# Developer's Guide

## Monitoring Alarms

12/18/2025

# Contents

# Monitoring Alarms

> ### Tip
> Genesys Co-browse Server **8.5.002** extended metrics functionality by adding monitoring alarms. You can use monitoring alarms to improve Co-browse performance

A *monitoring alarm* is an alert that signals a problem discovered in Co-browse Server. Co-browse Server produces *predefined* and *generic* monitoring alarms.

Predefined monitoring alarms include:

- Heap Memory Usage
- GC Frequency
- GC Latency
- Inactive Sessions
- Jetty Thread Pool Usage
- Server Response Time
- Agent Side Render Latency

The criteria Co-browse Server uses to detect and cancel a problem depend on the monitored metric's specified threshold.

## Thresholds

A *threshold* is the basic element used to implement all generated monitoring alarms.

Each threshold is described by the following parameters:

- JMX metric
- Threshold type, predefined or generic
- Related option in the metrics section of the Co-browse Server's configuration
- Log Event ID for `detect` event
- Log Event ID for `cancel` event

### Predefined thresholds

Alert generations use predefined thresholds when threshold parameters like `metric`, `Detect Log ID`, and `Cancel Log ID` are predefined and cannot change through configuration.

### Generic Thresholds

Generic thresholds let you dynamically set thresholds on any registered metric of type `counter`, `histogram`, or `timer`.

## Configuring Monitoring Alarm Reports

You can configure the Logging Reporter and the Message Server Reporter to report monitoring alarms.

### Logging Reporter

You can report alarms in the logging subsystem using the *logging reporter*. The logging subsystem is configured in the **log section** of Co-browse Server configuration.

All alarms that detect events are reported in log messages with level **[ERROR]** while all alarms that cancel events have level **[WARN]**.

Detection alarms come in two types:

- fatal alarms with **alarm** log level
- standard alarms with **standard** log level

Cancellation alarms correspond to a **trace** log level.

### Message Server Reporter

Starting with release **8.5.002**, Co-browse Server supports a *Message Server reporter* you can use to display alarms in the **Active Alarms** section of Genesys Administrator. By reporting alarms in **Active Alarms**, you simplify application monitoring and avoid detailed logging that can affect system performance.

### Configuring Monitoring Alarms

Alarms are log messages reported according to the configured log subsystem. To report a particular alarm in **Active Alarms**, you must configure:

- Message Server Reporter
- Alarm Condition object
- related threshold option in the server application

You can see the dependencies between Alarm Condition objects and related application server configuration options in the Co-browse Alarms Configuration Table.

> ### Important
> To apply new Alarm Condition objects, restart Solution Control Server.

## Configuring Message Server Reporter

To configure Message Server reporter, specify the following:

1. Message Server Application:

   In the **messages** section, set **db_storage** to `true`.

2. Co-browse Cluster Application:

   1. Add a connection to the Message Server application.

   2. Configure the **metrics** section:

      - Set **reporter.messageServer.enabled** to `true` (default value).

      - Set the **reporter.messageServer.logFrequency**. The default value is 30 minutes.

3. Co-browse Node application's **log** section:

   - Set the **verbose** option to `standard` for only error messages or to `trace` for error and info messages.

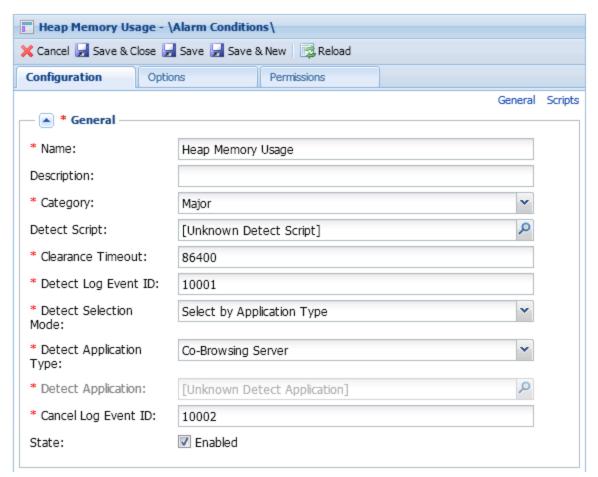   - Set the all, trace, or debug options to value `network`.

## Configuring Alarm Condition Object

Message Server reporter needs each predefined threshold to have a related **Alarm Condition** object in the Genesys Configuration.

While each predefined alarm can contain dedicated Alarm Condition object, only one Alarm Condition object is allowed for generic alerts because their Detect Log Event ID is the same.

