Just-In-Time Compiler Statistics Rules
System Memory Usage Rules
Custom MBeans Attributes Rules
Custom MBeans Attribute Delta Rules
Used In Which Relators?
Update Rule
%VARIABLES%
Copyright © 2019 Argent Software. All Rights Reserved.
Current Time Zone: IST
Click To Change
v5.1A-R8

JVM Old Space (Tenured Generation) Maximum Size Rule Definition: JVM_OMS_AT_LIMIT

Rule Is Broken If The Maximum Amount Of Tenured Generation Memory That Can Be Used For Memory Management Is Greater Than 6GB

Console Comment: JVM: Excessive Maximum Amount Of Tenured Generation Memory That Can Be Used For Memory Management - Over 6GB

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

□ Post Event Even If Same Event Is Still Outstanding (Unanswered)

□ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

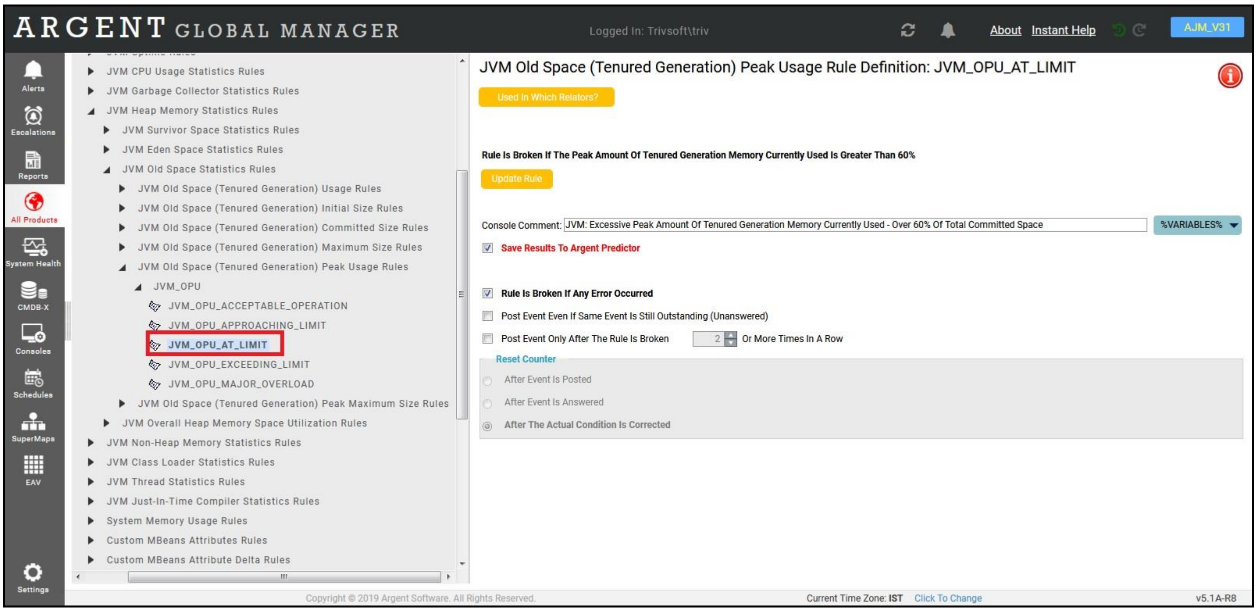
○ After Event Is Posted

○ After Event Is Answered

● After The Actual Condition Is Corrected

Old Space Peak Usage

This rule monitors the peak usage of old or tenured memory in a JVM at any given time



- JVM CPU Usage Statistics Rules
- JVM Garbage Collector Statistics Rules
- JVM Heap Memory Statistics Rules
  - JVM Survivor Space Statistics Rules
  - JVM Eden Space Statistics Rules
  - JVM Old Space Statistics Rules
    - JVM Old Space (Tenured Generation) Usage Rules
    - JVM Old Space (Tenured Generation) Initial Size Rules
    - JVM Old Space (Tenured Generation) Committed Size Rules
    - JVM Old Space (Tenured Generation) Maximum Size Rules
    - JVM Old Space (Tenured Generation) Peak Usage Rules
      - JVM_OPU
        - JVM_OPU_ACCEPTABLE_OPERATION
        - JVM_OPU_APPROACHING_LIMIT
        - JVM_OPU_AT_LIMIT
        - JVM_OPU_EXCEEDING_LIMIT
        - JVM_OPU_MAJOR_OVERLOAD
    - JVM Old Space (Tenured Generation) Peak Maximum Size Rules
  - JVM Overall Heap Memory Space Utilization Rules
- JVM Non-Heap Memory Statistics Rules
- JVM Class Loader Statistics Rules
- JVM Thread Statistics Rules
- JVM Just-In-Time Compiler Statistics Rules
- System Memory Usage Rules
- Custom MBeans Attributes Rules
- Custom MBeans Attribute Delta Rules

JVM Old Space (Tenured Generation) Peak Usage Rule Definition: JVM_OPU_AT_LIMIT

Rule Is Broken If The Peak Amount Of Tenured Generation Memory Currently Used Is Greater Than 60%

Console Comment: JVM: Excessive Peak Amount Of Tenured Generation Memory Currently Used - Over 60% Of Total Committed Space

☑ **Save Results To Argent Predictor**

☑ **Rule Is Broken If Any Error Occurred**

☐ Post Event Even If Same Event Is Still Outstanding (Unanswered)

☐ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

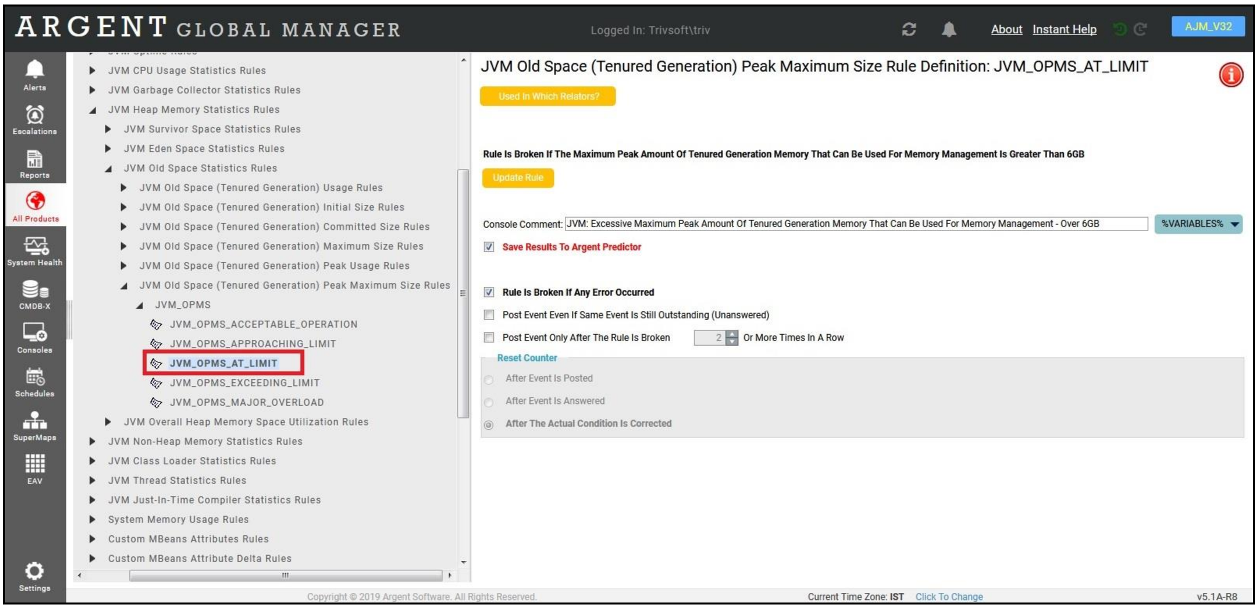
○ After Event Is Posted

○ After Event Is Answered

● After The Actual Condition Is Corrected

Old Space Peak Maximum Size

This rule monitors the maximum peak size of old or tenured memory available in a JVM


ARGENT GLOBAL MANAGER
Logged In: Trivsoft\triv
About
Instant Help
AJM_V32
Alerts
Escalations
Reports
All Products
System Health
CMDB-X
Consoles
Schedules
SuperMaps
EAV
Settings
JVM CPU Usage Statistics Rules
JVM Garbage Collector Statistics Rules
JVM Heap Memory Statistics Rules
JVM Survivor Space Statistics Rules
JVM Eden Space Statistics Rules
JVM Old Space Statistics Rules
JVM Old Space (Tenured Generation) Usage Rules
JVM Old Space (Tenured Generation) Initial Size Rules
JVM Old Space (Tenured Generation) Committed Size Rules
JVM Old Space (Tenured Generation) Maximum Size Rules
JVM Old Space (Tenured Generation) Peak Usage Rules
JVM Old Space (Tenured Generation) Peak Maximum Size Rules
JVM_OPMS
JVM_OPMS_ACCEPTABLE_OPERATION
JVM_OPMS_APPROACHING_LIMIT
JVM_OPMS_AT_LIMIT
JVM_OPMS_EXCEEDING_LIMIT
JVM_OPMS_MAJOR_OVERLOAD
JVM Overall Heap Memory Space Utilization Rules
JVM Non-Heap Memory Statistics Rules
JVM Class Loader Statistics Rules
JVM Thread Statistics Rules
JVM Just-In-Time Compiler Statistics Rules
System Memory Usage Rules
Custom MBeans Attributes Rules
Custom MBeans Attribute Delta Rules
Used In Which Relators?
Update Rule
%VARIABLES%
Copyright © 2019 Argent Software. All Rights Reserved.
Current Time Zone: IST
Click To Change
v5.1A-R8

JVM Old Space (Tenured Generation) Peak Maximum Size Rule Definition: JVM_OPMS_AT_LIMIT

Rule Is Broken If The Maximum Peak Amount Of Tenured Generation Memory That Can Be Used For Memory Management Is Greater Than 6GB

Console Comment: JVM: Excessive Maximum Peak Amount Of Tenured Generation Memory That Can Be Used For Memory Management - Over 6GB

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

□ Post Event Even If Same Event Is Still Outstanding (Unanswered)

□ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

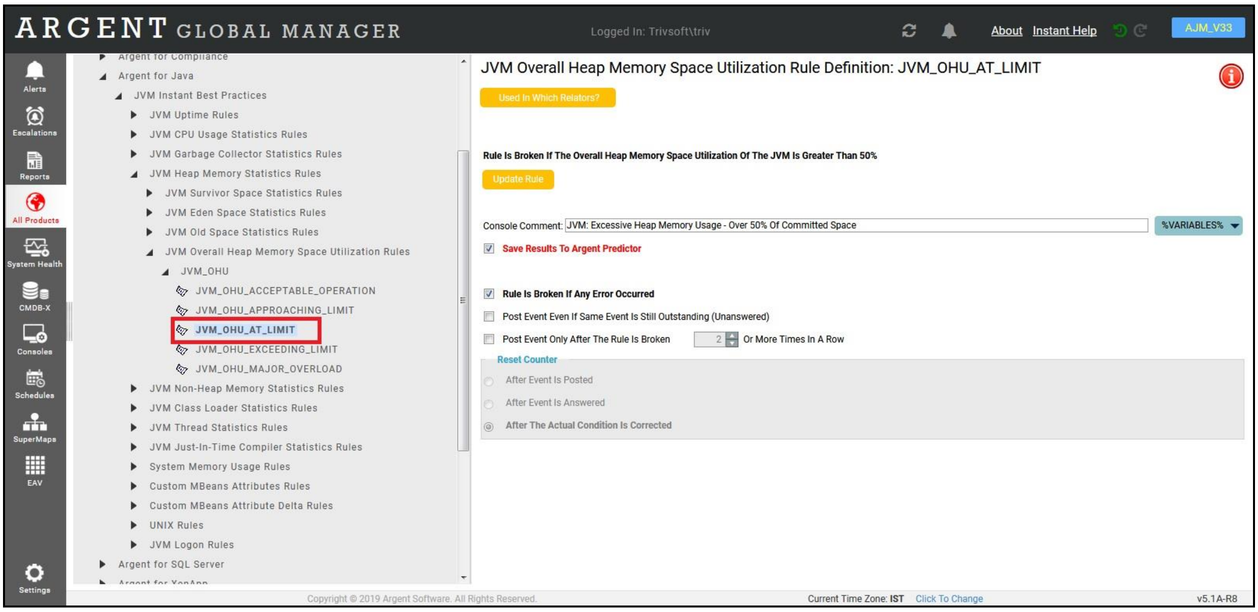
○ After Event Is Posted

○ After Event Is Answered

● After The Actual Condition Is Corrected

Overall Heap Memory Utilization

This rule monitors the overall Heap Memory consumption which is a sum of Eden Space, Survivor space and Old / Tenured Space



- Argent for Compliance
- Argent for Java
  - JVM Instant Best Practices
    - JVM Uptime Rules
    - JVM CPU Usage Statistics Rules
    - JVM Garbage Collector Statistics Rules
    - JVM Heap Memory Statistics Rules
      - JVM Survivor Space Statistics Rules
      - JVM Eden Space Statistics Rules
      - JVM Old Space Statistics Rules
      - JVM Overall Heap Memory Space Utilization Rules
        - JVM_OHU
          - JVM_OHU_ACCEPTABLE_OPERATION
          - JVM_OHU_APPROACHING_LIMIT
          - JVM_OHU_AT_LIMIT
          - JVM_OHU_EXCEEDING_LIMIT
          - JVM_OHU_MAJOR_OVERLOAD
    - JVM Non-Heap Memory Statistics Rules
    - JVM Class Loader Statistics Rules
    - JVM Thread Statistics Rules
    - JVM Just-In-Time Compiler Statistics Rules
    - System Memory Usage Rules
    - Custom MBeans Attributes Rules
    - Custom MBeans Attribute Delta Rules
    - UNIX Rules
    - JVM Logon Rules
- Argent for SQL Server
- Argent for XenApp

JVM Overall Heap Memory Space Utilization Rule Definition: JVM_OHU_AT_LIMIT

Used In Which Relators?

Rule Is Broken If The Overall Heap Memory Space Utilization Of The JVM Is Greater Than 50%

Update Rule

Console Comment: JVM: Excessive Heap Memory Usage - Over 50% Of Committed Space %VARIABLES%

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

□ Post Event Even If Same Event Is Still Outstanding (Unanswered)

□ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

○ After Event Is Posted

○ After Event Is Answered

● After The Actual Condition Is Corrected

Copyright © 2019 Argent Software. All Rights Reserved. Current Time Zone: IST Click To Change v5.1A-R8

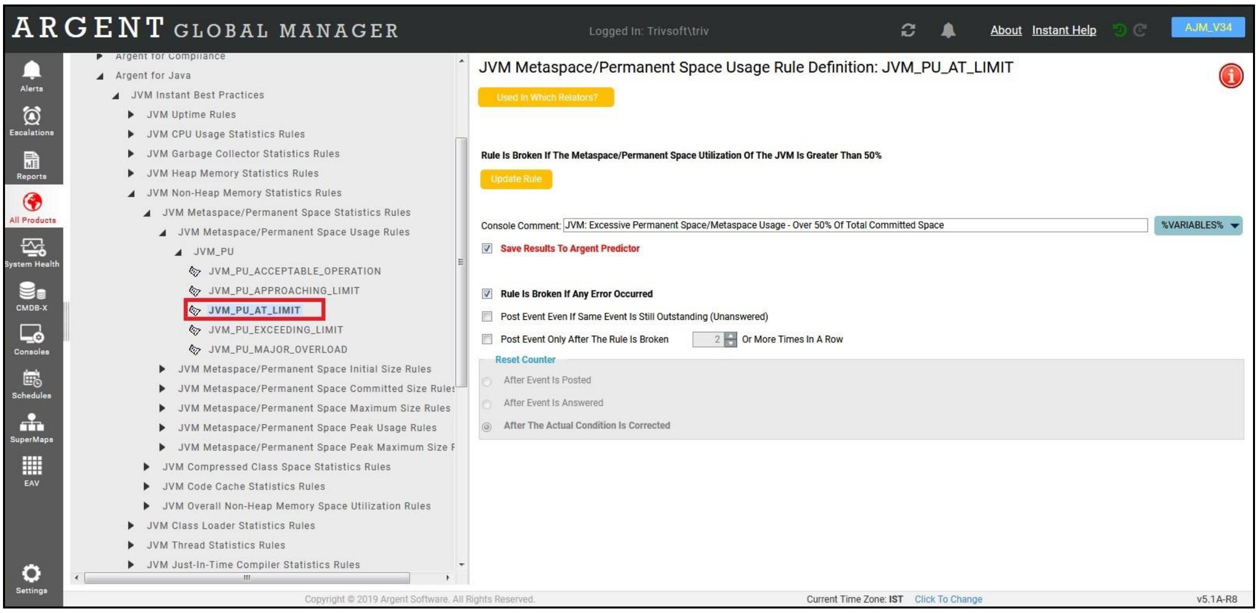
Non-Heap Memory Rules

The non-heap memory is a work area which is shared by a JVM across all threads and is the memory utilized by a JVM for internal processing and optimization. For each class, for example, the non-heap memory contains the runtime constant pool, variables and functional data, and the actual program data for the functions or methods and constructors.

Non-heap memory is further split into Permanent Generation and Code Cache.

Permanent Space Utilization

The permanent space in a JVM contains information such as declared classes and functions. Argent for Java monitors the Permanent Space utilization..



- Argent for Compliance
- Argent for Java
  - JVM Instant Best Practices
    - JVM Uptime Rules
    - JVM CPU Usage Statistics Rules
    - JVM Garbage Collector Statistics Rules
    - JVM Heap Memory Statistics Rules
    - JVM Non-Heap Memory Statistics Rules
      - JVM Metaspace/Permanent Space Statistics Rules
        - JVM Metaspace/Permanent Space Usage Rules
          - JVM_PU
            - JVM_PU_ACCEPTABLE_OPERATION
            - JVM_PU_APPROACHING_LIMIT
            - JVM_PU_AT_LIMIT
            - JVM_PU_EXCEEDING_LIMIT
            - JVM_PU_MAJOR_OVERLOAD
        - JVM Metaspace/Permanent Space Initial Size Rules
        - JVM Metaspace/Permanent Space Committed Size Rules
        - JVM Metaspace/Permanent Space Maximum Size Rules
        - JVM Metaspace/Permanent Space Peak Usage Rules
        - JVM Metaspace/Permanent Space Peak Maximum Size F
      - JVM Compressed Class Space Statistics Rules
      - JVM Code Cache Statistics Rules
      - JVM Overall Non-Heap Memory Space Utilization Rules
    - JVM Class Loader Statistics Rules
    - JVM Thread Statistics Rules
    - JVM Just-In-Time Compiler Statistics Rules

JVM Metaspace/Permanent Space Usage Rule Definition: JVM_PU_AT_LIMIT

Used In Which Relators?

Rule Is Broken If The Metaspace/Permanent Space Utilization Of The JVM Is Greater Than 50%

Update Rule

Console Comment: JVM: Excessive Permanent Space/Metaspace Usage - Over 50% Of Total Committed Space %VARIABLES%

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

☐ Post Event Even If Same Event Is Still Outstanding (Unanswered)

☐ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

○ After Event Is Posted

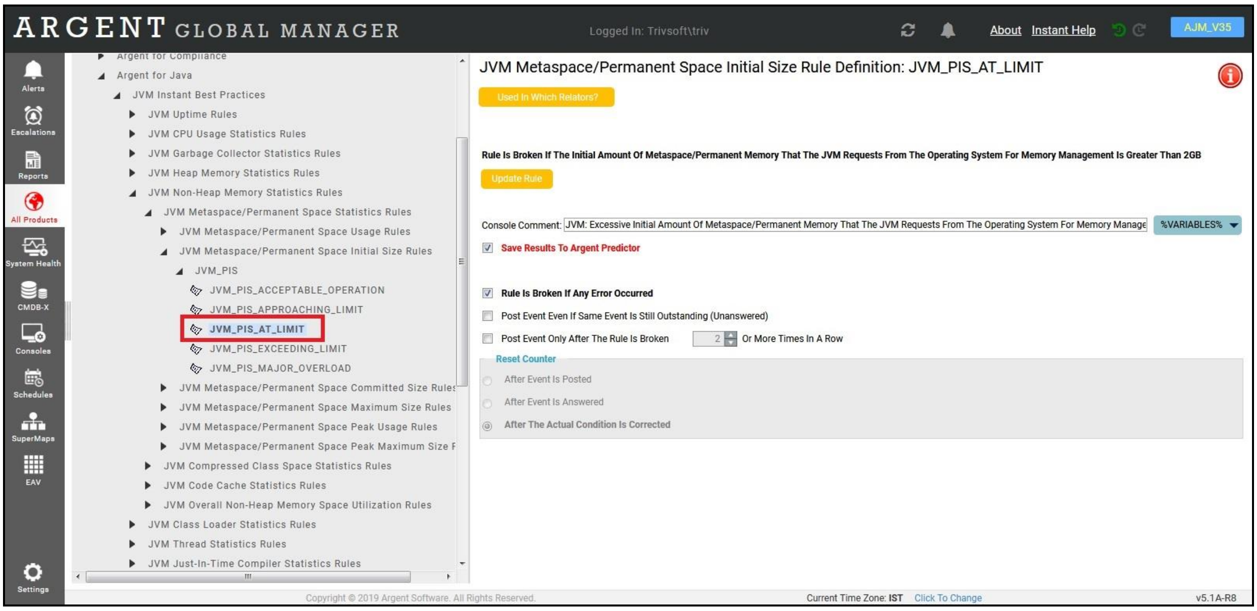
○ After Event Is Answered

● After The Actual Condition Is Corrected

Copyright © 2019 Argent Software. All Rights Reserved. Current Time Zone: IST Click To Change v5.1A-R8

Permanent Space Initial Size

This rule monitors the initial size of metaspace or permanent memory requested by a JVM


ARGENT GLOBAL MANAGER
Logged In: Trivsoft\triv
About
Instant Help
AJM_V35
Alerts
Escalations
Reports
All Products
System Health
CMDB-X
Consoles
Schedules
SuperMaps
EAV
Settings
Argent for Compliance
Argent for Java
JVM Instant Best Practices
JVM Uptime Rules
JVM CPU Usage Statistics Rules
JVM Garbage Collector Statistics Rules
JVM Heap Memory Statistics Rules
JVM Non-Heap Memory Statistics Rules
JVM Metaspace/Permanent Space Statistics Rules
JVM Metaspace/Permanent Space Usage Rules
JVM Metaspace/Permanent Space Initial Size Rules
JVM_PIS
JVM_PIS_ACCEPTABLE_OPERATION
JVM_PIS_APPROACHING_LIMIT
JVM_PIS_AT_LIMIT
JVM_PIS_EXCEEDING_LIMIT
JVM_PIS_MAJOR_OVERLOAD
JVM Metaspace/Permanent Space Committed Size Rules
JVM Metaspace/Permanent Space Maximum Size Rules
JVM Metaspace/Permanent Space Peak Usage Rules
JVM Metaspace/Permanent Space Peak Maximum Size F
JVM Compressed Class Space Statistics Rules
JVM Code Cache Statistics Rules
JVM Overall Non-Heap Memory Space Utilization Rules
JVM Class Loader Statistics Rules
JVM Thread Statistics Rules
JVM Just-In-Time Compiler Statistics Rules
Used In Which Relators?
Update Rule
%VARIABLES%

JVM Metaspace/Permanent Space Initial Size Rule Definition: JVM_PIS_AT_LIMIT

Rule Is Broken If The Initial Amount Of Metaspace/Permanent Memory That The JVM Requests From The Operating System For Memory Management Is Greater Than 2GB

Console Comment: JVM: Excessive Initial Amount Of Metaspace/Permanent Memory That The JVM Requests From The Operating System For Memory Manage

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

□ Post Event Even If Same Event Is Still Outstanding (Unanswered)

□ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

○ After Event Is Posted

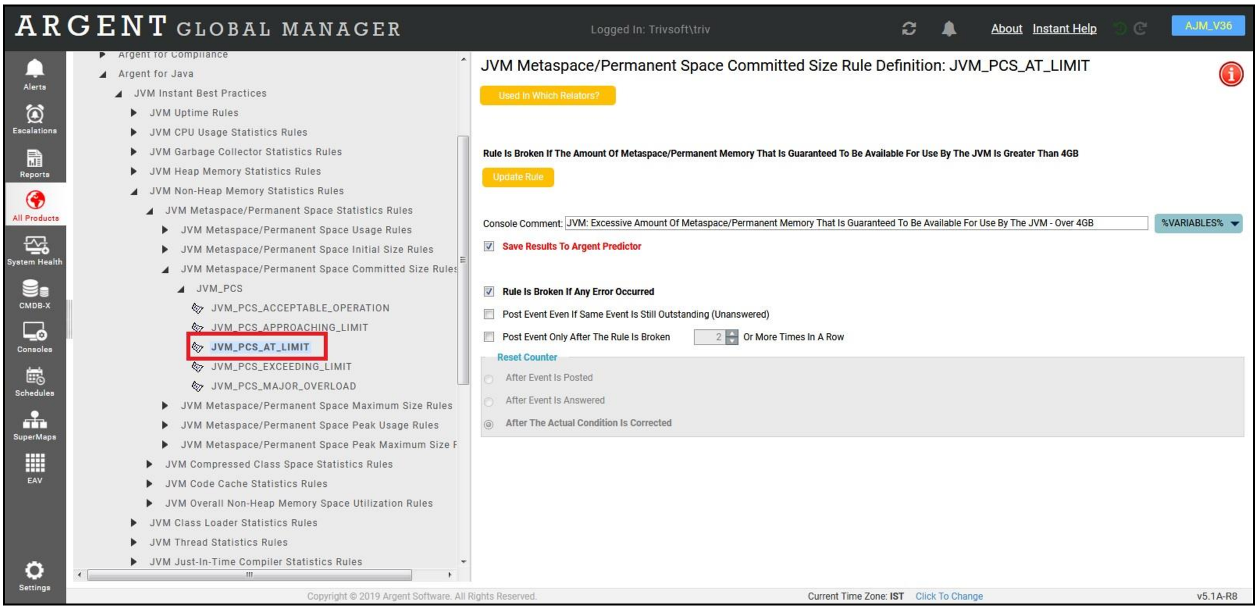
○ After Event Is Answered

● After The Actual Condition Is Corrected

Copyright © 2019 Argent Software. All Rights Reserved. Current Time Zone: IST Click To Change v5.1A-R8

Permanent Space Committed Size

This rule monitors the guaranteed size of permanent memory space available for a JVM.


ARGENT GLOBAL MANAGER
Logged In: Trivsoft\triv
About
Instant Help
AJM_V36
Alerts
Escalations
Reports
All Products
System Health
CMDB-X
Consoles
Schedules
SuperMaps
EAV
Settings
Argent for Compliance
Argent for Java
JVM Instant Best Practices
JVM Uptime Rules
JVM CPU Usage Statistics Rules
JVM Garbage Collector Statistics Rules
JVM Heap Memory Statistics Rules
JVM Non-Heap Memory Statistics Rules
JVM Metaspace/Permanent Space Statistics Rules
JVM Metaspace/Permanent Space Usage Rules
JVM Metaspace/Permanent Space Initial Size Rules
JVM Metaspace/Permanent Space Committed Size Rules
JVM_PCS
JVM_PCS_ACCEPTABLE_OPERATION
JVM_PCS_APPROACHING_LIMIT
JVM_PCS_AT_LIMIT
JVM_PCS_EXCEEDING_LIMIT
JVM_PCS_MAJOR_OVERLOAD
JVM Metaspace/Permanent Space Maximum Size Rules
JVM Metaspace/Permanent Space Peak Usage Rules
JVM Metaspace/Permanent Space Peak Maximum Size F
JVM Compressed Class Space Statistics Rules
JVM Code Cache Statistics Rules
JVM Overall Non-Heap Memory Space Utilization Rules
JVM Class Loader Statistics Rules
JVM Thread Statistics Rules
JVM Just-In-Time Compiler Statistics Rules
Used In Which Relators?
Update Rule
%VARIABLES%
Copyright © 2019 Argent Software. All Rights Reserved.
Current Time Zone: IST
Click To Change
v5.1A-R8

JVM Metaspace/Permanent Space Committed Size Rule Definition: JVM_PCS_AT_LIMIT

Rule Is Broken If The Amount Of Metaspace/Permanent Memory That Is Guaranteed To Be Available For Use By The JVM Is Greater Than 4GB

Console Comment: JVM: Excessive Amount Of Metaspace/Permanent Memory That Is Guaranteed To Be Available For Use By The JVM - Over 4GB

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

□ Post Event Even If Same Event Is Still Outstanding (Unanswered)

□ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

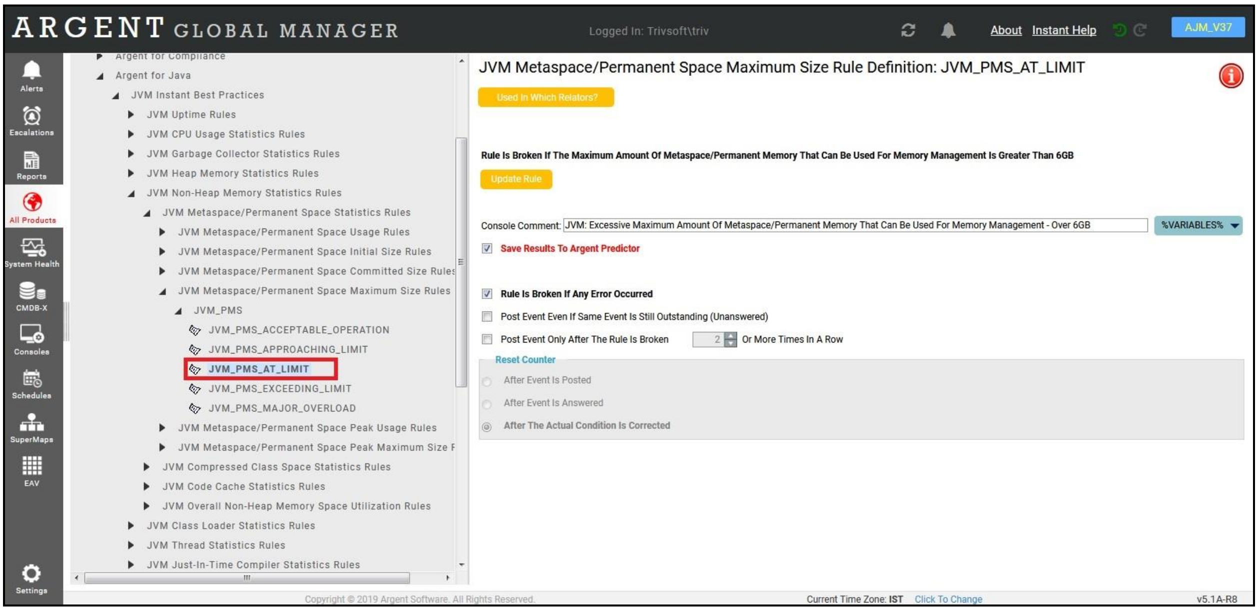
○ After Event Is Posted

○ After Event Is Answered

● After The Actual Condition Is Corrected

Permanent Space Maximum Size

This rule monitors the maximum allowed size of permanent space in a JVM.


ARGENT GLOBAL MANAGER
Logged In: Trivsoft\triv
About
Instant Help
AJM_V37
Alerts
Escalations
Reports
All Products
System Health
CMDB-X
Consoles
Schedules
SuperMaps
EAV
Settings
Argent for Compliance
Argent for Java
JVM Instant Best Practices
JVM Uptime Rules
JVM CPU Usage Statistics Rules
JVM Garbage Collector Statistics Rules
JVM Heap Memory Statistics Rules
JVM Non-Heap Memory Statistics Rules
JVM Metaspace/Permanent Space Statistics Rules
JVM Metaspace/Permanent Space Usage Rules
JVM Metaspace/Permanent Space Initial Size Rules
JVM Metaspace/Permanent Space Committed Size Rules
JVM Metaspace/Permanent Space Maximum Size Rules
JVM_PMS
JVM_PMS_ACCEPTABLE_OPERATION
JVM_PMS_APPROACHING_LIMIT
JVM_PMS_AT_LIMIT
JVM_PMS_EXCEEDING_LIMIT
JVM_PMS_MAJOR_OVERLOAD
JVM Metaspace/Permanent Space Peak Usage Rules
JVM Metaspace/Permanent Space Peak Maximum Size F
JVM Compressed Class Space Statistics Rules
JVM Code Cache Statistics Rules
JVM Overall Non-Heap Memory Space Utilization Rules
JVM Class Loader Statistics Rules
JVM Thread Statistics Rules
JVM Just-In-Time Compiler Statistics Rules
Used In Which Relators?
Update Rule
%VARIABLES%

JVM Metaspace/Permanent Space Maximum Size Rule Definition: JVM_PMS_AT_LIMIT

Rule Is Broken If The Maximum Amount Of Metaspace/Permanent Memory That Can Be Used For Memory Management Is Greater Than 6GB

Console Comment: JVM: Excessive Maximum Amount Of Metaspace/Permanent Memory That Can Be Used For Memory Management - Over 6GB

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

□ Post Event Even If Same Event Is Still Outstanding (Unanswered)

□ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

○ After Event Is Posted

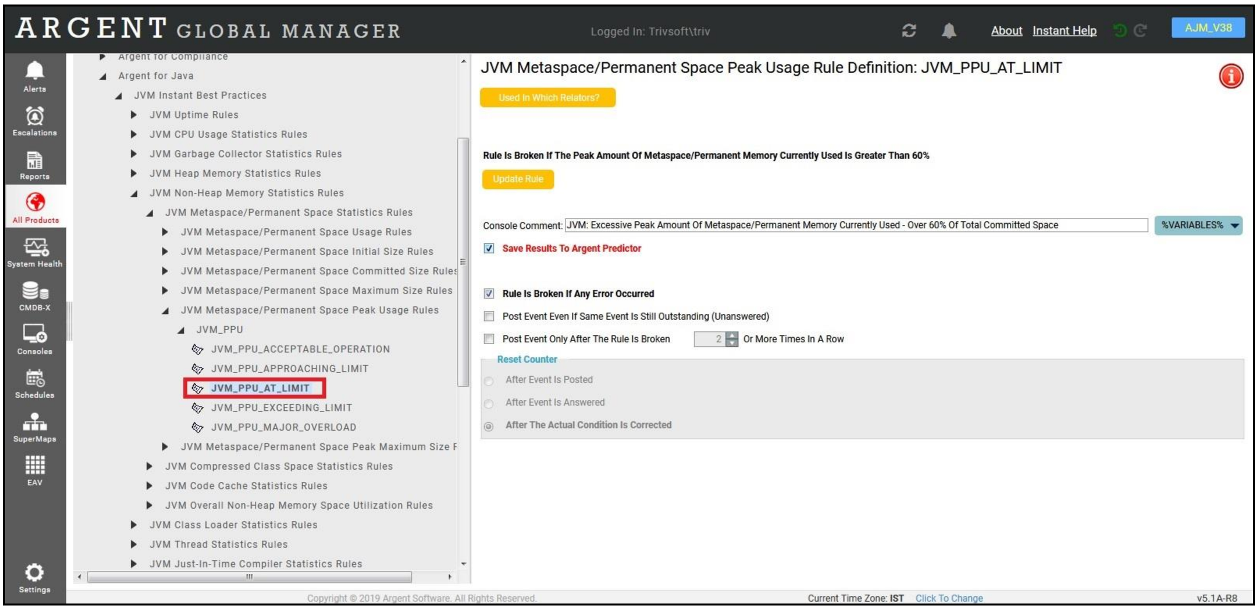
○ After Event Is Answered

● After The Actual Condition Is Corrected

Copyright © 2019 Argent Software. All Rights Reserved.   Current Time Zone: IST Click To Change   v5.1A-R8

Permanent Space Peak Usage

This rule monitors peak permanent space usage in use at any given time


ARGENT GLOBAL MANAGER
Logged In: Trivsoft\triv
About
Instant Help
AJM_V38
Alerts
Escalations
Reports
All Products
System Health
CMDB-X
Consoles
Schedules
SuperMaps
EAV
Settings
Argent for Compliance
Argent for Java
JVM Instant Best Practices
JVM Uptime Rules
JVM CPU Usage Statistics Rules
JVM Garbage Collector Statistics Rules
JVM Heap Memory Statistics Rules
JVM Non-Heap Memory Statistics Rules
JVM Metaspace/Permanent Space Statistics Rules
JVM Metaspace/Permanent Space Usage Rules
JVM Metaspace/Permanent Space Initial Size Rules
JVM Metaspace/Permanent Space Committed Size Rules
JVM Metaspace/Permanent Space Maximum Size Rules
JVM Metaspace/Permanent Space Peak Usage Rules
JVM_PPU
JVM_PPU_ACCEPTABLE_OPERATION
JVM_PPU_APPROACHING_LIMIT
JVM_PPU_AT_LIMIT
JVM_PPU_EXCEEDING_LIMIT
JVM_PPU_MAJOR_OVERLOAD
JVM Metaspace/Permanent Space Peak Maximum Size F
JVM Compressed Class Space Statistics Rules
JVM Code Cache Statistics Rules
JVM Overall Non-Heap Memory Space Utilization Rules
JVM Class Loader Statistics Rules
JVM Thread Statistics Rules
JVM Just-In-Time Compiler Statistics Rules
Used In Which Relators?
Update Rule
%VARIABLES%
Copyright © 2019 Argent Software. All Rights Reserved.
Current Time Zone: IST
Click To Change
v5.1A-R8

JVM Metaspace/Permanent Space Peak Usage Rule Definition: JVM_PPU_AT_LIMIT

Rule Is Broken If The Peak Amount Of Metaspace/Permanent Memory Currently Used Is Greater Than 60%

Console Comment: JVM: Excessive Peak Amount Of Metaspace/Permanent Memory Currently Used - Over 60% Of Total Committed Space

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

□ Post Event Even If Same Event Is Still Outstanding (Unanswered)

□ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

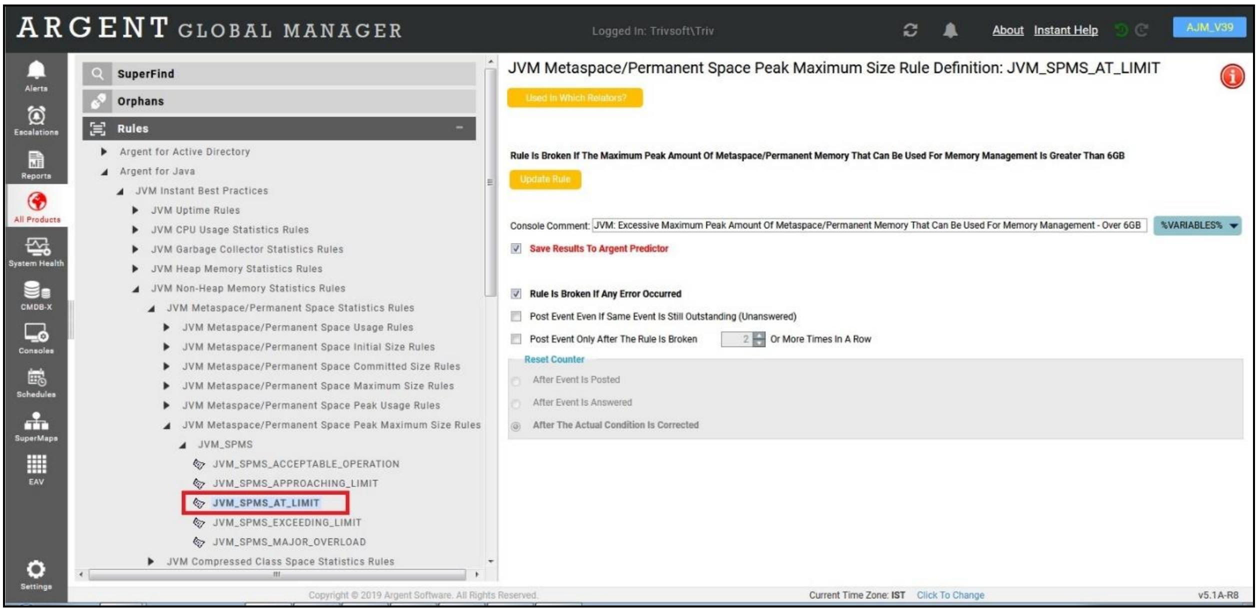
○ After Event Is Posted

○ After Event Is Answered

● After The Actual Condition Is Corrected

Permanent Space Peak Maximum Size

This rule monitors the maximum peak amount of metaspace/permanent memory that can be used for memory management


ARGENT GLOBAL MANAGER
Logged In: Trivsoft\Triv
About
Instant Help
AJM_V39
Alerts
Escalations
Reports
All Products
System Health
CMDB-X
Consoles
Schedules
SuperMaps
EAV
Settings
SuperFind
Orphans
Rules
Argent for Active Directory
Argent for Java
JVM Instant Best Practices
JVM Uptime Rules
JVM CPU Usage Statistics Rules
JVM Garbage Collector Statistics Rules
JVM Heap Memory Statistics Rules
JVM Non-Heap Memory Statistics Rules
JVM Metaspace/Permanent Space Statistics Rules
JVM Metaspace/Permanent Space Usage Rules
JVM Metaspace/Permanent Space Initial Size Rules
JVM Metaspace/Permanent Space Committed Size Rules
JVM Metaspace/Permanent Space Maximum Size Rules
JVM Metaspace/Permanent Space Peak Usage Rules
JVM Metaspace/Permanent Space Peak Maximum Size Rules
JVM_SPMS
JVM_SPMS_ACCEPTABLE_OPERATION
JVM_SPMS_APPROACHING_LIMIT
JVM_SPMS_AT_LIMIT
JVM_SPMS_EXCEEDING_LIMIT
JVM_SPMS_MAJOR_OVERLOAD
JVM Compressed Class Space Statistics Rules
Used In Which Relators?
Update Rule
%VARIABLES%
Copyright © 2019 Argent Software. All Rights Reserved.
Current Time Zone: IST
Click To Change
v5.1A-R8

JVM Metaspace/Permanent Space Peak Maximum Size Rule Definition: JVM_SPMS_AT_LIMIT

Rule Is Broken If The Maximum Peak Amount Of Metaspace/Permanent Memory That Can Be Used For Memory Management Is Greater Than 6GB

Console Comment: JVM: Excessive Maximum Peak Amount Of Metaspace/Permanent Memory That Can Be Used For Memory Management - Over 6GB

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

☐ Post Event Even If Same Event Is Still Outstanding (Unanswered)

☐ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

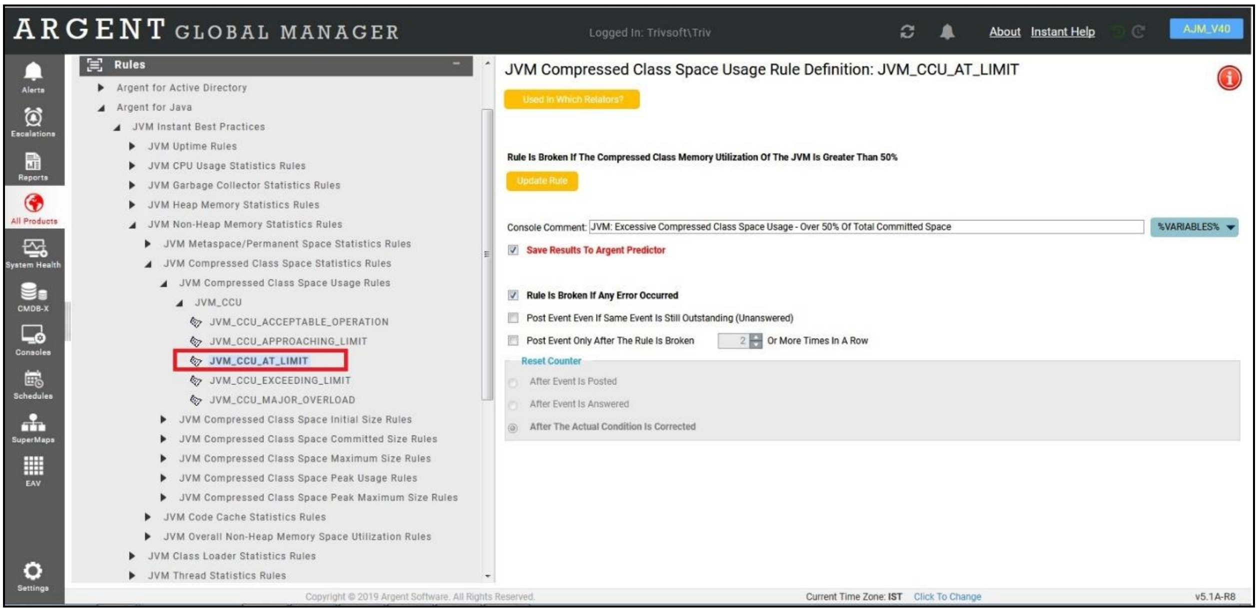
○ After Event Is Posted

○ After Event Is Answered

● After The Actual Condition Is Corrected

Compressed Class Space Usage

This rule monitors Java Virtual Machine's Compressed Class space utilization.This statistics is only available in JVM running in 64 bit version of JDK 8 or above


ARGENT GLOBAL MANAGER
Logged In: Trivsoft\Triv
About
Instant Help
AJM_V40
Alerts
Escalations
Reports
All Products
System Health
CMDB-X
Consoles
Schedules
SuperMaps
EAV
Settings
Rules
Argent for Active Directory
Argent for Java
JVM Instant Best Practices
JVM Uptime Rules
JVM CPU Usage Statistics Rules
JVM Garbage Collector Statistics Rules
JVM Heap Memory Statistics Rules
JVM Non-Heap Memory Statistics Rules
JVM Metaspace/Permanent Space Statistics Rules
JVM Compressed Class Space Statistics Rules
JVM Compressed Class Space Usage Rules
JVM_CCU
JVM_CCU_ACCEPTABLE_OPERATION
JVM_CCU_APPROACHING_LIMIT
JVM_CCU_AT_LIMIT
JVM_CCU_EXCEEDING_LIMIT
JVM_CCU_MAJOR_OVERLOAD
JVM Compressed Class Space Initial Size Rules
JVM Compressed Class Space Committed Size Rules
JVM Compressed Class Space Maximum Size Rules
JVM Compressed Class Space Peak Usage Rules
JVM Compressed Class Space Peak Maximum Size Rules
JVM Code Cache Statistics Rules
JVM Overall Non-Heap Memory Space Utilization Rules
JVM Class Loader Statistics Rules
JVM Thread Statistics Rules
Used In Which Relators?
Update Rule
%VARIABLES%
Current Time Zone: IST
Click To Change
v5.1A-R8

JVM Compressed Class Space Usage Rule Definition: JVM_CCU_AT_LIMIT

Rule Is Broken If The Compressed Class Memory Utilization Of The JVM Is Greater Than 50%

Console Comment: JVM: Excessive Compressed Class Space Usage - Over 50% Of Total Committed Space

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

□ Post Event Even If Same Event Is Still Outstanding (Unanswered)

□ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

○ After Event Is Posted

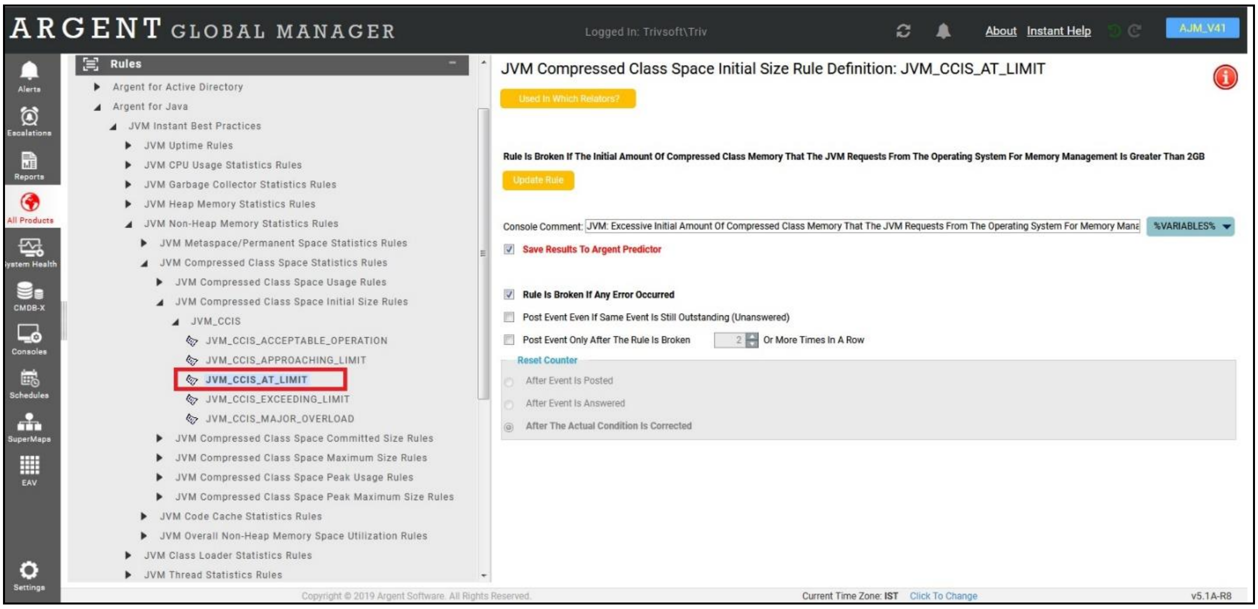
○ After Event Is Answered

● After The Actual Condition Is Corrected

Copyright © 2019 Argent Software. All Rights Reserved.

Compressed Class Space Initial Size

This rule monitors the initial amount of compressed class memory that the jvm requests from the operating system for memory management.This statistics is only available in JVM running in 64 bit version of JDK 8 or above


ARGENT GLOBAL MANAGER
Logged In: Trivsoft\Triv
About
Instant Help
AJM_V41
Alerts
Escalations
Reports
All Products
System Health
CMDB-X
Consoles
Schedules
SuperMaps
EAV
Settings
Rules
Argent for Active Directory
Argent for Java
JVM Instant Best Practices
JVM Uptime Rules
JVM CPU Usage Statistics Rules
JVM Garbage Collector Statistics Rules
JVM Heap Memory Statistics Rules
JVM Non-Heap Memory Statistics Rules
JVM Metaspace/Permanent Space Statistics Rules
JVM Compressed Class Space Statistics Rules
JVM Compressed Class Space Usage Rules
JVM Compressed Class Space Initial Size Rules
JVM_CCIS
JVM_CCIS_ACCEPTABLE_OPERATION
JVM_CCIS_APPROACHING_LIMIT
JVM_CCIS_AT_LIMIT
JVM_CCIS_EXCEEDING_LIMIT
JVM_CCIS_MAJOR_OVERLOAD
JVM Compressed Class Space Committed Size Rules
JVM Compressed Class Space Maximum Size Rules
JVM Compressed Class Space Peak Usage Rules
JVM Compressed Class Space Peak Maximum Size Rules
JVM Code Cache Statistics Rules
JVM Overall Non-Heap Memory Space Utilization Rules
JVM Class Loader Statistics Rules
JVM Thread Statistics Rules
Used In Which Relators?
Update Rule
%VARIABLES%

JVM Compressed Class Space Initial Size Rule Definition: JVM_CCIS_AT_LIMIT

Rule Is Broken If The Initial Amount Of Compressed Class Memory That The JVM Requests From The Operating System For Memory Management Is Greater Than 2GB

Console Comment: JVM: Excessive Initial Amount Of Compressed Class Memory That The JVM Requests From The Operating System For Memory Mana

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

□ Post Event Even If Same Event Is Still Outstanding (Unanswered)

□ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

○ After Event Is Posted

○ After Event Is Answered

● After The Actual Condition Is Corrected

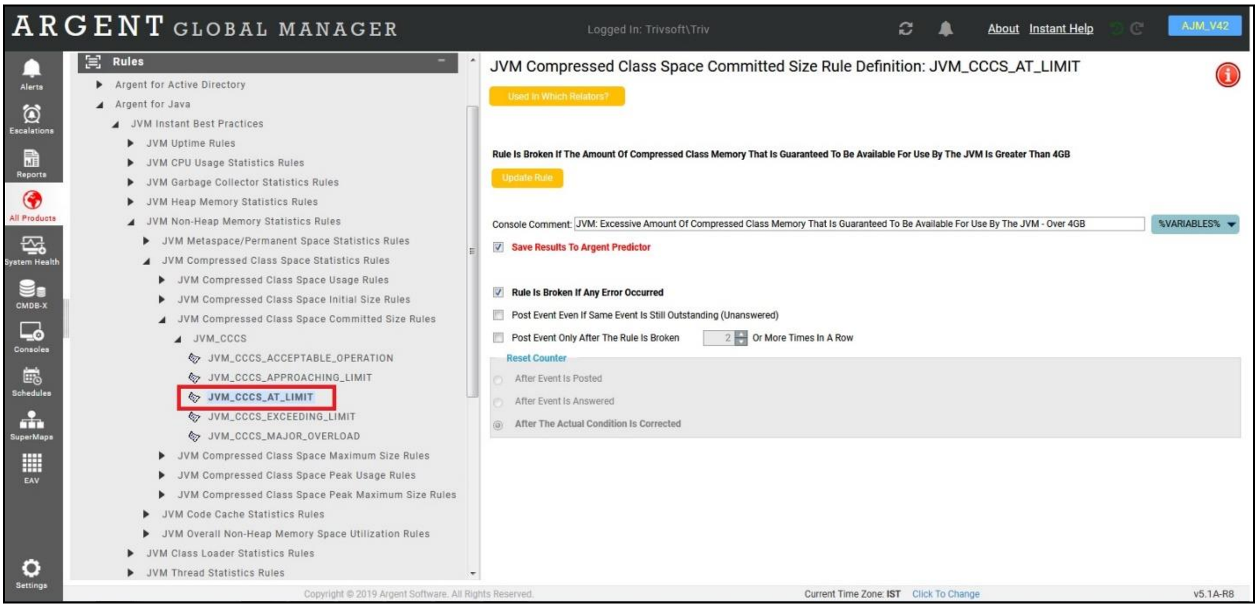
Copyright © 2019 Argent Software. All Rights Reserved.

Current Time Zone: IST Click To Change

v5.1A-R8

Compressed Class Space Committed Size

This rule monitors the amount of compressed class memory that is guaranteed to be available for use by the JVM.This statistics is only available in JVM running in 64 bit version of JDK 8 or above



- Argent for Active Directory
- Argent for Java
  - JVM Instant Best Practices
    - JVM Uptime Rules
    - JVM CPU Usage Statistics Rules
    - JVM Garbage Collector Statistics Rules
    - JVM Heap Memory Statistics Rules
    - JVM Non-Heap Memory Statistics Rules
      - JVM Metaspace/Permanent Space Statistics Rules
      - JVM Compressed Class Space Statistics Rules
        - JVM Compressed Class Space Usage Rules
        - JVM Compressed Class Space Initial Size Rules
        - JVM Compressed Class Space Committed Size Rules
          - JVM_CCCS
            - JVM_CCCS_ACCEPTABLE_OPERATION
            - JVM_CCCS_APPROACHING_LIMIT
            - JVM_CCCS_AT_LIMIT
            - JVM_CCCS_EXCEEDING_LIMIT
            - JVM_CCCS_MAJOR_OVERLOAD
        - JVM Compressed Class Space Maximum Size Rules
        - JVM Compressed Class Space Peak Usage Rules
        - JVM Compressed Class Space Peak Maximum Size Rules
      - JVM Code Cache Statistics Rules
      - JVM Overall Non-Heap Memory Space Utilization Rules
    - JVM Class Loader Statistics Rules
    - JVM Thread Statistics Rules

JVM Compressed Class Space Committed Size Rule Definition: JVM_CCCS_AT_LIMIT

Rule Is Broken If The Amount Of Compressed Class Memory That Is Guaranteed To Be Available For Use By The JVM Is Greater Than 4GB

Console Comment: JVM: Excessive Amount Of Compressed Class Memory That Is Guaranteed To Be Available For Use By The JVM - Over 4GB

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

☐ Post Event Even If Same Event Is Still Outstanding (Unanswered)

☐ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

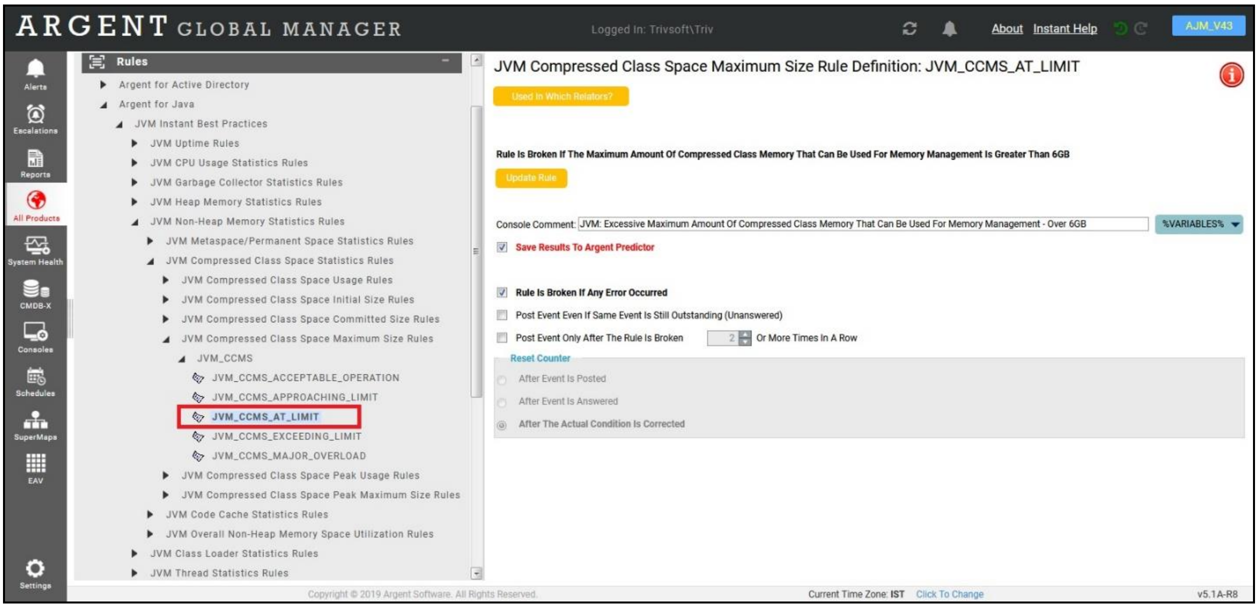
○ After Event Is Posted

○ After Event Is Answered

● After The Actual Condition Is Corrected

Compressed Class Space Maximum Size

This rule monitors the maximum amount of compressed class memory that can be used for memory management.This statistics is only available in JVM running in 64 bit version of JDK 8 or above



ARGENT GLOBAL MANAGER
Logged In: Trivsoft\Triv
About
Instant Help
AJM_V43
Alerts
Escalations
Reports
All Products
System Health
CMDB-X
Consoles
Schedules
SuperMaps
EAV
Settings
Rules
Argent for Active Directory
Argent for Java
JVM Instant Best Practices
JVM Uptime Rules
JVM CPU Usage Statistics Rules
JVM Garbage Collector Statistics Rules
JVM Heap Memory Statistics Rules
JVM Non-Heap Memory Statistics Rules
JVM Metaspace/Permanent Space Statistics Rules
JVM Compressed Class Space Statistics Rules
JVM Compressed Class Space Usage Rules
JVM Compressed Class Space Initial Size Rules
JVM Compressed Class Space Committed Size Rules
JVM Compressed Class Space Maximum Size Rules
JVM_CCMS
JVM_CCMS_ACCEPTABLE_OPERATION
JVM_CCMS_APPROACHING_LIMIT
JVM_CCMS_AT_LIMIT
JVM_CCMS_EXCEEDING_LIMIT
JVM_CCMS_MAJOR_OVERLOAD
JVM Compressed Class Space Peak Usage Rules
JVM Compressed Class Space Peak Maximum Size Rules
JVM Code Cache Statistics Rules
JVM Overall Non-Heap Memory Space Utilization Rules
JVM Class Loader Statistics Rules
JVM Thread Statistics Rules
Used In Which Relators?
Update Rule
%VARIABLES%
Copyright © 2019 Argent Software. All Rights Reserved.
Current Time Zone: IST
Click To Change
v5.1A-R8

JVM Compressed Class Space Maximum Size Rule Definition: JVM_CCMS_AT_LIMIT

Rule Is Broken If The Maximum Amount Of Compressed Class Memory That Can Be Used For Memory Management Is Greater Than 6GB

Console Comment: JVM: Excessive Maximum Amount Of Compressed Class Memory That Can Be Used For Memory Management - Over 6GB

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

□ Post Event Even If Same Event Is Still Outstanding (Unanswered)

□ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

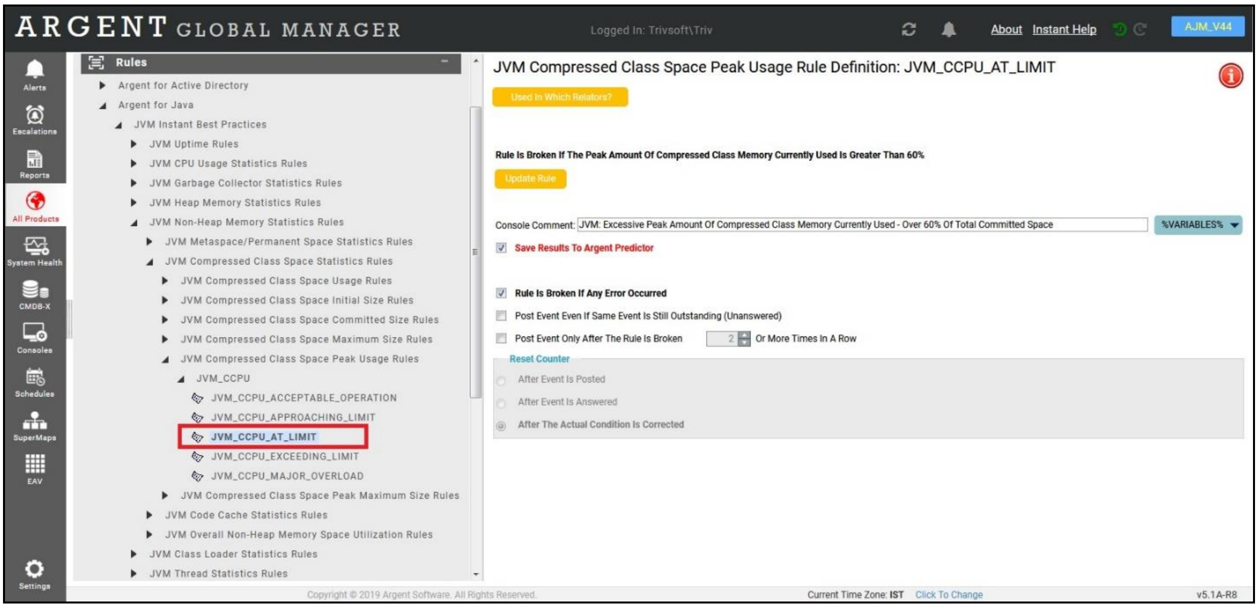
○ After Event Is Posted

○ After Event Is Answered

● After The Actual Condition Is Corrected

Compressed Class Space Peak Usage

This rule monitors the peak amount of compressed class memory currently used.This statistics is only available in JVM running in 64 bit version of JDK 8 or above



ARGENT GLOBAL MANAGER
Logged In: Trivsoft\Triv
About
Instant Help
AJM_V44
Alerts
Escalations
Reports
All Products
System Health
CMDB-X
Consoles
Schedules
SuperMaps
EAV
Settings
Rules
Argent for Active Directory
Argent for Java
JVM Instant Best Practices
JVM Uptime Rules
JVM CPU Usage Statistics Rules
JVM Garbage Collector Statistics Rules
JVM Heap Memory Statistics Rules
JVM Non-Heap Memory Statistics Rules
JVM Metaspace/Permanent Space Statistics Rules
JVM Compressed Class Space Statistics Rules
JVM Compressed Class Space Usage Rules
JVM Compressed Class Space Initial Size Rules
JVM Compressed Class Space Committed Size Rules
JVM Compressed Class Space Maximum Size Rules
JVM Compressed Class Space Peak Usage Rules
JVM_CCPU
JVM_CCPU_ACCEPTABLE_OPERATION
JVM_CCPU_APPROACHING_LIMIT
JVM_CCPU_AT_LIMIT
JVM_CCPU_EXCEEDING_LIMIT
JVM_CCPU_MAJOR_OVERLOAD
JVM Compressed Class Space Peak Maximum Size Rules
JVM Code Cache Statistics Rules
JVM Overall Non-Heap Memory Space Utilization Rules
JVM Class Loader Statistics Rules
JVM Thread Statistics Rules
Used In Which Relators?
Update Rule
%VARIABLES%
Copyright © 2019 Argent Software. All Rights Reserved.
Current Time Zone: IST
Click To Change
v5.1A-R8

JVM Compressed Class Space Peak Usage Rule Definition: JVM_CCPU_AT_LIMIT

Rule Is Broken If The Peak Amount Of Compressed Class Memory Currently Used Is Greater Than 60%

Console Comment: JVM: Excessive Peak Amount Of Compressed Class Memory Currently Used - Over 60% Of Total Committed Space

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

☐ Post Event Even If Same Event Is Still Outstanding (Unanswered)

☐ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

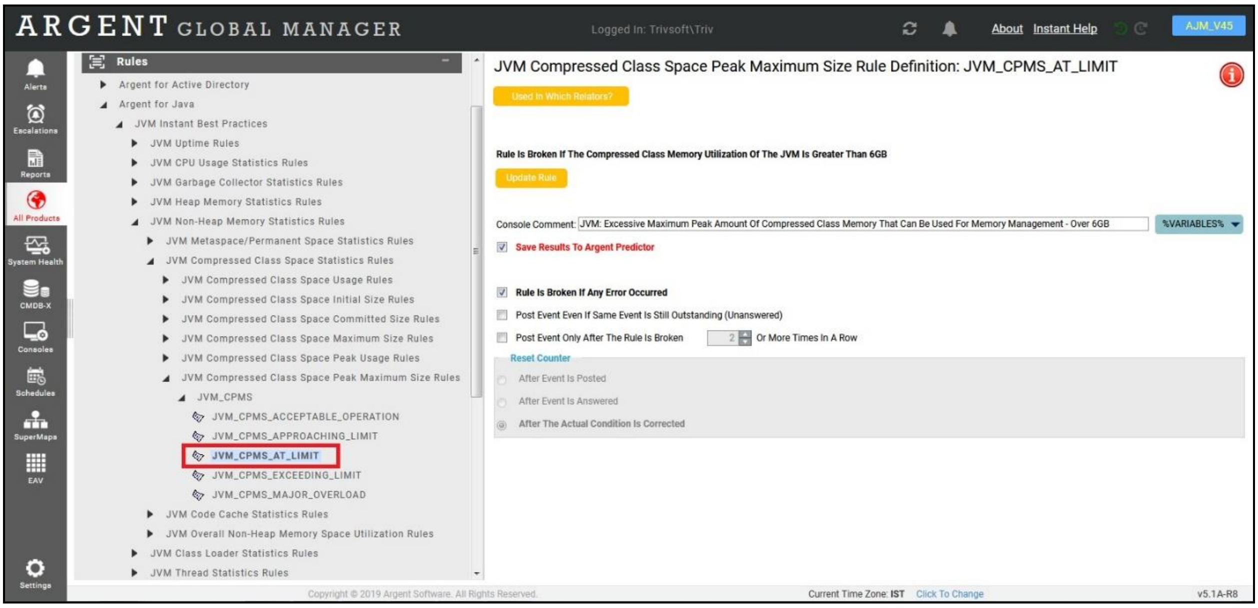
○ After Event Is Posted

○ After Event Is Answered

● After The Actual Condition Is Corrected

Compressed Class Space Peak Maximum Size

This rule monitors the maximum peak amount of compressed class memory that can be used for memory management.This statistics is only available in JVM running in 64 bit version of JDK 8 or above


ARGENT GLOBAL MANAGER
Logged In: Trivsoft\Triv
About
Instant Help
AJM_V45
Alerts
Escalations
Reports
All Products
System Health
CMDB-X
Consoles
Schedules
SuperMaps
EAV
Settings
Rules
Argent for Active Directory
Argent for Java
JVM Instant Best Practices
JVM Uptime Rules
JVM CPU Usage Statistics Rules
JVM Garbage Collector Statistics Rules
JVM Heap Memory Statistics Rules
JVM Non-Heap Memory Statistics Rules
JVM Metaspace/Permanent Space Statistics Rules
JVM Compressed Class Space Statistics Rules
JVM Compressed Class Space Usage Rules
JVM Compressed Class Space Initial Size Rules
JVM Compressed Class Space Committed Size Rules
JVM Compressed Class Space Maximum Size Rules
JVM Compressed Class Space Peak Usage Rules
JVM Compressed Class Space Peak Maximum Size Rules
JVM_CPMS
JVM_CPMS_ACCEPTABLE_OPERATION
JVM_CPMS_APPROACHING_LIMIT
JVM_CPMS_AT_LIMIT
JVM_CPMS_EXCEEDING_LIMIT
JVM_CPMS_MAJOR_OVERLOAD
JVM Code Cache Statistics Rules
JVM Overall Non-Heap Memory Space Utilization Rules
JVM Class Loader Statistics Rules
JVM Thread Statistics Rules
Used In Which Relators?
Update Rule
%VARIABLES%
Copyright © 2019 Argent Software. All Rights Reserved.
Current Time Zone: IST
Click To Change
v5.1A-R8

JVM Compressed Class Space Peak Maximum Size Rule Definition: JVM_CPMS_AT_LIMIT

Rule Is Broken If The Compressed Class Memory Utilization Of The JVM Is Greater Than 6GB

Console Comment: JVM: Excessive Maximum Peak Amount Of Compressed Class Memory That Can Be Used For Memory Management - Over 6GB

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

□ Post Event Even If Same Event Is Still Outstanding (Unanswered)

□ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

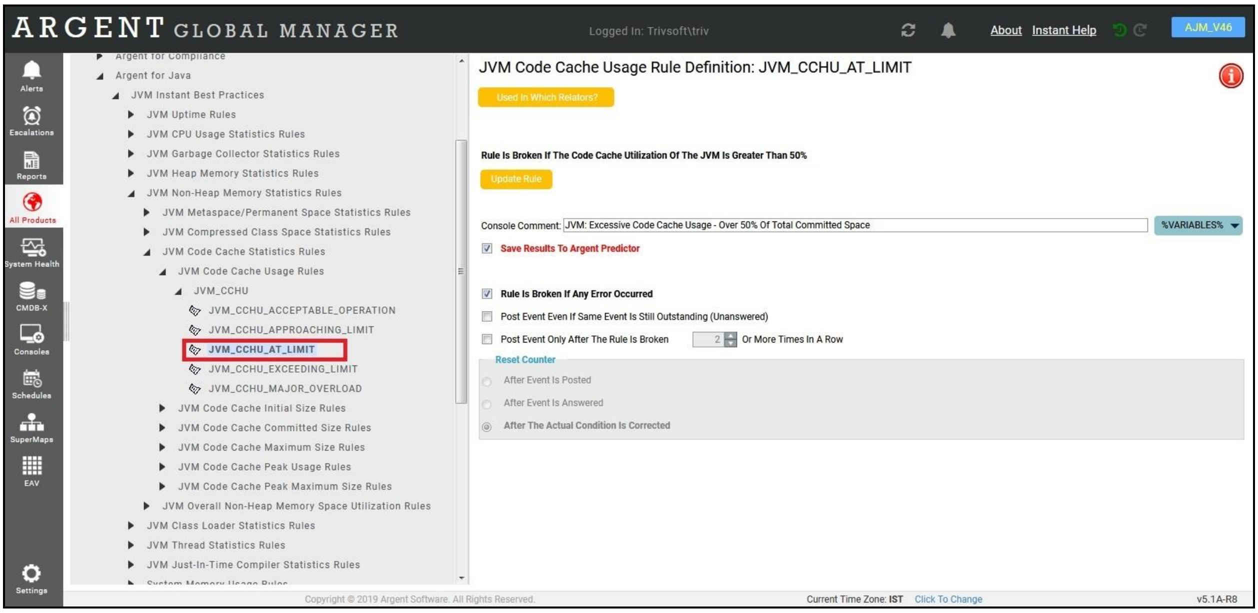
○ After Event Is Posted

○ After Event Is Answered

● After The Actual Condition Is Corrected

Code Cache Utilization

Memory used for compiling and storing native Java code is stored in an area of memory called the code cache.   Argent for Java monitors the code cache.


ARGENT GLOBAL MANAGER
Logged In: Trivsoft\triv
About
Instant Help
AJM_V46
Alerts
Escalations
Reports
All Products
System Health
CMDB-X
Consoles
Schedules
SuperMaps
EAV
Settings
Argent for Compliance
Argent for Java
JVM Instant Best Practices
JVM Uptime Rules
JVM CPU Usage Statistics Rules
JVM Garbage Collector Statistics Rules
JVM Heap Memory Statistics Rules
JVM Non-Heap Memory Statistics Rules
JVM Metaspace/Permanent Space Statistics Rules
JVM Compressed Class Space Statistics Rules
JVM Code Cache Statistics Rules
JVM Code Cache Usage Rules
JVM_CCHU
JVM_CCHU_ACCEPTABLE_OPERATION
JVM_CCHU_APPROACHING_LIMIT
JVM_CCHU_AT_LIMIT
JVM_CCHU_EXCEEDING_LIMIT
JVM_CCHU_MAJOR_OVERLOAD
JVM Code Cache Initial Size Rules
JVM Code Cache Committed Size Rules
JVM Code Cache Maximum Size Rules
JVM Code Cache Peak Usage Rules
JVM Code Cache Peak Maximum Size Rules
JVM Overall Non-Heap Memory Space Utilization Rules
JVM Class Loader Statistics Rules
JVM Thread Statistics Rules
JVM Just-In-Time Compiler Statistics Rules
System Memory Usage Rules
Used In Which Relators?
Update Rule
%VARIABLES%
Copyright © 2019 Argent Software. All Rights Reserved.
Current Time Zone: IST
Click To Change
v5.1A-R8

JVM Code Cache Usage Rule Definition: JVM_CCHU_AT_LIMIT

Rule Is Broken If The Code Cache Utilization Of The JVM Is Greater Than 50%

Console Comment: JVM: Excessive Code Cache Usage - Over 50% Of Total Committed Space

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

☐ Post Event Even If Same Event Is Still Outstanding (Unanswered)

☐ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

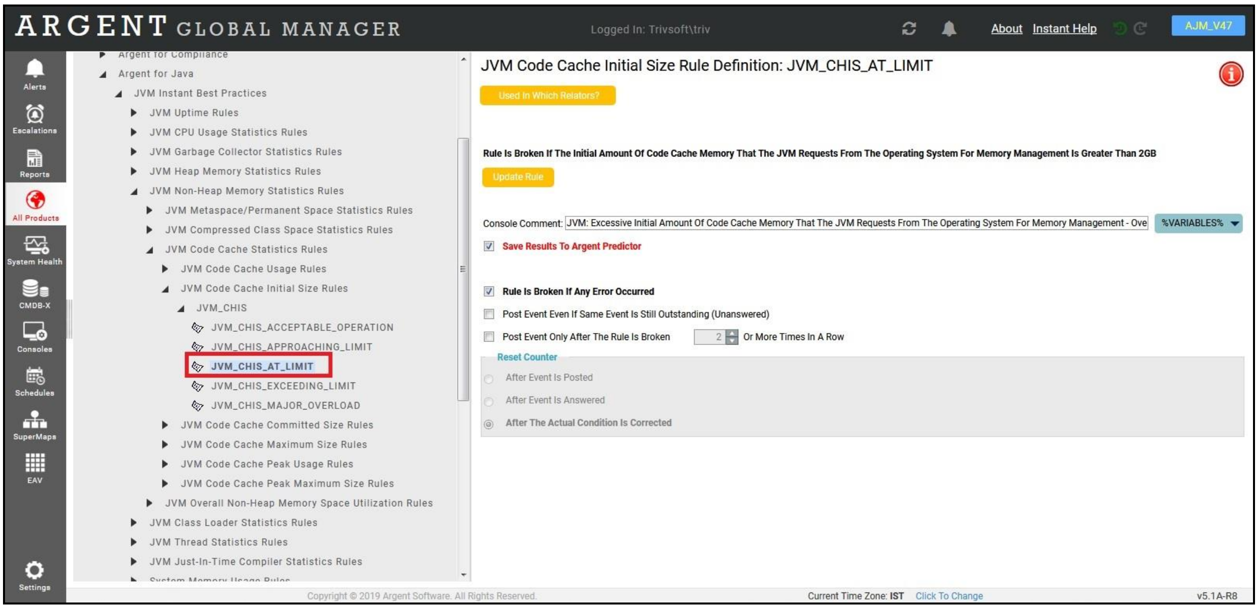
○ After Event Is Posted

○ After Event Is Answered

● After The Actual Condition Is Corrected

Code Cache Initial Size

This rule monitors initial size of memory requested for the code cache


ARGENT GLOBAL MANAGER
Logged In: Trivsoft\triv
About
Instant Help
AJM_V47
Alerts
Escalations
Reports
All Products
System Health
CMDB-X
Consoles
Schedules
SuperMaps
EAV
Settings
Argent for Compliance
Argent for Java
JVM Instant Best Practices
JVM Uptime Rules
JVM CPU Usage Statistics Rules
JVM Garbage Collector Statistics Rules
JVM Heap Memory Statistics Rules
JVM Non-Heap Memory Statistics Rules
JVM Metaspace/Permanent Space Statistics Rules
JVM Compressed Class Space Statistics Rules
JVM Code Cache Statistics Rules
JVM Code Cache Usage Rules
JVM Code Cache Initial Size Rules
JVM_CHIS
JVM_CHIS_ACCEPTABLE_OPERATION
JVM_CHIS_APPROACHING_LIMIT
JVM_CHIS_AT_LIMIT
JVM_CHIS_EXCEEDING_LIMIT
JVM_CHIS_MAJOR_OVERLOAD
JVM Code Cache Committed Size Rules
JVM Code Cache Maximum Size Rules
JVM Code Cache Peak Usage Rules
JVM Code Cache Peak Maximum Size Rules
JVM Overall Non-Heap Memory Space Utilization Rules
JVM Class Loader Statistics Rules
JVM Thread Statistics Rules
JVM Just-In-Time Compiler Statistics Rules
System Memory Usage Rules
Used In Which Relators?
Update Rule
%VARIABLES%
Copyright © 2019 Argent Software. All Rights Reserved.
Current Time Zone: IST
Click To Change
v5.1A-R8

JVM Code Cache Initial Size Rule Definition: JVM_CHIS_AT_LIMIT

Rule Is Broken If The Initial Amount Of Code Cache Memory That The JVM Requests From The Operating System For Memory Management Is Greater Than 2GB

Console Comment: JVM: Excessive Initial Amount Of Code Cache Memory That The JVM Requests From The Operating System For Memory Management - Ove

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

☐ Post Event Even If Same Event Is Still Outstanding (Unanswered)

☐ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

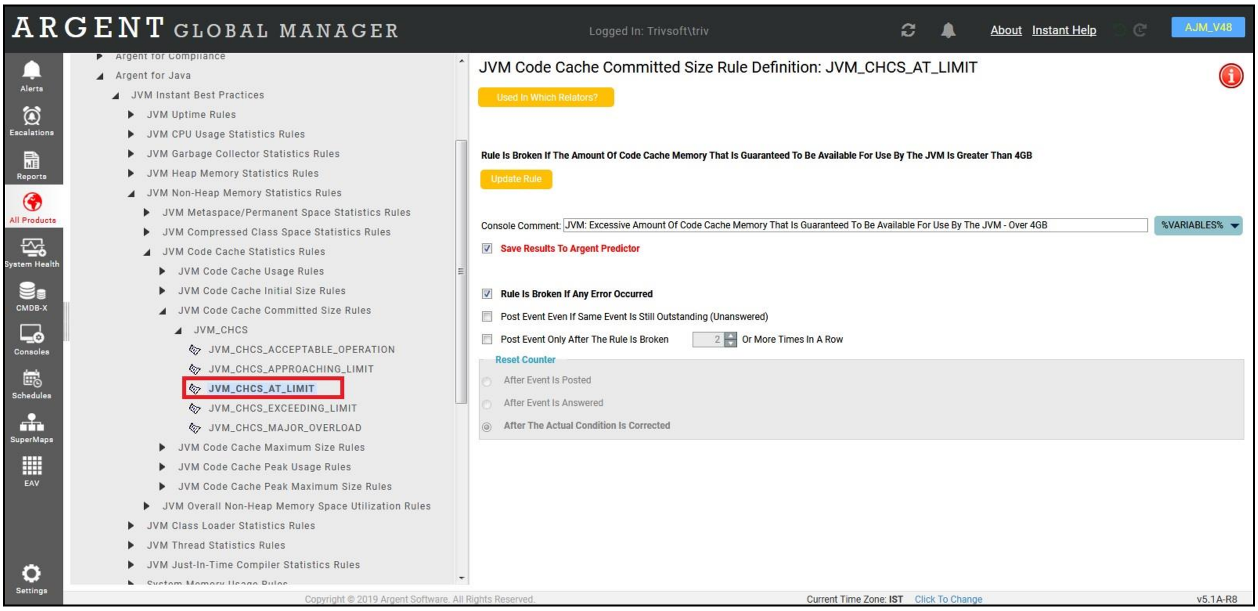
○ After Event Is Posted

○ After Event Is Answered

● After The Actual Condition Is Corrected

Code Cache Committed Size

This rule monitors the committed code cache memory available in a JVM


ARGENT GLOBAL MANAGER
Logged In: Trivsoft\triv
About
Instant Help
AJM_V48
Alerts
Escalations
Reports
All Products
System Health
CMDB-X
Consoles
Schedules
SuperMaps
EAV
Settings
Argent for Compliance
Argent for Java
JVM Instant Best Practices
JVM Uptime Rules
JVM CPU Usage Statistics Rules
JVM Garbage Collector Statistics Rules
JVM Heap Memory Statistics Rules
JVM Non-Heap Memory Statistics Rules
JVM Metaspace/Permanent Space Statistics Rules
JVM Compressed Class Space Statistics Rules
JVM Code Cache Statistics Rules
JVM Code Cache Usage Rules
JVM Code Cache Initial Size Rules
JVM Code Cache Committed Size Rules
JVM_CHCS
JVM_CHCS_ACCEPTABLE_OPERATION
JVM_CHCS_APPROACHING_LIMIT
JVM_CHCS_AT_LIMIT
JVM_CHCS_EXCEEDING_LIMIT
JVM_CHCS_MAJOR_OVERLOAD
JVM Code Cache Maximum Size Rules
JVM Code Cache Peak Usage Rules
JVM Code Cache Peak Maximum Size Rules
JVM Overall Non-Heap Memory Space Utilization Rules
JVM Class Loader Statistics Rules
JVM Thread Statistics Rules
JVM Just-In-Time Compiler Statistics Rules
System Memory Usage Rules
Used In Which Relators?
Update Rule
%VARIABLES%

JVM Code Cache Committed Size Rule Definition: JVM_CHCS_AT_LIMIT

Rule Is Broken If The Amount Of Code Cache Memory That Is Guaranteed To Be Available For Use By The JVM Is Greater Than 4GB

Console Comment: JVM: Excessive Amount Of Code Cache Memory That Is Guaranteed To Be Available For Use By The JVM - Over 4GB

☑ **Save Results To Argent Predictor**

☑ **Rule Is Broken If Any Error Occurred**

☐ Post Event Even If Same Event Is Still Outstanding (Unanswered)

☐ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

○ After Event Is Posted

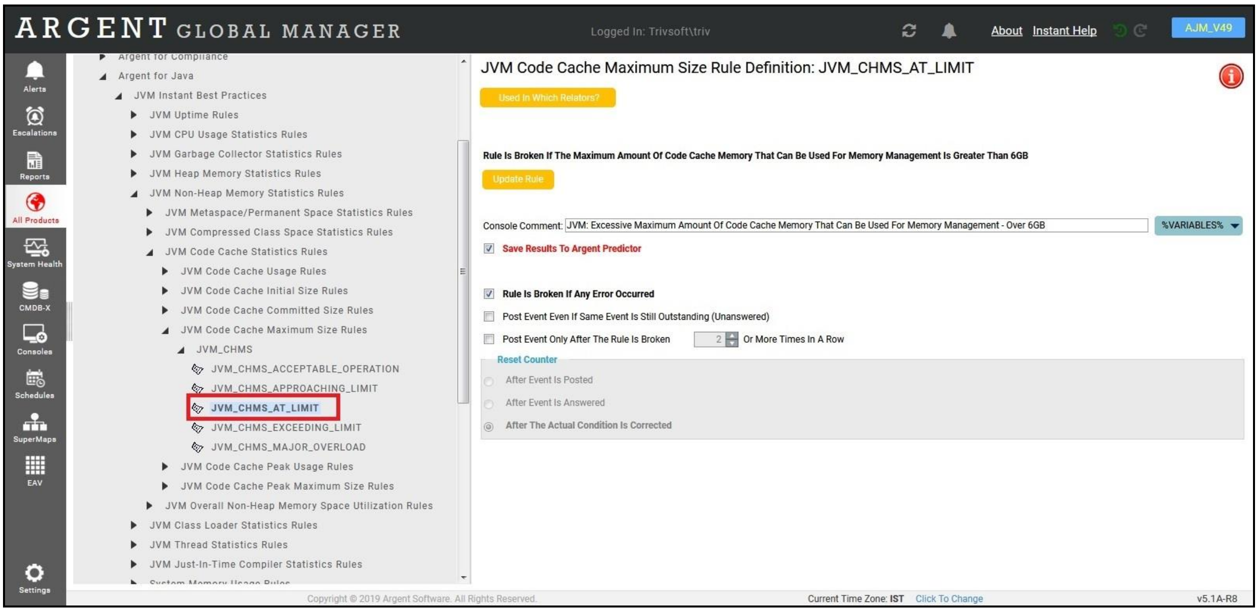
○ After Event Is Answered

● **After The Actual Condition Is Corrected**

Copyright © 2019 Argent Software. All Rights Reserved.   Current Time Zone: IST   Click To Change   v5.1A-R8

Code Cache Maximum Size

This rule monitors the maximum size of the space allowed for management of the code cache


ARGENT GLOBAL MANAGER
Logged In: Trivsoft\triv
About
Instant Help
AJM_V49
Alerts
Escalations
Reports
All Products
System Health
CMDB-X
Consoles
Schedules
SuperMaps
EAV
Settings
Argent for Compliance
Argent for Java
JVM Instant Best Practices
JVM Uptime Rules
JVM CPU Usage Statistics Rules
JVM Garbage Collector Statistics Rules
JVM Heap Memory Statistics Rules
JVM Non-Heap Memory Statistics Rules
JVM Metaspace/Permanent Space Statistics Rules
JVM Compressed Class Space Statistics Rules
JVM Code Cache Statistics Rules
JVM Code Cache Usage Rules
JVM Code Cache Initial Size Rules
JVM Code Cache Committed Size Rules
JVM Code Cache Maximum Size Rules
JVM_CHMS
JVM_CHMS_ACCEPTABLE_OPERATION
JVM_CHMS_APPROACHING_LIMIT
JVM_CHMS_AT_LIMIT
JVM_CHMS_EXCEEDING_LIMIT
JVM_CHMS_MAJOR_OVERLOAD
JVM Code Cache Peak Usage Rules
JVM Code Cache Peak Maximum Size Rules
JVM Overall Non-Heap Memory Space Utilization Rules
JVM Class Loader Statistics Rules
JVM Thread Statistics Rules
JVM Just-In-Time Compiler Statistics Rules
System Memory Usage Rules
Used In Which Relators?
Update Rule
%VARIABLES%
Copyright © 2019 Argent Software. All Rights Reserved.
Current Time Zone: IST
Click To Change
v5.1A-R8

JVM Code Cache Maximum Size Rule Definition: JVM_CHMS_AT_LIMIT

Rule Is Broken If The Maximum Amount Of Code Cache Memory That Can Be Used For Memory Management Is Greater Than 6GB

Console Comment: JVM: Excessive Maximum Amount Of Code Cache Memory That Can Be Used For Memory Management - Over 6GB

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

□ Post Event Even If Same Event Is Still Outstanding (Unanswered)

□ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

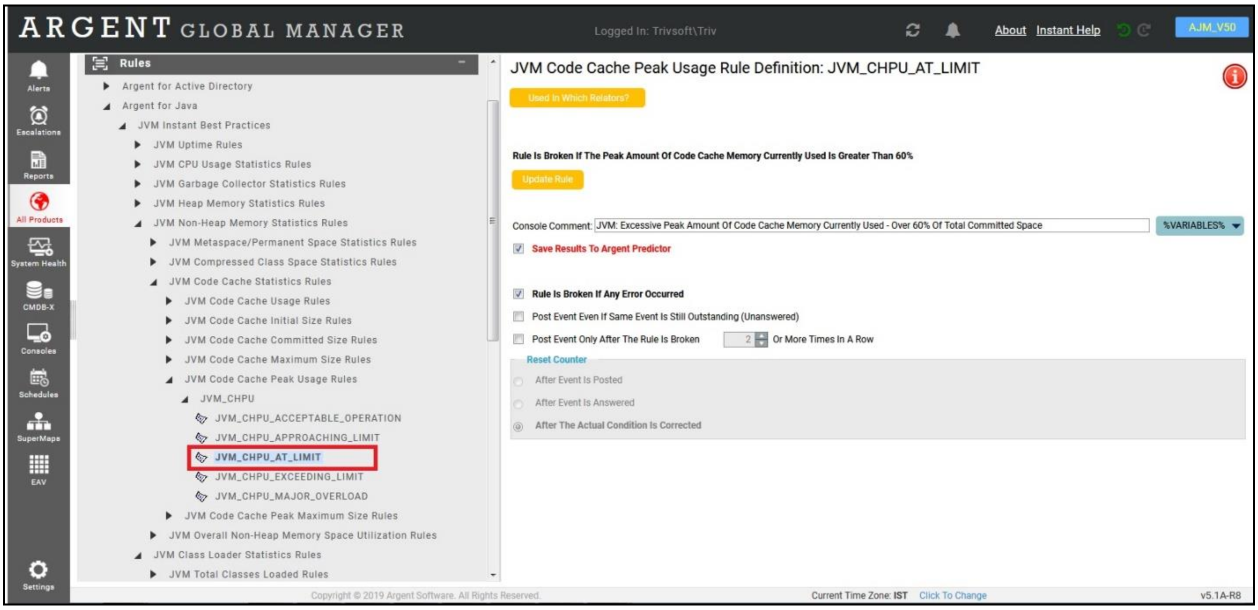
○ After Event Is Posted

○ After Event Is Answered

● After The Actual Condition Is Corrected

Code Cache Peak Usage

This rule monitors the peak amount of code cache memory currently used


ARGENT GLOBAL MANAGER
Logged In: Trivsoft\Triv
About
Instant Help
AJM_V50
Alerts
Escalations
Reports
All Products
System Health
CMDB-X
Consoles
Schedules
SuperMaps
EAV
Settings
Rules
Argent for Active Directory
Argent for Java
JVM Instant Best Practices
JVM Uptime Rules
JVM CPU Usage Statistics Rules
JVM Garbage Collector Statistics Rules
JVM Heap Memory Statistics Rules
JVM Non-Heap Memory Statistics Rules
JVM Metaspace/Permanent Space Statistics Rules
JVM Compressed Class Space Statistics Rules
JVM Code Cache Statistics Rules
JVM Code Cache Usage Rules
JVM Code Cache Initial Size Rules
JVM Code Cache Committed Size Rules
JVM Code Cache Maximum Size Rules
JVM Code Cache Peak Usage Rules
JVM_CHPU
JVM_CHPU_ACCEPTABLE_OPERATION
JVM_CHPU_APPROACHING_LIMIT
JVM_CHPU_AT_LIMIT
JVM_CHPU_EXCEEDING_LIMIT
JVM_CHPU_MAJOR_OVERLOAD
JVM Code Cache Peak Maximum Size Rules
JVM Overall Non-Heap Memory Space Utilization Rules
JVM Class Loader Statistics Rules
JVM Total Classes Loaded Rules
Used In Which Relators?
Update Rule
%VARIABLES%
Copyright © 2019 Argent Software. All Rights Reserved.
Current Time Zone: IST
Click To Change
v5.1A-R8

JVM Code Cache Peak Usage Rule Definition: JVM_CHPU_AT_LIMIT

Rule Is Broken If The Peak Amount Of Code Cache Memory Currently Used Is Greater Than 60%

Console Comment: JVM: Excessive Peak Amount Of Code Cache Memory Currently Used - Over 60% Of Total Committed Space

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

□ Post Event Even If Same Event Is Still Outstanding (Unanswered)

□ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

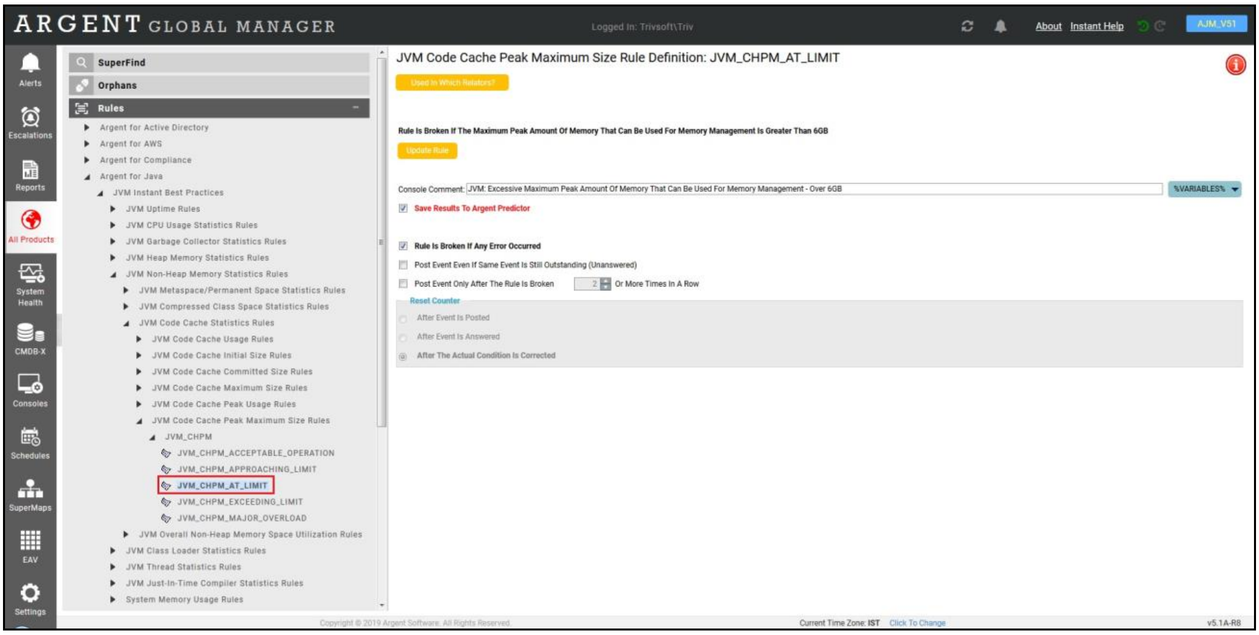
○ After Event Is Posted

○ After Event Is Answered

● After The Actual Condition Is Corrected

Code Cache Peak Maximum Size

This rule monitors the peak maximum size of code cache memory



ARGENT GLOBAL MANAGER
Logged In: Trivsoft\Triv
About
Instant Help
AJM_V51
Alerts
Escalations
Reports
All Products
System Health
CMDB-X
Consoles
Schedules
SuperMaps
EAV
Settings
SuperFind
Orphans
Rules
Argent for Active Directory
Argent for AWS
Argent for Compliance
Argent for Java
JVM Instant Best Practices
JVM Uptime Rules
JVM CPU Usage Statistics Rules
JVM Garbage Collector Statistics Rules
JVM Heap Memory Statistics Rules
JVM Non-Heap Memory Statistics Rules
JVM Metaspace/Permanent Space Statistics Rules
JVM Compressed Class Space Statistics Rules
JVM Code Cache Statistics Rules
JVM Code Cache Usage Rules
JVM Code Cache Initial Size Rules
JVM Code Cache Committed Size Rules
JVM Code Cache Maximum Size Rules
JVM Code Cache Peak Usage Rules
JVM Code Cache Peak Maximum Size Rules
JVM_CHPM
JVM_CHPM_ACCEPTABLE_OPERATION
JVM_CHPM_APPROACHING_LIMIT
JVM_CHPM_AT_LIMIT
JVM_CHPM_EXCEEDING_LIMIT
JVM_CHPM_MAJOR_OVERLOAD
JVM Overall Non-Heap Memory Space Utilization Rules
JVM Class Loader Statistics Rules
JVM Thread Statistics Rules
JVM Just-In-Time Compiler Statistics Rules
System Memory Usage Rules
Used In Which Relators?
Update Rule
%VARIABLES%
Copyright © 2019 Argent Software. All Rights Reserved.
Current Time Zone: IST
Click To Change
v5.1A-R8

JVM Code Cache Peak Maximum Size Rule Definition: JVM_CHPM_AT_LIMIT

Rule Is Broken If The Maximum Peak Amount Of Memory That Can Be Used For Memory Management Is Greater Than 6GB

Console Comment: JVM: Excessive Maximum Peak Amount Of Memory That Can Be Used For Memory Management - Over 6GB

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

☐ Post Event Even If Same Event Is Still Outstanding (Unanswered)

☐ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

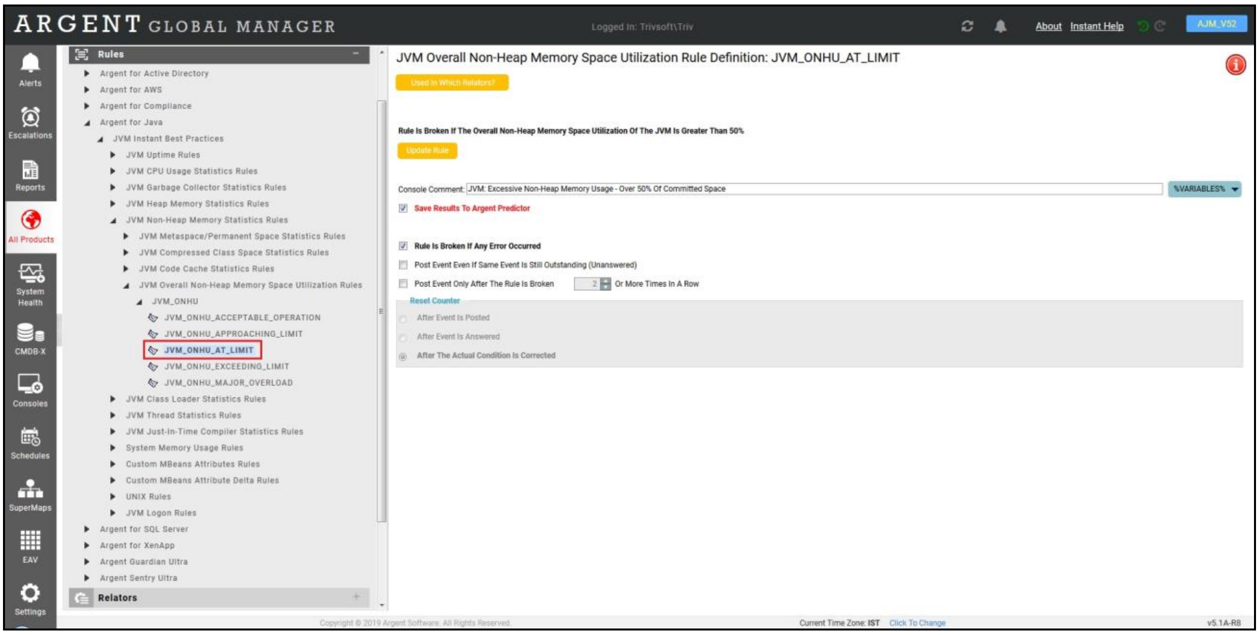
○ After Event Is Posted

○ After Event Is Answered

● After The Actual Condition Is Corrected

Overall Non-Heap Memory Utilization

This rule monitors the total amount of non-heap memory utilized


ARGENT GLOBAL MANAGER
Logged In: Trivsoft\Triv
About
Instant Help
AJM_V52
Alerts
Escalations
Reports
All Products
System Health
CMDB-X
Consoles
Schedules
SuperMaps
EAV
Settings
Rules
Argent for Active Directory
Argent for AWS
Argent for Compliance
Argent for Java
JVM Instant Best Practices
JVM Uptime Rules
JVM CPU Usage Statistics Rules
JVM Garbage Collector Statistics Rules
JVM Heap Memory Statistics Rules
JVM Non-Heap Memory Statistics Rules
JVM Metaspace/Permanent Space Statistics Rules
JVM Compressed Class Space Statistics Rules
JVM Code Cache Statistics Rules
JVM Overall Non-Heap Memory Space Utilization Rules
JVM_ONHU
JVM_ONHU_ACCEPTABLE_OPERATION
JVM_ONHU_APPROACHING_LIMIT
JVM_ONHU_AT_LIMIT
JVM_ONHU_EXCEEDING_LIMIT
JVM_ONHU_MAJOR_OVERLOAD
JVM Class Loader Statistics Rules
JVM Thread Statistics Rules
JVM Just-In-Time Compiler Statistics Rules
System Memory Usage Rules
Custom MBeans Attributes Rules
Custom MBeans Attribute Delta Rules
UNIX Rules
JVM Logon Rules
Argent for SQL Server
Argent for XenApp
Argent Guardian Ultra
Argent Sentry Ultra
Relators
Used In Which Relators?
Update Rule
%VARIABLES%
Copyright © 2019 Argent Software. All Rights Reserved.
Current Time Zone: IST
Click To Change
v5.1A-R8

JVM Overall Non-Heap Memory Space Utilization Rule Definition: JVM_ONHU_AT_LIMIT

Rule Is Broken If The Overall Non-Heap Memory Space Utilization Of The JVM Is Greater Than 50%

Console Comment: JVM: Excessive Non-Heap Memory Usage - Over 50% Of Committed Space

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

☐ Post Event Even If Same Event Is Still Outstanding (Unanswered)

☐ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

○ After Event Is Posted

○ After Event Is Answered

● After The Actual Condition Is Corrected

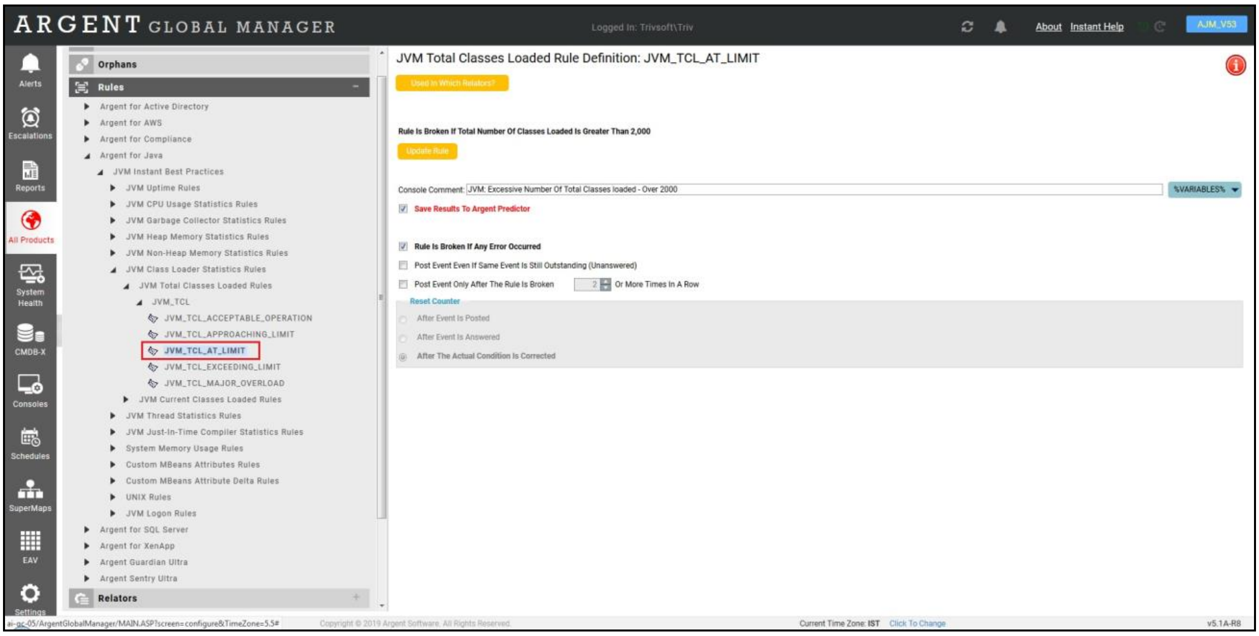
Java Class Loader Statistics Rules

One of Java’s main features is the Class Loader which is part of the Java Runtime Environment or JRE. The JRE loads all required classes during execution of a Java program. The JRE instantiates class objects at run time on demand.

Argent for Java provides the following rules for monitoring the JRE Class Loader.

Total Classes Loaded Rule

This rule monitors the number of classes loaded from the start of a JVM


ARGENT GLOBAL MANAGER
Logged In: Trivsoft\Triv
About
Instant Help
AJM_V53
Alerts
Escalations
Reports
All Products
System Health
CMDB-X
Consoles
Schedules
SuperMaps
EAV
Settings
Orphans
Rules
Argent for Active Directory
Argent for AWS
Argent for Compliance
Argent for Java
JVM Instant Best Practices
JVM Uptime Rules
JVM CPU Usage Statistics Rules
JVM Garbage Collector Statistics Rules
JVM Heap Memory Statistics Rules
JVM Non-Heap Memory Statistics Rules
JVM Class Loader Statistics Rules
JVM Total Classes Loaded Rules
JVM_TCL
JVM_TCL_ACCEPTABLE_OPERATION
JVM_TCL_APPROACHING_LIMIT
JVM_TCL_AT_LIMIT
JVM_TCL_EXCEEDING_LIMIT
JVM_TCL_MAJOR_OVERLOAD
JVM Current Classes Loaded Rules
JVM Thread Statistics Rules
JVM Just-In-Time Compiler Statistics Rules
System Memory Usage Rules
Custom MBeans Attributes Rules
Custom MBeans Attribute Delta Rules
UNIX Rules
JVM Logon Rules
Argent for SQL Server
Argent for XenApp
Argent Guardian Ultra
Argent Sentry Ultra
Relators
Used In Which Relators?
Update Rule
%VARIABLES%
ai-gc-05/ArgentGlobalManager/MAIN.ASP?screen=configure&TimeZone=5.5#
Copyright © 2019 Argent Software. All Rights Reserved.
Current Time Zone: IST
Click To Change
v5.1A-R8

JVM Total Classes Loaded Rule Definition: JVM_TCL_AT_LIMIT

Rule Is Broken If Total Number Of Classes Loaded Is Greater Than 2,000

Console Comment: JVM: Excessive Number Of Total Classes loaded - Over 2000

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

□ Post Event Even If Same Event Is Still Outstanding (Unanswered)

□ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

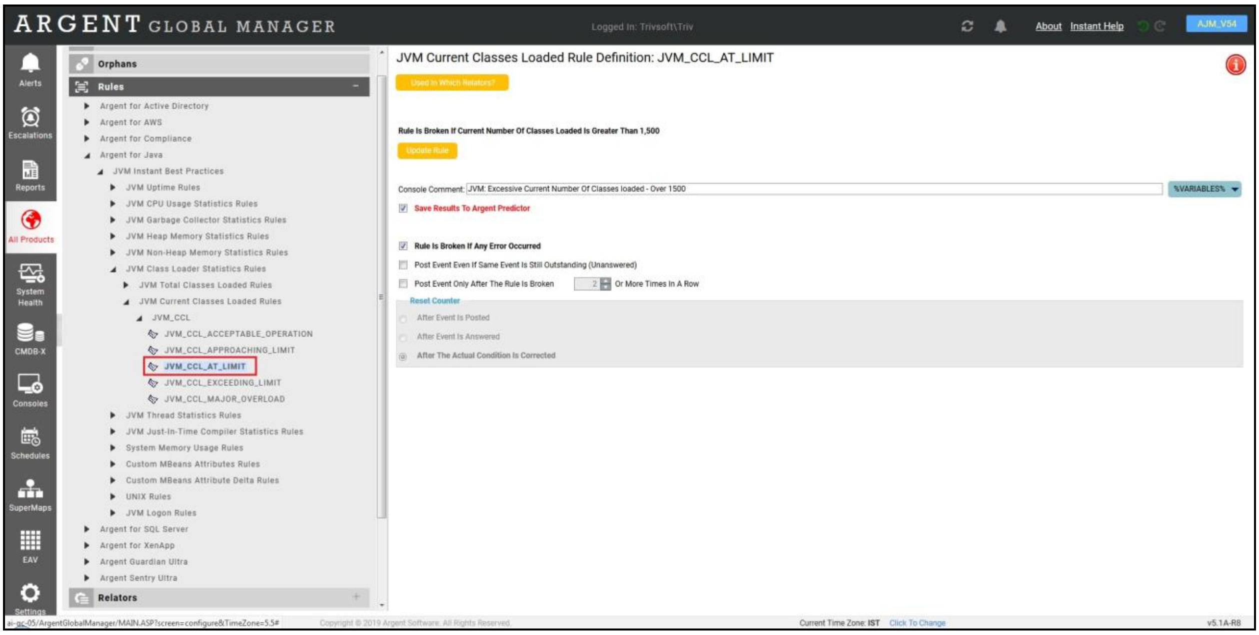
○ After Event Is Posted

○ After Event Is Answered

● After The Actual Condition Is Corrected

Current Classes Loaded Count

This rule monitors the number of classes loaded in a JVM at any given time


ARGENT GLOBAL MANAGER
Logged In: Trivsoft\Triv
About
Instant Help
AJM_V54
Alerts
Escalations
Reports
All Products
System Health
CMDB-X
Consoles
Schedules
SuperMaps
EAV
Settings
Orphans
Rules
Argent for Active Directory
Argent for AWS
Argent for Compliance
Argent for Java
JVM Instant Best Practices
JVM Uptime Rules
JVM CPU Usage Statistics Rules
JVM Garbage Collector Statistics Rules
JVM Heap Memory Statistics Rules
JVM Non-Heap Memory Statistics Rules
JVM Class Loader Statistics Rules
JVM Total Classes Loaded Rules
JVM Current Classes Loaded Rules
JVM_CCL
JVM_CCL_ACCEPTABLE_OPERATION
JVM_CCL_APPROACHING_LIMIT
JVM_CCL_AT_LIMIT
JVM_CCL_EXCEEDING_LIMIT
JVM_CCL_MAJOR_OVERLOAD
JVM Thread Statistics Rules
JVM Just-In-Time Compiler Statistics Rules
System Memory Usage Rules
Custom MBeans Attributes Rules
Custom MBeans Attribute Delta Rules
UNIX Rules
JVM Logon Rules
Argent for SQL Server
Argent for XenApp
Argent Guardian Ultra
Argent Sentry Ultra
Relators
Used In Which Relators?
Update Rule
%VARIABLES%
ai-gc-05/ArgentGlobalManager/MAIN.ASP?screen=configure&TimeZone=5.5#
Copyright © 2019 Argent Software. All Rights Reserved.
Current Time Zone: IST
Click To Change
v5.1A-R8

JVM Current Classes Loaded Rule Definition: JVM_CCL_AT_LIMIT

Rule Is Broken If Current Number Of Classes Loaded Is Greater Than 1,500

Console Comment: JVM: Excessive Current Number Of Classes loaded - Over 1500

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

□ Post Event Even If Same Event Is Still Outstanding (Unanswered)

□ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

○ After Event Is Posted

○ After Event Is Answered

● After The Actual Condition Is Corrected

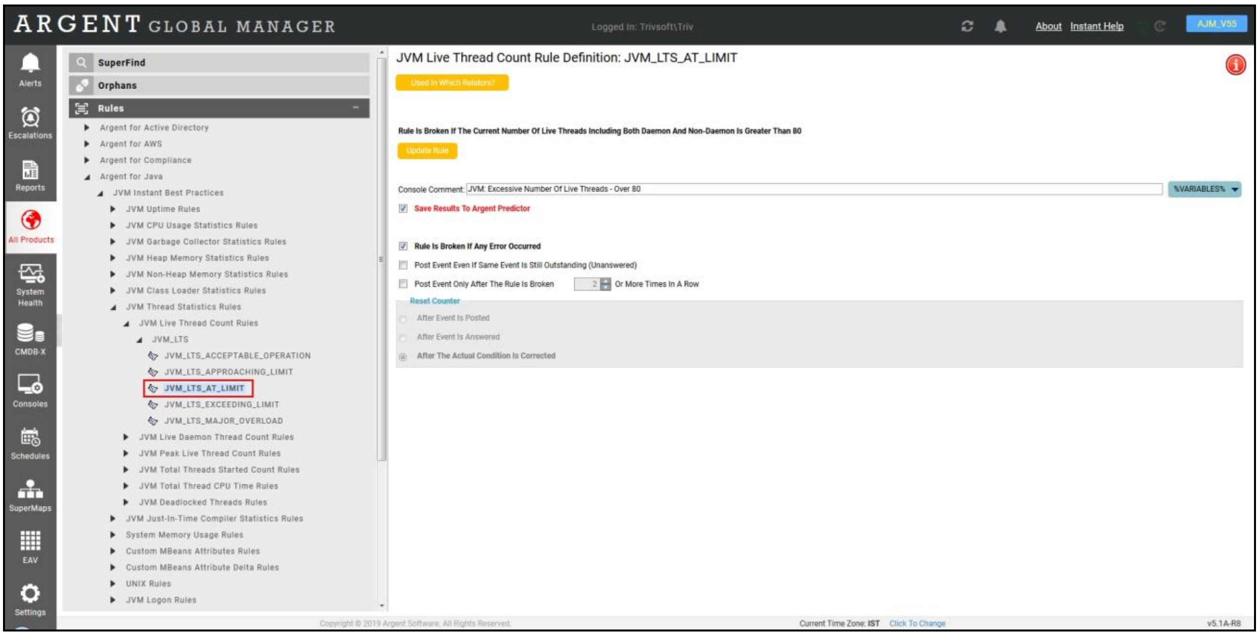
Thread Statistics Rules

A JVM will always contain multiple Java threads, including Java internal JVM threads.

Argent for Java provides the following rules to monitor JVM threads.

Live Thread Count Rule

This rule monitors the count of threads in a JVM at any given point of time


ARGENT GLOBAL MANAGER
Logged In: Trivsoft\Triv
About
Instant Help
AJM_V55
Alerts
Escalations
Reports
All Products
System Health
CMDB-X
Consoles
Schedules
SuperMaps
EAV
Settings
SuperFind
Orphans
Rules
Argent for Active Directory
Argent for AWS
Argent for Compliance
Argent for Java
JVM Instant Best Practices
JVM Uptime Rules
JVM CPU Usage Statistics Rules
JVM Garbage Collector Statistics Rules
JVM Heap Memory Statistics Rules
JVM Non-Heap Memory Statistics Rules
JVM Class Loader Statistics Rules
JVM Thread Statistics Rules
JVM Live Thread Count Rules
JVM_LTS
JVM_LTS_ACCEPTABLE_OPERATION
JVM_LTS_APPROACHING_LIMIT
JVM_LTS_AT_LIMIT
JVM_LTS_EXCEEDING_LIMIT
JVM_LTS_MAJOR_OVERLOAD
JVM Live Daemon Thread Count Rules
JVM Peak Live Thread Count Rules
JVM Total Threads Started Count Rules
JVM Total Thread CPU Time Rules
JVM Deadlocked Threads Rules
JVM Just-In-Time Compiler Statistics Rules
System Memory Usage Rules
Custom MBeans Attributes Rules
Custom MBeans Attribute Delta Rules
UNIX Rules
JVM Logon Rules
Used In Which Relators?
Update Rule
%VARIABLES%
Copyright © 2019 Argent Software. All Rights Reserved.
Current Time Zone: IST
Click To Change
v5.1A-R8

JVM Live Thread Count Rule Definition: JVM_LTS_AT_LIMIT

Rule Is Broken If The Current Number Of Live Threads Including Both Daemon And Non-Daemon Is Greater Than 80

Console Comment: JVM: Excessive Number Of Live Threads - Over 80

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

□ Post Event Even If Same Event Is Still Outstanding (Unanswered)

□ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

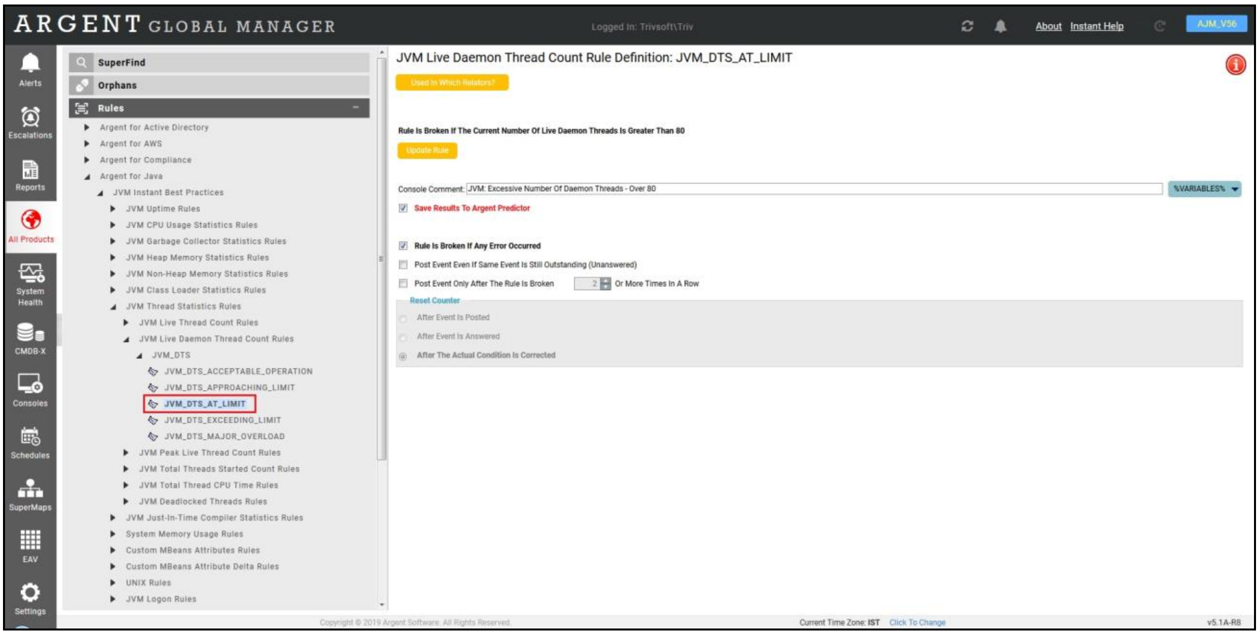
○ After Event Is Posted

○ After Event Is Answered

● After The Actual Condition Is Corrected

Live Daemon Thread Count Rule

Java performs some low-priority tasks in daemon threads that execute only after all the non-daemon threads have executed. The JVM can even exit a program while the Daemon threads are running. Argent for Java provides a rule to monitor the number of daemon threads running.


ARGENT GLOBAL MANAGER
Logged In: Trivsoft\Triv
About
Instant Help
AJM_V56
Alerts
Escalations
Reports
All Products
System Health
CMDB-X
Consoles
Schedules
SuperMaps
EAV
Settings
SuperFind
Orphans
Rules
Argent for Active Directory
Argent for AWS
Argent for Compliance
Argent for Java
JVM Instant Best Practices
JVM Uptime Rules
JVM CPU Usage Statistics Rules
JVM Garbage Collector Statistics Rules
JVM Heap Memory Statistics Rules
JVM Non-Heap Memory Statistics Rules
JVM Class Loader Statistics Rules
JVM Thread Statistics Rules
JVM Live Thread Count Rules
JVM Live Daemon Thread Count Rules
JVM_DTS
JVM_DTS_ACCEPTABLE_OPERATION
JVM_DTS_APPROACHING_LIMIT
JVM_DTS_AT_LIMIT
JVM_DTS_EXCEEDING_LIMIT
JVM_DTS_MAJOR_OVERLOAD
JVM Peak Live Thread Count Rules
JVM Total Threads Started Count Rules
JVM Total Thread CPU Time Rules
JVM Deadlocked Threads Rules
JVM Just-In-Time Compiler Statistics Rules
System Memory Usage Rules
Custom MBeans Attributes Rules
Custom MBeans Attribute Delta Rules
UNIX Rules
JVM Logon Rules
Used In Which Relators?
Update Rule
%VARIABLES%
Copyright © 2019 Argent Software. All Rights Reserved.
Current Time Zone: IST
Click To Change
v5.1A-R8

JVM Live Daemon Thread Count Rule Definition: JVM_DTS_AT_LIMIT

Rule Is Broken If The Current Number Of Live Daemon Threads Is Greater Than 80

Console Comment: JVM: Excessive Number Of Daemon Threads - Over 80

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

☐ Post Event Even If Same Event Is Still Outstanding (Unanswered)

☐ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

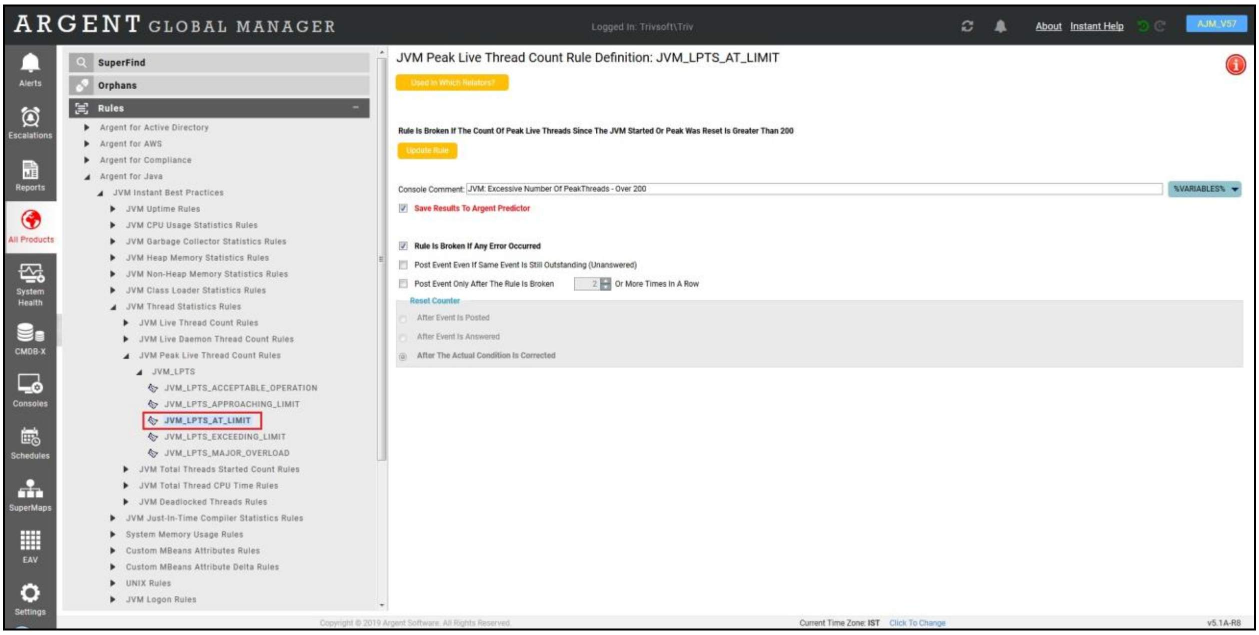
○ After Event Is Posted

○ After Event Is Answered

● After The Actual Condition Is Corrected

Peak Live Thread Count Rule

This rule monitors the peak thread live count since the JVM started (or the peak value was reset)


ARGENT GLOBAL MANAGER
Logged In: Trivsoft\Triv
About
Instant Help
AJM_V57
Alerts
Escalations
Reports
All Products
System Health
CMDB-X
Consoles
Schedules
SuperMaps
EAV
Settings
SuperFind
Orphans
Rules
Argent for Active Directory
Argent for AWS
Argent for Compliance
Argent for Java
JVM Instant Best Practices
JVM Uptime Rules
JVM CPU Usage Statistics Rules
JVM Garbage Collector Statistics Rules
JVM Heap Memory Statistics Rules
JVM Non-Heap Memory Statistics Rules
JVM Class Loader Statistics Rules
JVM Thread Statistics Rules
JVM Live Thread Count Rules
JVM Live Daemon Thread Count Rules
JVM Peak Live Thread Count Rules
JVM_LPTS
JVM_LPTS_ACCEPTABLE_OPERATION
JVM_LPTS_APPROACHING_LIMIT
JVM_LPTS_AT_LIMIT
JVM_LPTS_EXCEEDING_LIMIT
JVM_LPTS_MAJOR_OVERLOAD
JVM Total Threads Started Count Rules
JVM Total Thread CPU Time Rules
JVM Deadlocked Threads Rules
JVM Just-In-Time Compiler Statistics Rules
System Memory Usage Rules
Custom MBeans Attributes Rules
Custom MBeans Attribute Delta Rules
UNIX Rules
JVM Logon Rules
Used In Which Relators?
Update Rule
%VARIABLES%
Copyright © 2019 Argent Software. All Rights Reserved.
Current Time Zone: IST
Click To Change
v5.1A-R8

JVM Peak Live Thread Count Rule Definition: JVM_LPTS_AT_LIMIT

Rule Is Broken If The Count Of Peak Live Threads Since The JVM Started Or Peak Was Reset Is Greater Than 200

Console Comment: JVM: Excessive Number Of PeakThreads - Over 200

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

□ Post Event Even If Same Event Is Still Outstanding (Unanswered)

□ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

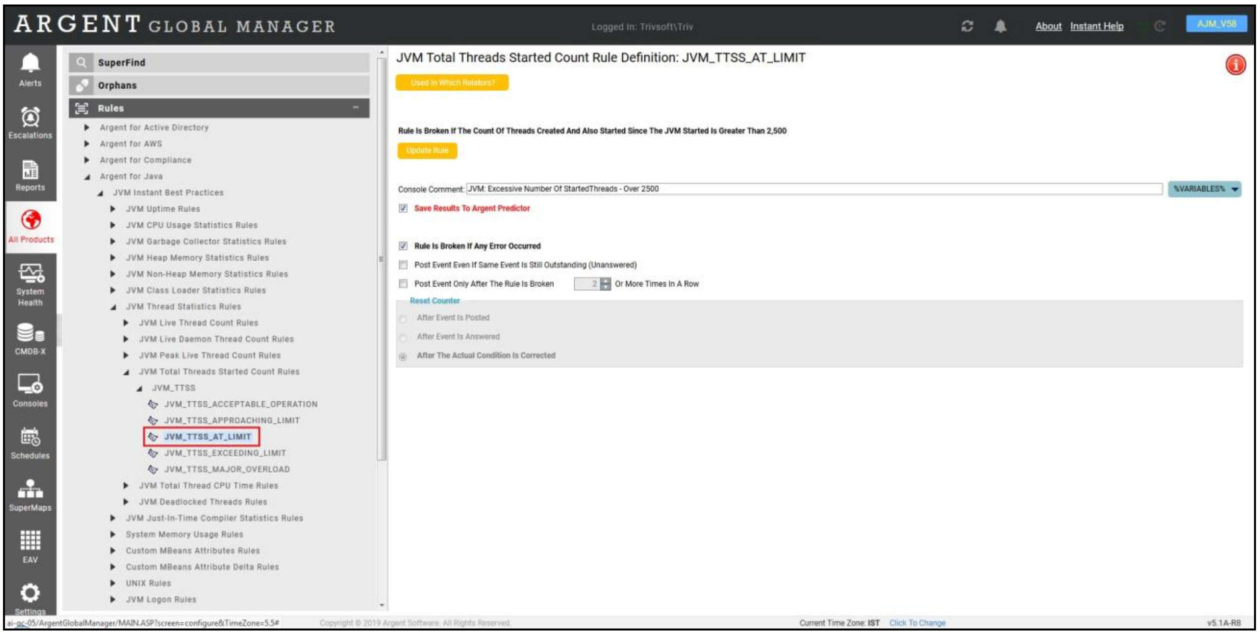
○ After Event Is Posted

○ After Event Is Answered

● After The Actual Condition Is Corrected

Total Threads Started Count Rule

This rule monitors the total number of threads created or started since the JVM started


ARGENT GLOBAL MANAGER
Logged In: Trivsoft\Triv
About
Instant Help
AJM_V58
Alerts
Escalations
Reports
All Products
System Health
CMDB-X
Consoles
Schedules
SuperMaps
EAV
Settings
SuperFind
Orphans
Rules
Argent for Active Directory
Argent for AWS
Argent for Compliance
Argent for Java
JVM Instant Best Practices
JVM Uptime Rules
JVM CPU Usage Statistics Rules
JVM Garbage Collector Statistics Rules
JVM Heap Memory Statistics Rules
JVM Non-Heap Memory Statistics Rules
JVM Class Loader Statistics Rules
JVM Thread Statistics Rules
JVM Live Thread Count Rules
JVM Live Daemon Thread Count Rules
JVM Peak Live Thread Count Rules
JVM Total Threads Started Count Rules
JVM_TTSS
JVM_TTSS_ACCEPTABLE_OPERATION
JVM_TTSS_APPROACHING_LIMIT
JVM_TTSS_AT_LIMIT
JVM_TTSS_EXCEEDING_LIMIT
JVM_TTSS_MAJOR_OVERLOAD
JVM Total Thread CPU Time Rules
JVM Deadlocked Threads Rules
JVM Just-In-Time Compiler Statistics Rules
System Memory Usage Rules
Custom MBeans Attributes Rules
Custom MBeans Attribute Delta Rules
UNIX Rules
JVM Logon Rules
Used In Which Relators?
Update Rule
%VARIABLES%

JVM Total Threads Started Count Rule Definition: JVM_TTSS_AT_LIMIT

Rule Is Broken If The Count Of Threads Created And Also Started Since The JVM Started Is Greater Than 2,500

Console Comment: JVM: Excessive Number Of StartedThreads - Over 2500

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

☐ Post Event Even If Same Event Is Still Outstanding (Unanswered)

☐ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

○ After Event Is Posted

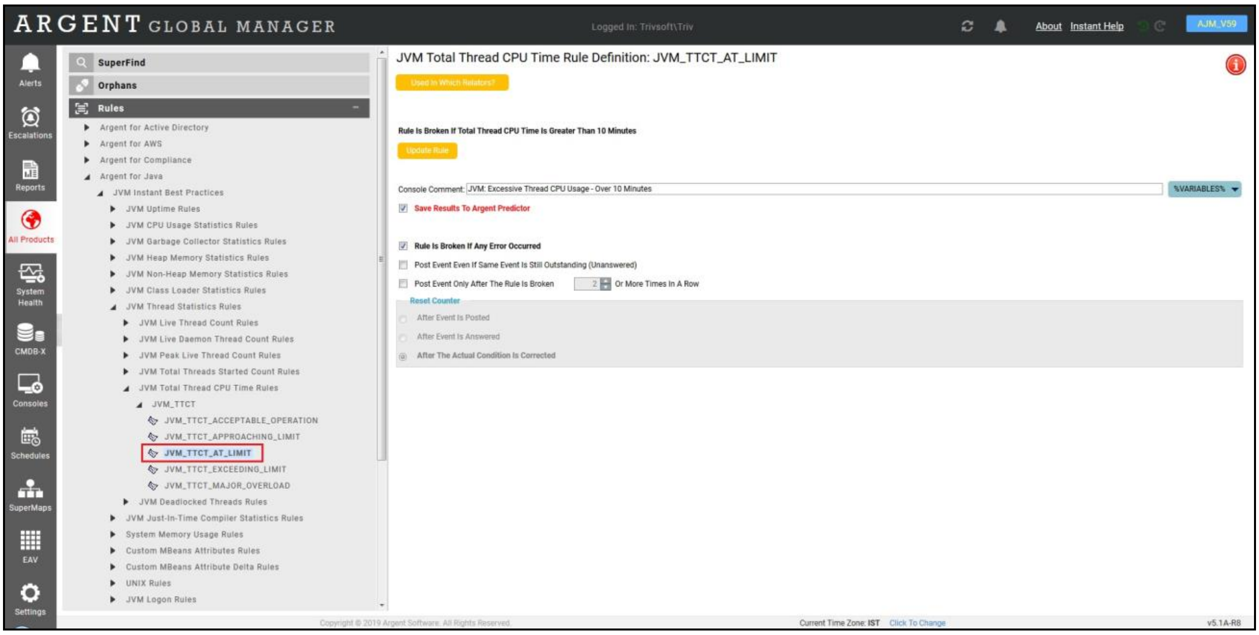
○ After Event Is Answered

● After The Actual Condition Is Corrected

ai-gc-05/ArgentGlobalManager/MAIN.ASP?screen=configure&TimeZone=5.5#   Copyright © 2019 Argent Software. All Rights Reserved.   Current Time Zone: IST Click To Change   v5.1A-R8

Total Thread CPU Time Rule

This rule monitors the total CPU time utilized by all threads in a JVM


ARGENT GLOBAL MANAGER
Logged In: Trivsoft\Triv
About
Instant Help
AJM_V59
Alerts
Escalations
Reports
All Products
System Health
CMDB-X
Consoles
Schedules
SuperMaps
EAV
Settings
SuperFind
Orphans
Rules
Argent for Active Directory
Argent for AWS
Argent for Compliance
Argent for Java
JVM Instant Best Practices
JVM Uptime Rules
JVM CPU Usage Statistics Rules
JVM Garbage Collector Statistics Rules
JVM Heap Memory Statistics Rules
JVM Non-Heap Memory Statistics Rules
JVM Class Loader Statistics Rules
JVM Thread Statistics Rules
JVM Live Thread Count Rules
JVM Live Daemon Thread Count Rules
JVM Peak Live Thread Count Rules
JVM Total Threads Started Count Rules
JVM Total Thread CPU Time Rules
JVM_TTCT
JVM_TTCT_ACCEPTABLE_OPERATION
JVM_TTCT_APPROACHING_LIMIT
JVM_TTCT_AT_LIMIT
JVM_TTCT_EXCEEDING_LIMIT
JVM_TTCT_MAJOR_OVERLOAD
JVM Deadlocked Threads Rules
JVM Just-In-Time Compiler Statistics Rules
System Memory Usage Rules
Custom MBeans Attributes Rules
Custom MBeans Attribute Delta Rules
UNIX Rules
JVM Logon Rules
Used In Which Relators?
Update Rule
%VARIABLES%
Copyright © 2019 Argent Software. All Rights Reserved.
Current Time Zone: IST
Click To Change
v5.1A-R8

JVM Total Thread CPU Time Rule Definition: JVM_TTCT_AT_LIMIT

Rule Is Broken If Total Thread CPU Time Is Greater Than 10 Minutes

Console Comment: JVM: Excessive Thread CPU Usage - Over 10 Minutes

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

☐ Post Event Even If Same Event Is Still Outstanding (Unanswered)

☐ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

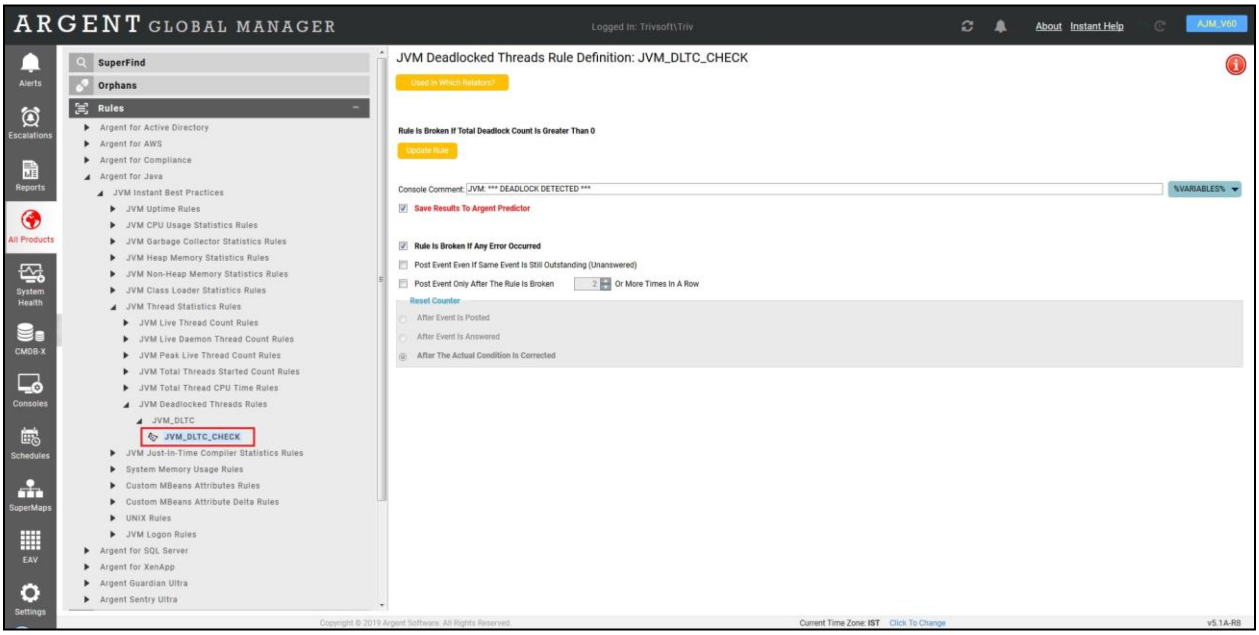
○ After Event Is Posted

○ After Event Is Answered

● After The Actual Condition Is Corrected

Deadlocked Threads Rule

When two or more threads in the JVM are waiting for the same resource the threads become deadlocked. Deadlocks cannot be resolved the executing Java program appears to be stuck. Argent for Java monitors all live threads in a JVM for deadlock conditions.


ARGENT GLOBAL MANAGER
Logged In: Trivsoft\Triv
About
Instant Help
AJM_V60
Alerts
Escalations
Reports
All Products
System Health
CMDB-X
Consoles
Schedules
SuperMaps
EAV
Settings
SuperFind
Orphans
Rules
Argent for Active Directory
Argent for AWS
Argent for Compliance
Argent for Java
JVM Instant Best Practices
JVM Uptime Rules
JVM CPU Usage Statistics Rules
JVM Garbage Collector Statistics Rules
JVM Heap Memory Statistics Rules
JVM Non-Heap Memory Statistics Rules
JVM Class Loader Statistics Rules
JVM Thread Statistics Rules
JVM Live Thread Count Rules
JVM Live Daemon Thread Count Rules
JVM Peak Live Thread Count Rules
JVM Total Threads Started Count Rules
JVM Total Thread CPU Time Rules
JVM Deadlocked Threads Rules
JVM_DLTC
JVM_DLTC_CHECK
JVM Just-In-Time Compiler Statistics Rules
System Memory Usage Rules
Custom MBeans Attributes Rules
Custom MBeans Attribute Delta Rules
UNIX Rules
JVM Logon Rules
Argent for SQL Server
Argent for XenApp
Argent Guardian Ultra
Argent Sentry Ultra
Used In Which Relators?
Update Rule
%VARIABLES%

JVM Deadlocked Threads Rule Definition: JVM_DLTC_CHECK

Rule Is Broken If Total Deadlock Count Is Greater Than 0

Console Comment: JVM: *** DEADLOCK DETECTED ***

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

□ Post Event Even If Same Event Is Still Outstanding (Unanswered)

□ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

○ After Event Is Posted

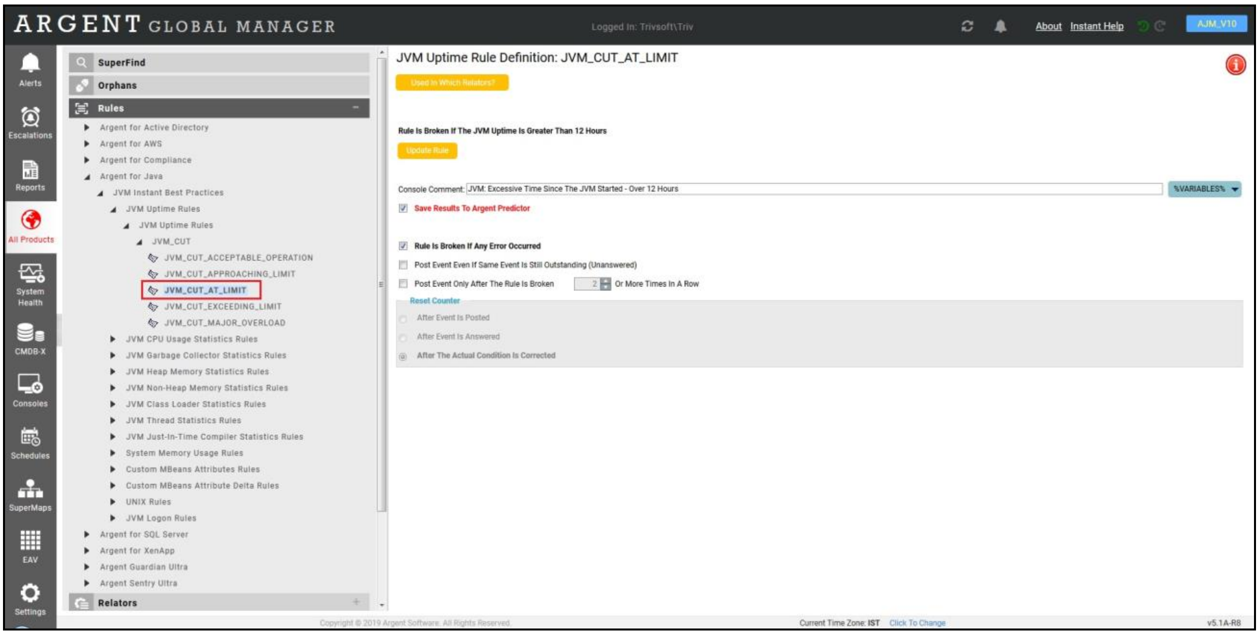
○ After Event Is Answered

● After The Actual Condition Is Corrected

Copyright © 2019 Argent Software. All Rights Reserved. Current Time Zone: IST Click To Change v5.1A-R8

Uptime Rule

This rule monitors the length of time the JVM has been active



JVM Uptime Rule Definition: JVM_CUT_AT_LIMIT

Rule Is Broken If The JVM Uptime Is Greater Than 12 Hours

Console Comment: JVM: Excessive Time Since The JVM Started - Over 12 Hours

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

□ Post Event Even If Same Event Is Still Outstanding (Unanswered)

□ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

○ After Event Is Posted

○ After Event Is Answered

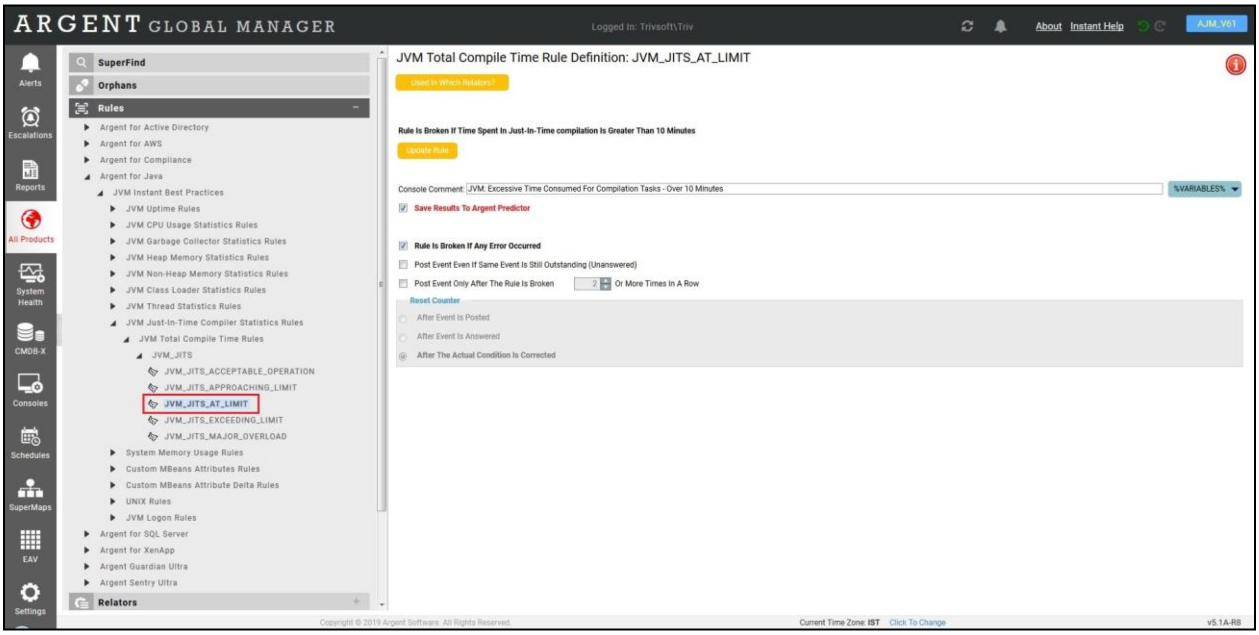
● After The Actual Condition Is Corrected

Just-In-Time Compiler Statistics Rules

The Java compiler converts the Java code into a bytecode which is then translated into machine instructions understandable by a particular machine or device. The Just-in-Time (JIT) compiler is how Java can be adapted to so many platforms.   Since the JIT compilation process takes place as a Java program executes, it can impact performance.

Total Compilation Time Rule

This rule monitors total time consumed by JIT compilation within a JVM



JVM Total Compile Time Rule Definition: JVM_JITS_AT_LIMIT

Rule Is Broken If Time Spent In Just-In-Time compilation Is Greater Than 10 Minutes

Console Comment: JVM: Excessive Time Consumed For Compilation Tasks - Over 10 Minutes

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

□ Post Event Even If Same Event Is Still Outstanding (Unanswered)

□ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

○ After Event Is Posted

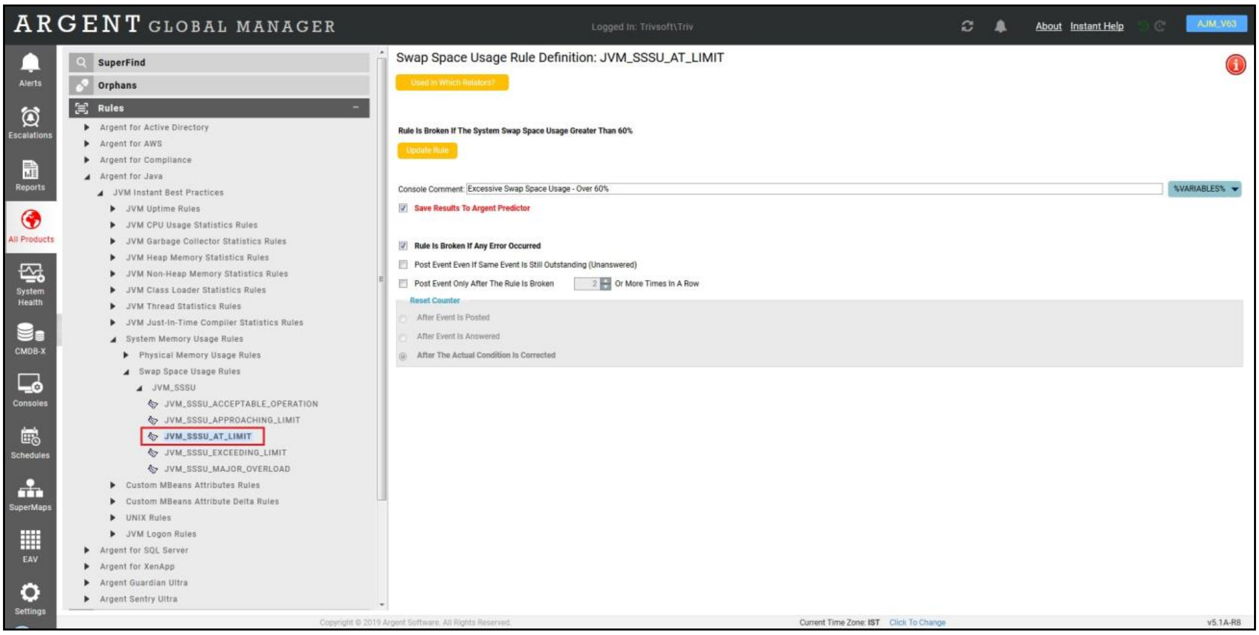
○ After Event Is Answered

● After The Actual Condition Is Corrected

System Memory Usage Rule

Swap Space Usage Rule

This rule monitors the total amount of swap space usage



Swap Space Usage Rule Definition: JVM_SSSU_AT_LIMIT

Rule Is Broken If The System Swap Space Usage Greater Than 60%

Console Comment: Excessive Swap Space Usage - Over 60%

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

□ Post Event Even If Same Event Is Still Outstanding (Unanswered)

□ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

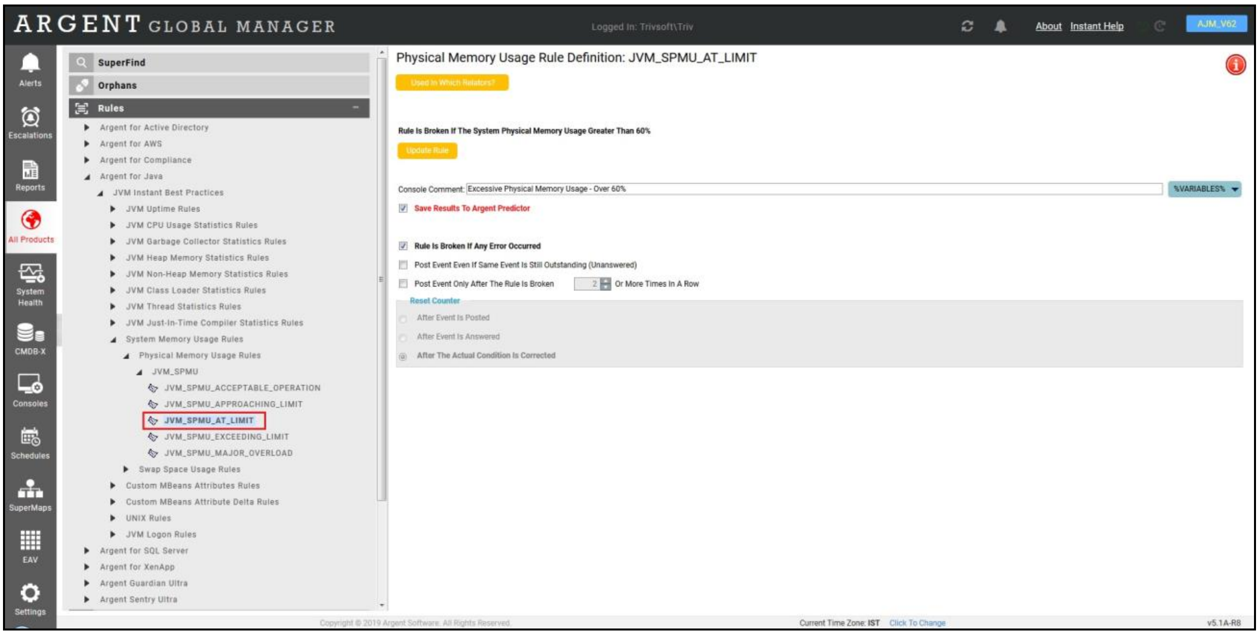
○ After Event Is Posted

○ After Event Is Answered

● After The Actual Condition Is Corrected

Physical Memory Usage Rule

This rule monitors physical memory usage within a JVM



Physical Memory Usage Rule Definition: JVM_SPMU_AT_LIMIT

Rule Is Broken If The System Physical Memory Usage Greater Than 60%

Console Comment: Excessive Physical Memory Usage - Over 60%

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

□ Post Event Even If Same Event Is Still Outstanding (Unanswered)

□ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

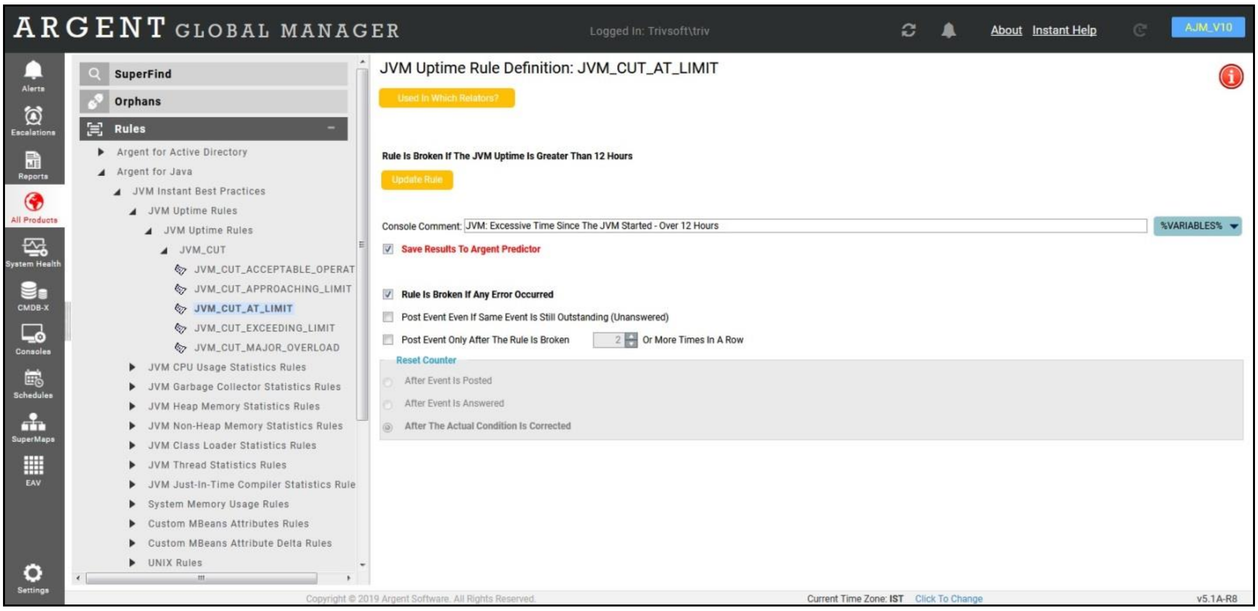
○ After Event Is Posted

○ After Event Is Answered

● After The Actual Condition Is Corrected

CPU Uptime Rule

This rule monitors JVM uptime


ARGENT GLOBAL MANAGER
Logged In: Trivsoft\triv
About
Instant Help
AJM_V10
Alerts
Escalations
Reports
All Products
System Health
CMDB-X
Consoles
Schedules
SuperMaps
EAV
Settings
SuperFind
Orphans
Rules
Argent for Active Directory
Argent for Java
JVM Instant Best Practices
JVM Uptime Rules
JVM Uptime Rules
JVM_CUT
JVM_CUT_ACCEPTABLE_OPERAT
JVM_CUT_APPROACHING_LIMIT
JVM_CUT_AT_LIMIT
JVM_CUT_EXCEEDING_LIMIT
JVM_CUT_MAJOR_OVERLOAD
JVM CPU Usage Statistics Rules
JVM Garbage Collector Statistics Rules
JVM Heap Memory Statistics Rules
JVM Non-Heap Memory Statistics Rules
JVM Class Loader Statistics Rules
JVM Thread Statistics Rules
JVM Just-In-Time Compiler Statistics Rule
System Memory Usage Rules
Custom MBeans Attributes Rules
Custom MBeans Attribute Delta Rules
UNIX Rules
Used In Which Relators?
Update Rule
%VARIABLES%
Copyright © 2019 Argent Software. All Rights Reserved.
Current Time Zone: IST
Click To Change
v5.1A-R8

JVM Uptime Rule Definition: JVM_CUT_AT_LIMIT

Rule Is Broken If The JVM Uptime Is Greater Than 12 Hours

Console Comment: JVM: Excessive Time Since The JVM Started - Over 12 Hours

☑ Save Results To Argent Predictor

☑ Rule Is Broken If Any Error Occurred

□ Post Event Even If Same Event Is Still Outstanding (Unanswered)

□ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

○ After Event Is Posted

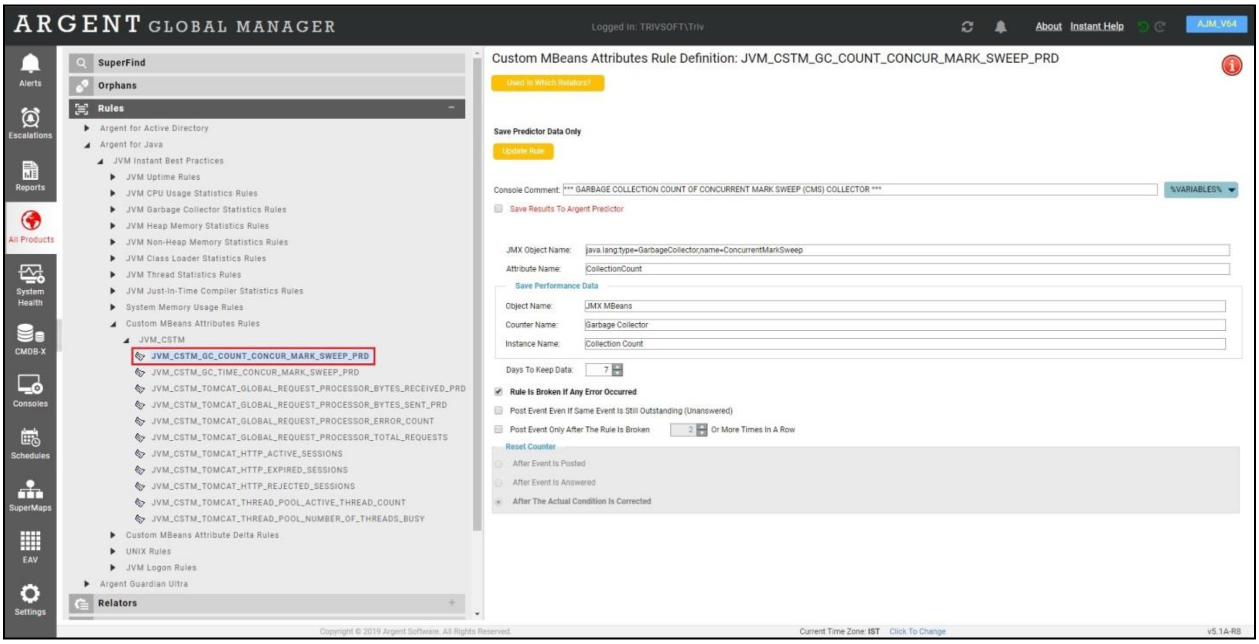
○ After Event Is Answered

● After The Actual Condition Is Corrected

Custom MBeans Attributes Rules

This Rule is configured to save the collection count of Concurrent Mark Sweep (CMS) Collector to Argent Predictor database and to a custom database table named 'ARGSOFT_JAVA_MBEANS'

Configure 'Custom MBeans Attribute Delta Rules' to Alert based on the values saved in 'ARGSOFT_JAVA_MBEANS' table for a specific Object\Counter\Instance


ARGENT GLOBAL MANAGER
Logged In: TRIVSOFT\Triv
About
Instant Help
AJM_V64
Alerts
Escalations
Reports
All Products
System Health
CMDB-X
Consoles
Schedules
SuperMaps
EAV
Settings
SuperFind
Orphans
Rules
Argent for Active Directory
Argent for Java
JVM Instant Best Practices
JVM Uptime Rules
JVM CPU Usage Statistics Rules
JVM Garbage Collector Statistics Rules
JVM Heap Memory Statistics Rules
JVM Non-Heap Memory Statistics Rules
JVM Class Loader Statistics Rules
JVM Thread Statistics Rules
JVM Just-In-Time Compiler Statistics Rules
System Memory Usage Rules
Custom MBeans Attributes Rules
JVM_CSTM
JVM_CSTM_GC_COUNT_CONCUR_MARK_SWEEP_PRD
JVM_CSTM_GC_TIME_CONCUR_MARK_SWEEP_PRD
JVM_CSTM_TOMCAT_GLOBAL_REQUEST_PROCESSOR_BYTES_RECEIVED_PRD
JVM_CSTM_TOMCAT_GLOBAL_REQUEST_PROCESSOR_BYTES_SENT_PRD
JVM_CSTM_TOMCAT_GLOBAL_REQUEST_PROCESSOR_ERROR_COUNT
JVM_CSTM_TOMCAT_GLOBAL_REQUEST_PROCESSOR_TOTAL_REQUESTS
JVM_CSTM_TOMCAT_HTTP_ACTIVE_SESSIONS
JVM_CSTM_TOMCAT_HTTP_EXPIRED_SESSIONS
JVM_CSTM_TOMCAT_HTTP_REJECTED_SESSIONS
JVM_CSTM_TOMCAT_THREAD_POOL_ACTIVE_THREAD_COUNT
JVM_CSTM_TOMCAT_THREAD_POOL_NUMBER_OF_THREADS_BUSY
Custom MBeans Attribute Delta Rules
UNIX Rules
JVM Logon Rules
Argent Guardian Ultra
Relators
Used In Which Relators?
Update Rule
%VARIABLES%
Copyright © 2019 Argent Software. All Rights Reserved.
Current Time Zone: IST
Click To Change
v5.1A-R8

Custom MBeans Attributes Rule Definition: JVM_CSTM_GC_COUNT_CONCUR_MARK_SWEEP_PRD

Save Predictor Data Only

Console Comment: *** GARBAGE COLLECTION COUNT OF CONCURRENT MARK SWEEP (CMS) COLLECTOR ***

□ Save Results To Argent Predictor

JMX Object Name: java.lang:type=GarbageCollector,name=ConcurrentMarkSweep

Attribute Name: CollectionCount

Save Performance Data

Object Name: JMX MBeans

Counter Name: Garbage Collector

Instance Name: Collection Count

Days To Keep Data: 7

☑ Rule Is Broken If Any Error Occurred

□ Post Event Even If Same Event Is Still Outstanding (Unanswered)

□ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

○ After Event Is Posted

○ After Event Is Answered

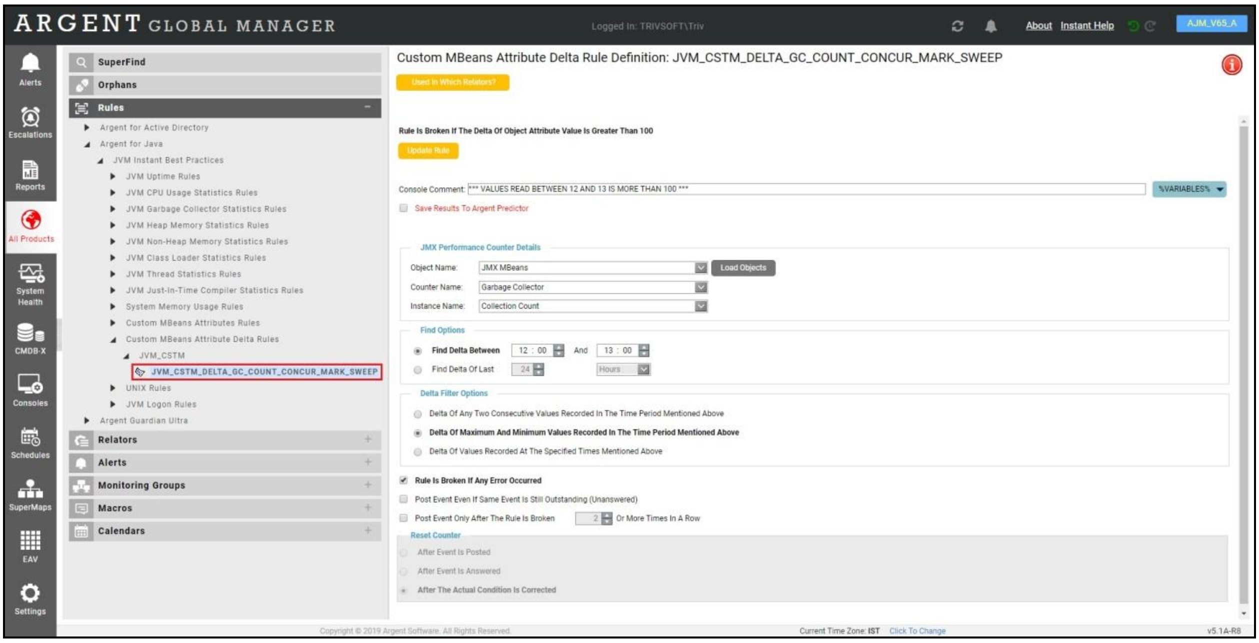
● After The Actual Condition Is Corrected

Custom MBeans Attribute Delta Rules

This Rule checks if the difference in maximum and minimum values recorded by a Custom MBeans Attribute Rule, in a specified interval of time, exceeds the threshold

Please make sure the correct combination of Object, Counter and Instance are configured, which is the replica of the corresponding Custom MBeans Attribute Rule

In this sample Rule, the Rule brakes if the difference between the maximum and minimum values recorded between12:00 and 13:00 exceeds 100


ARGENT GLOBAL MANAGER
Logged In: TRIVSOFT\Triv
About
Instant Help
AJM_V65_A
Alerts
Escalations
Reports
All Products
System Health
CMDB-X
Consoles
Schedules
SuperMaps
EAV
Settings
SuperFind
Orphans
Rules
Argent for Active Directory
Argent for Java
JVM Instant Best Practices
JVM Uptime Rules
JVM CPU Usage Statistics Rules
JVM Garbage Collector Statistics Rules
JVM Heap Memory Statistics Rules
JVM Non-Heap Memory Statistics Rules
JVM Class Loader Statistics Rules
JVM Thread Statistics Rules
JVM Just-In-Time Compiler Statistics Rules
System Memory Usage Rules
Custom MBeans Attributes Rules
Custom MBeans Attribute Delta Rules
JVM_CSTM
JVM_CSTM_DELTA_GC_COUNT_CONCUR_MARK_SWEEP
UNIX Rules
JVM Logon Rules
Argent Guardian Ultra
Relators
Alerts
Monitoring Groups
Macros
Calendars
Used In Which Relators?
Update Rule
%VARIABLES%
Load Objects

Custom MBeans Attribute Delta Rule Definition: JVM_CSTM_DELTA_GC_COUNT_CONCUR_MARK_SWEEP

Rule Is Broken If The Delta Of Object Attribute Value Is Greater Than 100

Console Comment: *** VALUES READ BETWEEN 12 AND 13 IS MORE THAN 100 ***

☐ Save Results To Argent Predictor

JMX Performance Counter Details

Object Name: JMX MBeans

Counter Name: Garbage Collector

Instance Name: Collection Count

Find Options

● Find Delta Between 12 : 00 And 13 : 00

○ Find Delta Of Last 24 Hours

Delta Filter Options

○ Delta Of Any Two Consecutive Values Recorded In The Time Period Mentioned Above

● Delta Of Maximum And Minimum Values Recorded In The Time Period Mentioned Above

○ Delta Of Values Recorded At The Specified Times Mentioned Above

☑ Rule Is Broken If Any Error Occurred

☐ Post Event Even If Same Event Is Still Outstanding (Unanswered)

☐ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

○ After Event Is Posted

○ After Event Is Answered

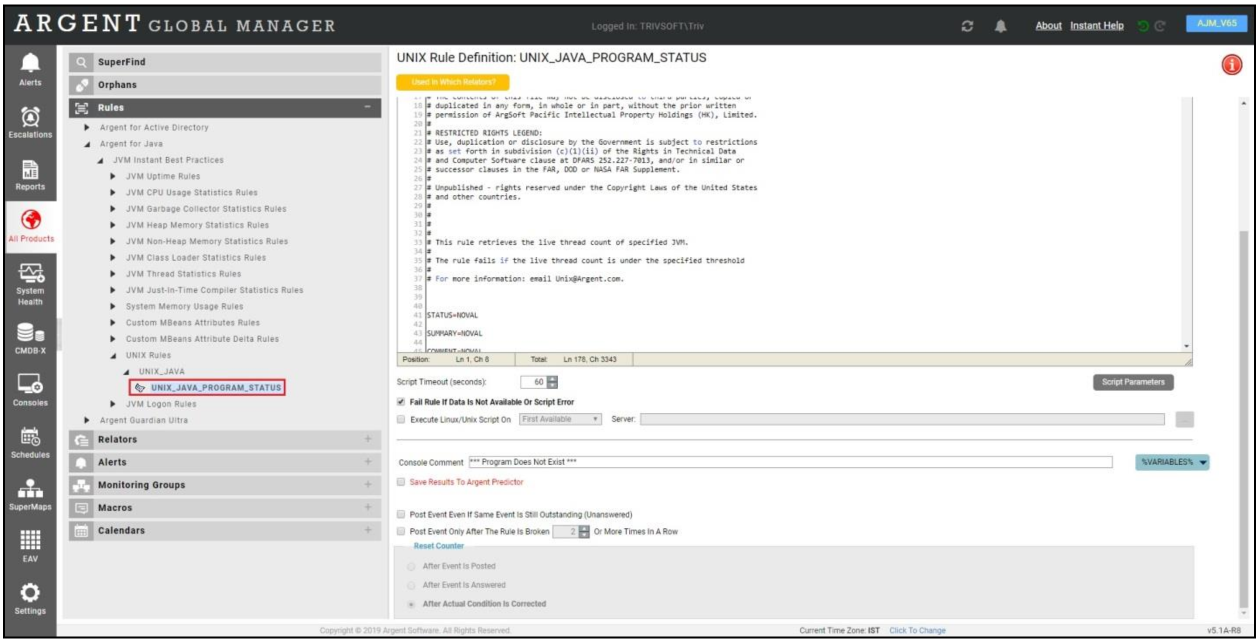
● After The Actual Condition Is Corrected

Copyright © 2019 Argent Software. All Rights Reserved. Current Time Zone: IST Click To Change v5.1A-R8

Check Appendix to know more about MBeans Attributes

UNIX Rules

This rule checks the existence of a Java program


ARGENT GLOBAL MANAGER
Logged In: TRIVSOFT\Triv
About
Instant Help
AJM_V65
Alerts
Escalations
Reports
All Products
System Health
CMDB-X
Consoles
Schedules
SuperMaps
EAV
Settings
SuperFind
Orphans
Rules
Argent for Active Directory
Argent for Java
JVM Instant Best Practices
JVM Uptime Rules
JVM CPU Usage Statistics Rules
JVM Garbage Collector Statistics Rules
JVM Heap Memory Statistics Rules
JVM Non-Heap Memory Statistics Rules
JVM Class Loader Statistics Rules
JVM Thread Statistics Rules
JVM Just-In-Time Compiler Statistics Rules
System Memory Usage Rules
Custom MBeans Attributes Rules
Custom MBeans Attribute Delta Rules
UNIX Rules
UNIX_JAVA
UNIX_JAVA_PROGRAM_STATUS
JVM Logon Rules
Argent Guardian Ultra
Relators
Alerts
Monitoring Groups
Macros
Calendars
Used In Which Relators?
Script Parameters
%VARIABLES%
Copyright © 2019 Argent Software. All Rights Reserved.
Current Time Zone: IST
Click To Change
v5.1A-R8

UNIX Rule Definition: UNIX_JAVA_PROGRAM_STATUS

```
# The contents of this file may not be disclosed to third parties, copied or
# duplicated in any form, in whole or in part, without the prior written
# permission of ArgSoft Pacific Intellectual Property Holdings (HK), Limited.
#
# RESTRICTED RIGHTS LEGEND:
# Use, duplication or disclosure by the Government is subject to restrictions
# as set forth in subdivision (c)(1)(ii) of the Rights in Technical Data
# and Computer Software clause at DFARS 252.227-7013, and/or in similar or
# successor clauses in the FAR, DOD or NASA FAR Supplement.
#
# Unpublished - rights reserved under the Copyright Laws of the United States
# and other countries.
#
#
#
#
# This rule retrieves the live thread count of specified JVM.
#
# The rule fails if the live thread count is under the specified threshold
#
# For more information: email Unix@Argent.com.



STATUS=NOVAL

SUMMARY=NOVAL

COMMENT=NOVAL
```

Position: Ln 1, Ch 8   Total: Ln 178, Ch 3343

Script Timeout (seconds): 60

☑ Fail Rule If Data Is Not Available Or Script Error

□ Execute Linux/Unix Script On First Available   Server:

Console Comment *** Program Does Not Exist ***

□ Save Results To Argent Predictor

□ Post Event Even If Same Event Is Still Outstanding (Unanswered)

□ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

○ After Event Is Posted

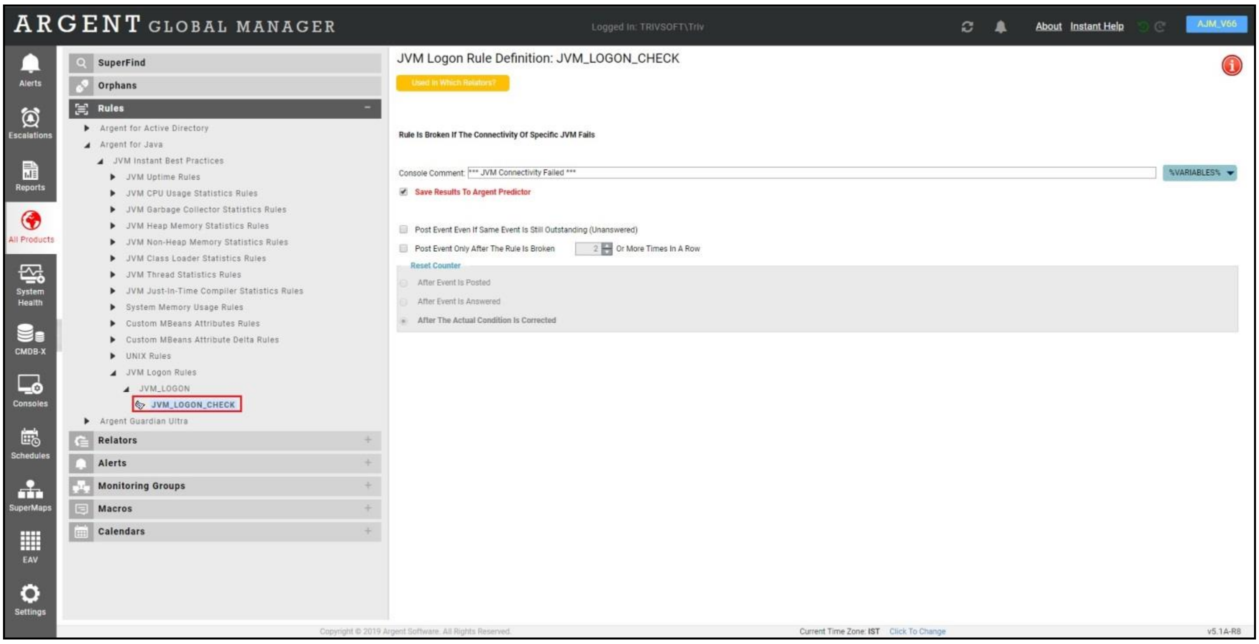
○ After Event Is Answered

● After Actual Condition Is Corrected

JVM Logon Rules

Determines if a JVM can be connected using JMX

The Logon credentials should be specified in the licensed server manager


ARGENT GLOBAL MANAGER
Logged In: TRIVSOFT\Triv
About
Instant Help
AJM_V66
Alerts
Escalations
Reports
All Products
System Health
CMDB-X
Consoles
Schedules
SuperMaps
EAV
Settings
SuperFind
Orphans
Rules
Argent for Active Directory
Argent for Java
JVM Instant Best Practices
JVM Uptime Rules
JVM CPU Usage Statistics Rules
JVM Garbage Collector Statistics Rules
JVM Heap Memory Statistics Rules
JVM Non-Heap Memory Statistics Rules
JVM Class Loader Statistics Rules
JVM Thread Statistics Rules
JVM Just-In-Time Compiler Statistics Rules
System Memory Usage Rules
Custom MBeans Attributes Rules
Custom MBeans Attribute Delta Rules
UNIX Rules
JVM Logon Rules
JVM_LOGON
JVM_LOGON_CHECK
Argent Guardian Ultra
Relators
Alerts
Monitoring Groups
Macros
Calendars
Used In Which Relators?
%VARIABLES%
Copyright © 2019 Argent Software. All Rights Reserved.
Current Time Zone: IST
Click To Change
v5.1A-R8

JVM Logon Rule Definition: JVM_LOGON_CHECK

Rule Is Broken If The Connectivity Of Specific JVM Fails

Console Comment: *** JVM Connectivity Failed ***

☑ Save Results To Argent Predictor

☐ Post Event Even If Same Event Is Still Outstanding (Unanswered)

☐ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

○ After Event Is Posted

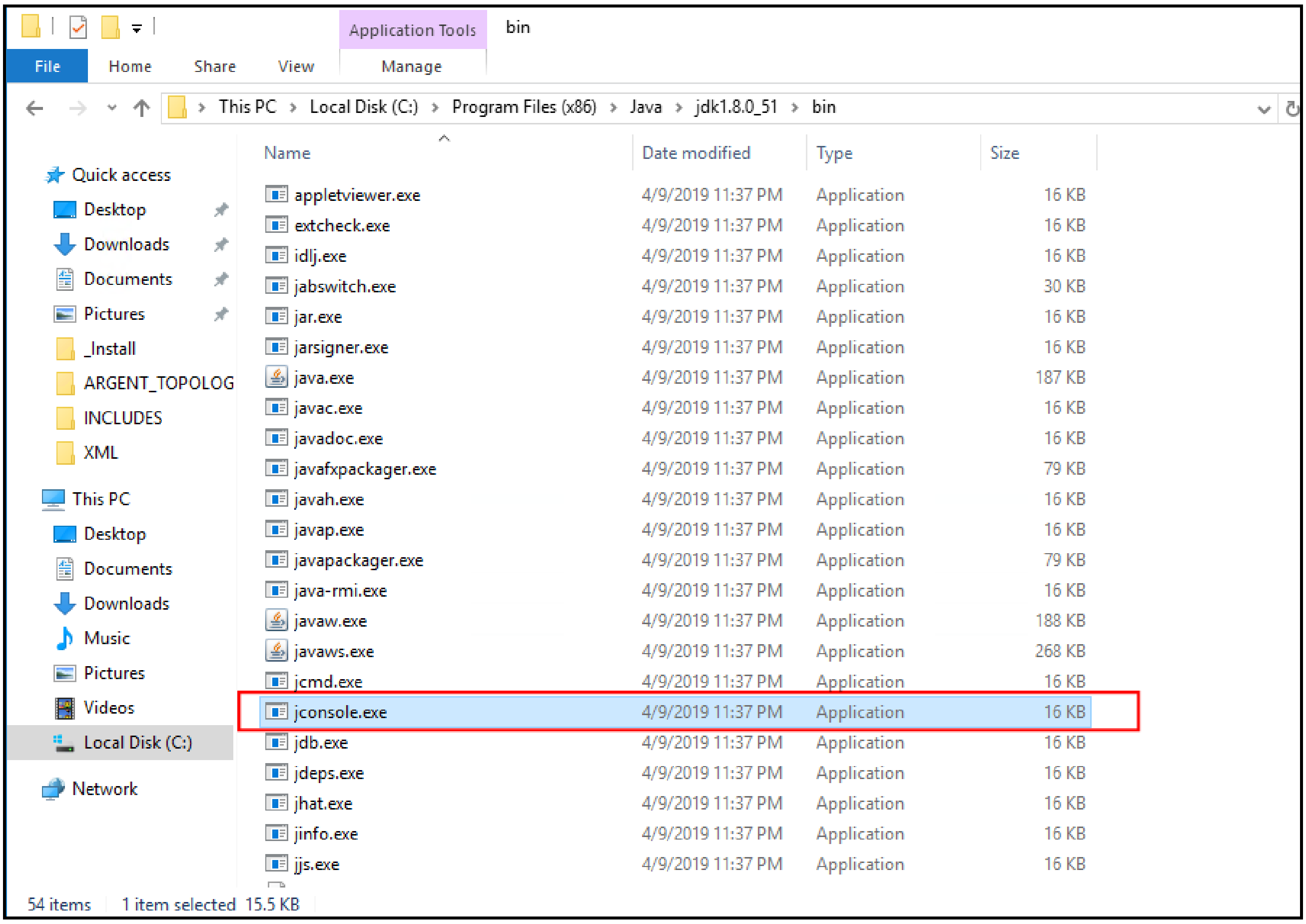
○ After Event Is Answered

● After The Actual Condition Is Corrected

Appendix

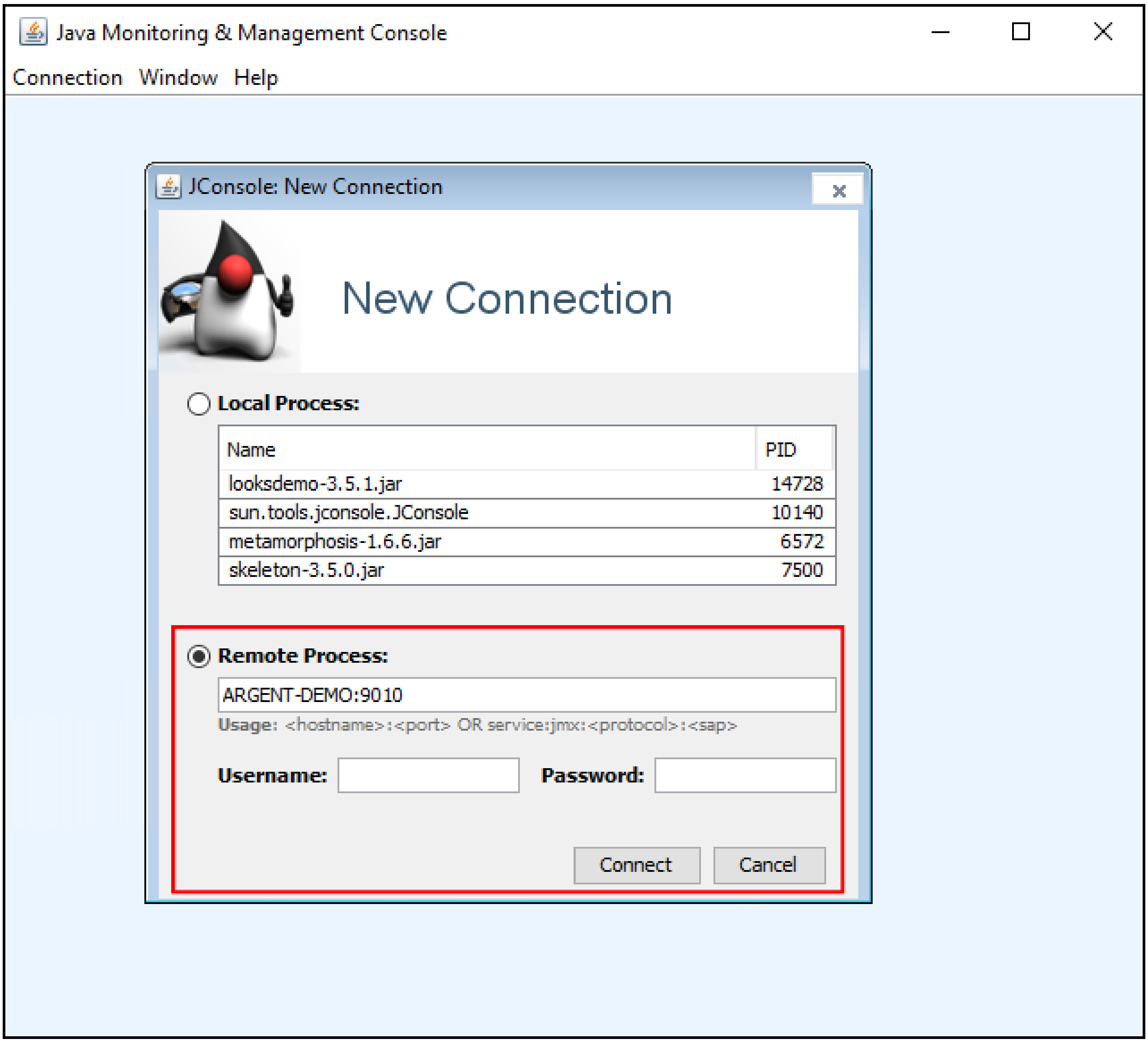
To check on the available attributes provided by the MBeans, and configure the Custom MBeans Attributes Rule, do the following

- Run the **jconsole.exe** file from the following path in the Argent Server: **{Java Installed Drive}:\Program Files (x86)\Java\jdk1.8.0_51\bin**



Name	Date modified	Type	Size
appletviewer.exe	4/9/2019 11:37 PM	Application	16 KB
extcheck.exe	4/9/2019 11:37 PM	Application	16 KB
idlj.exe	4/9/2019 11:37 PM	Application	16 KB
jabswitch.exe	4/9/2019 11:37 PM	Application	30 KB
jar.exe	4/9/2019 11:37 PM	Application	16 KB
jarsigner.exe	4/9/2019 11:37 PM	Application	16 KB
java.exe	4/9/2019 11:37 PM	Application	187 KB
javac.exe	4/9/2019 11:37 PM	Application	16 KB
javadoc.exe	4/9/2019 11:37 PM	Application	16 KB
javafxpackager.exe	4/9/2019 11:37 PM	Application	79 KB
javah.exe	4/9/2019 11:37 PM	Application	16 KB
javap.exe	4/9/2019 11:37 PM	Application	16 KB
javapackager.exe	4/9/2019 11:37 PM	Application	79 KB
java-rmi.exe	4/9/2019 11:37 PM	Application	16 KB
javaw.exe	4/9/2019 11:37 PM	Application	188 KB
javaws.exe	4/9/2019 11:37 PM	Application	268 KB
jcmd.exe	4/9/2019 11:37 PM	Application	16 KB
jconsole.exe	4/9/2019 11:37 PM	Application	16 KB
jdb.exe	4/9/2019 11:37 PM	Application	16 KB
jdeps.exe	4/9/2019 11:37 PM	Application	16 KB
jhat.exe	4/9/2019 11:37 PM	Application	16 KB
jinfo.exe	4/9/2019 11:37 PM	Application	16 KB
jjs.exe	4/9/2019 11:37 PM	Application	16 KB

- In the **Jconsole: New Connection** screen, select Rempte Process, specify the credentials and click the **Connect** button



○ **Local Process:**

Name	PID
looksdemo-3.5.1.jar	14728
sun.tools.jconsole.JConsole	10140
metamorphosis-1.6.6.jar	6572
skeleton-3.5.0.jar	7500

● **Remote Process:**

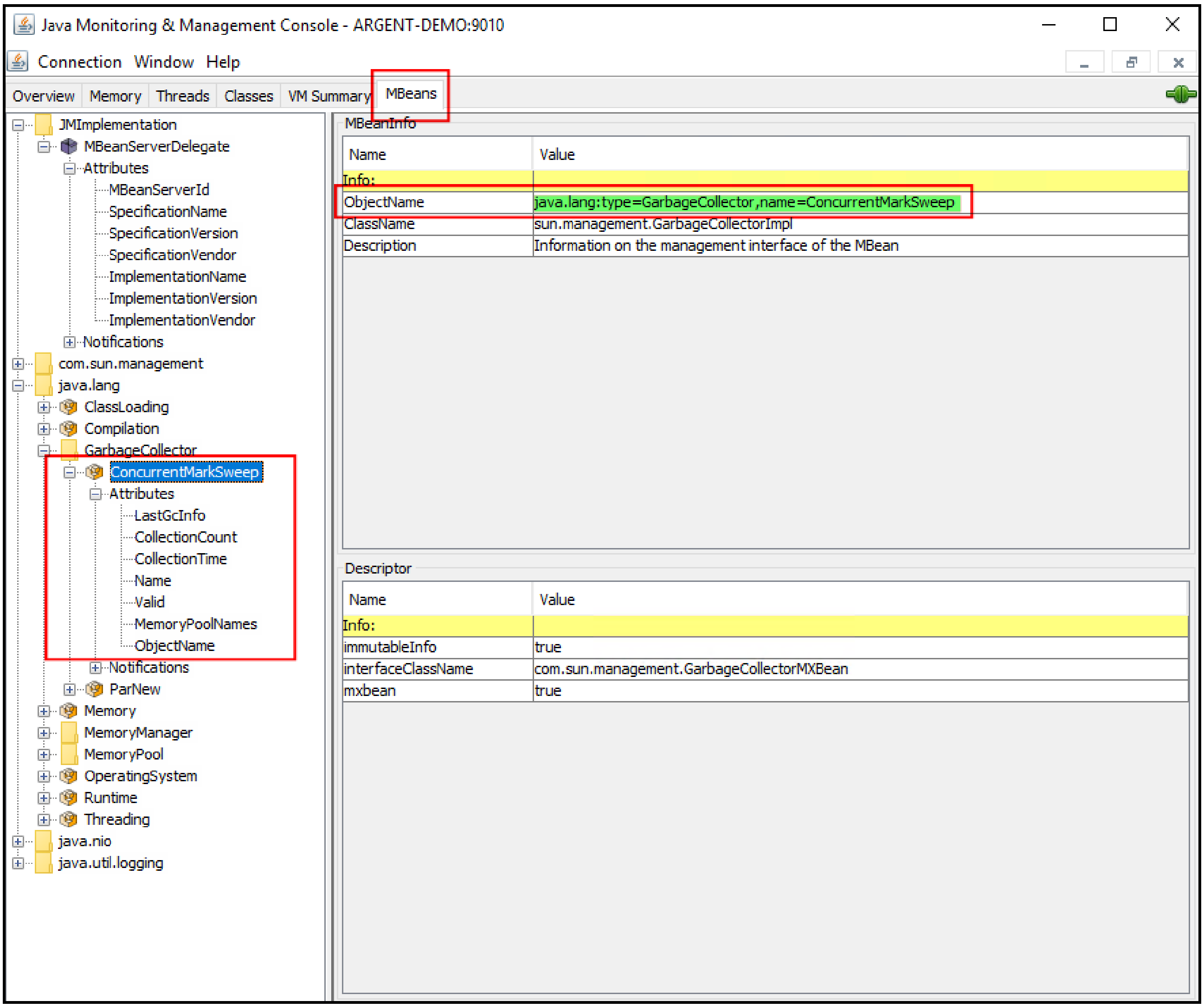
ARGENT-DEMO:9010

Usage: <hostname>:<port> OR service:jmx:<protocol>:<sap>

Username:

Password:

- In the **Java Monitoring & Management Console** screen, select **MBeans** tab, expand the tree and select an object to list the Attributes availble under it



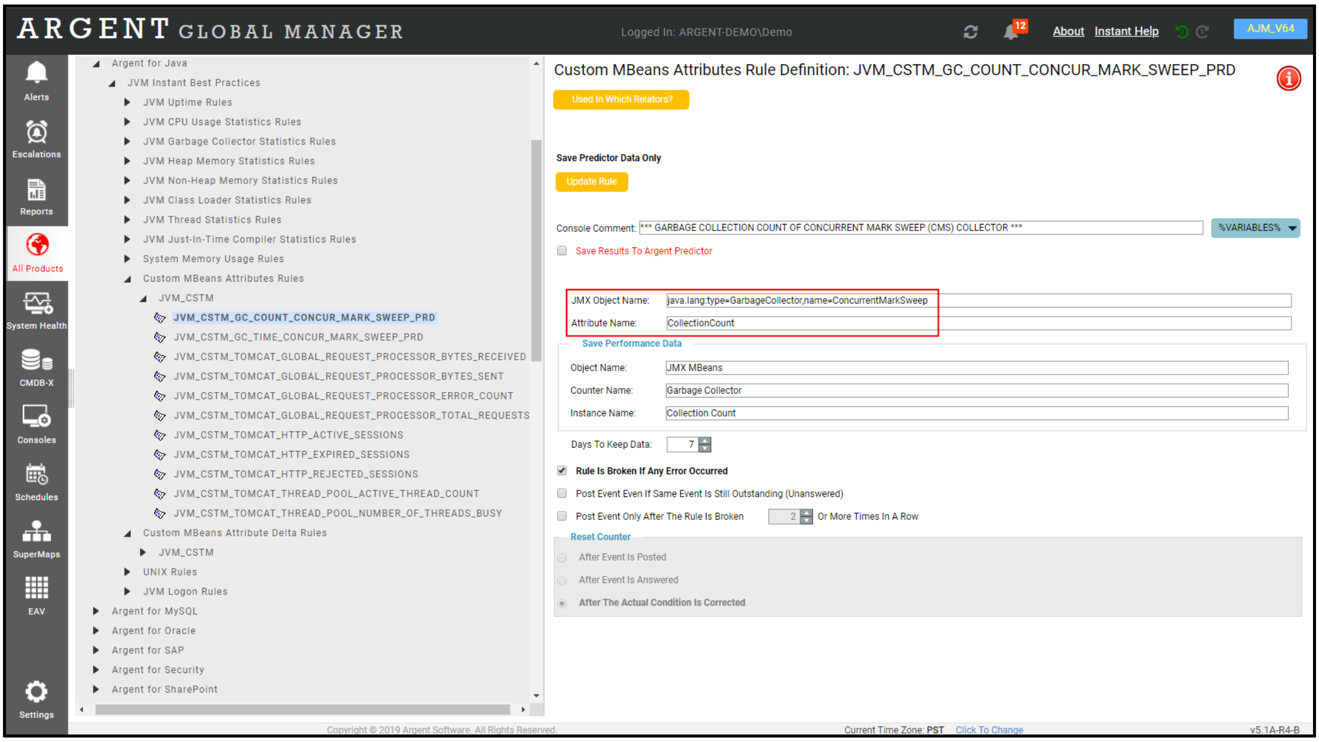
MBeanInfo

Name	Value
Info:	
ObjectName	java.lang:type=GarbageCollector,name=ConcurrentMarkSweep
ClassName	sun.management.GarbageCollectorImpl
Description	Information on the management interface of the MBean

Descriptor

Name	Value
Info:	
immutableInfo	true
interfaceClassName	com.sun.management.GarbageCollectorMXBean
mxbean	true

- Copy the Object name (highlighted in green in the above screenshot) from the screen and paste it in the **JMX Object Name** field of the **Custom MBeans Attributes Rule**. Name of the Attrinute to be monitored should be specified in the **Attribute Name** field


ARGENT GLOBAL MANAGER
Logged In: ARGENT-DEMO\Demo
12
About
Instant Help
AJM_V64
Alerts
Escalations
Reports
All Products
System Health
CMDB-X
Consoles
Schedules
SuperMaps
EAV
Settings
Argent for Java
JVM Instant Best Practices
JVM Uptime Rules
JVM CPU Usage Statistics Rules
JVM Garbage Collector Statistics Rules
JVM Heap Memory Statistics Rules
JVM Non-Heap Memory Statistics Rules
JVM Class Loader Statistics Rules
JVM Thread Statistics Rules
JVM Just-In-Time Compiler Statistics Rules
System Memory Usage Rules
Custom MBeans Attributes Rules
JVM_CSTM
JVM_CSTM_GC_COUNT_CONCUR_MARK_SWEEP_PRD
JVM_CSTM_GC_TIME_CONCUR_MARK_SWEEP_PRD
JVM_CSTM_TOMCAT_GLOBAL_REQUEST_PROCESSOR_BYTES_RECEIVED
JVM_CSTM_TOMCAT_GLOBAL_REQUEST_PROCESSOR_BYTES_SENT
JVM_CSTM_TOMCAT_GLOBAL_REQUEST_PROCESSOR_ERROR_COUNT
JVM_CSTM_TOMCAT_GLOBAL_REQUEST_PROCESSOR_TOTAL_REQUESTS
JVM_CSTM_TOMCAT_HTTP_ACTIVE_SESSIONS
JVM_CSTM_TOMCAT_HTTP_EXPIRED_SESSIONS
JVM_CSTM_TOMCAT_HTTP_REJECTED_SESSIONS
JVM_CSTM_TOMCAT_THREAD_POOL_ACTIVE_THREAD_COUNT
JVM_CSTM_TOMCAT_THREAD_POOL_NUMBER_OF_THREADS_BUSY
Custom MBeans Attribute Delta Rules
JVM_CSTM
UNIX Rules
JVM Logon Rules
Argent for MySQL
Argent for Oracle
Argent for SAP
Argent for Security
Argent for SharePoint
Used In Which Relators?
Update Rule
%VARIABLES%
Copyright © 2019 Argent Software. All Rights Reserved.
Current Time Zone: PST
Click To Change
v5.1A-R4-B

Custom MBeans Attributes Rule Definition: JVM_CSTM_GC_COUNT_CONCUR_MARK_SWEEP_PRD

Save Predictor Data Only

Console Comment: *** GARBAGE COLLECTION COUNT OF CONCURRENT MARK SWEEP (CMS) COLLECTOR ***

□ Save Results To Argent Predictor

JMX Object Name: java.lang:type=GarbageCollector,name=ConcurrentMarkSweep

Attribute Name: CollectionCount

Save Performance Data

Object Name: JMX MBeans

Counter Name: Garbage Collector

Instance Name: Collection Count

Days To Keep Data: 7

☑ Rule Is Broken If Any Error Occurred

□ Post Event Even If Same Event Is Still Outstanding (Unanswered)

□ Post Event Only After The Rule Is Broken 2 Or More Times In A Row

Reset Counter

○ After Event Is Posted

○ After Event Is Answered

● After The Actual Condition Is Corrected