You must manually create Alarm Condition objects in the **Alarm Conditions** section of Genesys Administrator:

Configuring an Alarm Condition Object in Genesys Administrator



1. Open the **Provisioning > Environment > Alarm Conditions** section in Genesys Administrator.

2. Click **New** to create a new object.

3. Specify a **Name**. The value can be any string.

4. Set the proper **Detect Log Event ID** and **CancelLog Event ID**, see the Co-browse alarms configuration table.

5. Set **Select by Application Type** to Detect Selection Mode.

6. Set Co-Browsing Server for **Detect Application Type**.

7. Save your changes.

> ### Important
> For *generic alarms*, you should leave the **Cancel Log Event ID** empty and set a smaller **Clearance Timeout** because generic alarms have no Cancel Log Event ID and they cannot be automatically deleted from the Active Alarms view.

Configuring the threshold option in the server configuration

Co-browse server configuration contains the following threshold options:

- Predefined
    - HeapMemoryUsage.threshold
    - GcFrequency.threshold
    - GcLatency.threshold
    - InactiveSessions.threshold
    - JettyThreadPoolUsage.threshold
    - ServerResponseTime.threshold
    - SlaveRenderLatency.threshold
- *Generic* threshold configurations use the option <metricName>.threshold.

To configure a predefined threshold set the proper value for the corresponding option.

To configure a generic threshold:

1. Substitute the metric name placeholder with the actual metric name, see Breakdown of Available Metrics.
2. Set the proper value for the metric's threshold.

Co-browse Alarms Configuration Table

| Alarm name | Alarm Condition object | | | | | Related configuration option, `metrics` section | |
|---|---|---|---|---|---|---|---|
| Threshold type | Selection mode | Application type | Detect Event ID | Cancel Event ID | Option | Default value | Description |
| Heap Memory Usage | predefined | Select by Application Type | Co-browse Server | 10001 | 10002 | HeapMemoryUsage.threshold | 0.8 | Defines heap memory usage threshold value. This is the ratio of the used heap memory to the maximum heap memory. |

| Alarm name | Alarm Condition object | | | | Related configuration option, `metrics section` | |
|---|---|---|---|---|---|---|
| GC Frequency | | | | 10003 | 10004 | GcFrequency.threshold 24 | Defines GC frequency threshold value for an hour. |
| GC Latency | | | | 10005 | 10006 | GcLatency.threshold 1000 | Defines GC Latency threshold value, in milliseconds, in relation to the last GC occurred in the configured time interval. |
| Inactive Sessions | | | | 100001 | 100002 | InactiveSession.threshold 0.2 | Defines the ratio of inactive sessions to all sessions from the configured time interval. It shows how many Co-browse sessions were created by customer but never joined by an agent. |
| Slave Render Latency | | | | 100003 | 100004 | SlaveRenderLatency.threshold 10000 | Defines, in milliseconds, the SlaveRenderLatency metric threshold |

| Alarm name | Alarm Condition object | | | | Related configuration option, `metrics section` | |
|---|---|---|---|---|---|---|
| | | | | | | value in the configured time interval. Agent side rendering latency shows whether reported agent side rendering is too slow. |
| Jetty Thread Pool Usage | | 100005 | 100006 | JettyThreadPoolUsage.threshold | 0.9 | Defines Jetty thread pool usage threshold value. This is the ratio of the used Jetty thread pool size to the maximum available. It signals whether too few free threads handle http requests. |
| Server Response Time | | 100007 | 100008 | ServerResponseTime.threshold | 100 | Defines, in milliseconds, the maximum value allowed for ServerResponseTime metric. The metric is |

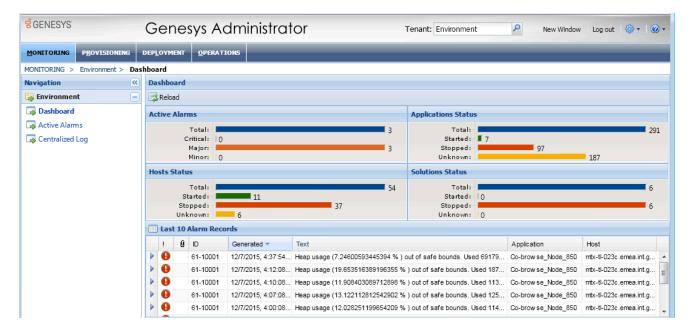| Alarm name | Alarm Condition object | | | | Related configuration option, `metrics` section | | |
|---|---|---|---|---|---|---|---|
| | | | | | | | calculated as average time for the latest N routings of data from customer to agent, where N is defined by the ServerResponseTime.slic option value. |
| | | | | | ServerResponseTime.slidingWindowSize | 100 | Defines the number of recent measurements applied for the ServerResponseTime metric calculation. |
| Generic alarm | generic | | | 10007 | Generic threshold option | | Defines threshold value for the particular metric. |

# Using Alarms to Improve Co-browse Performance
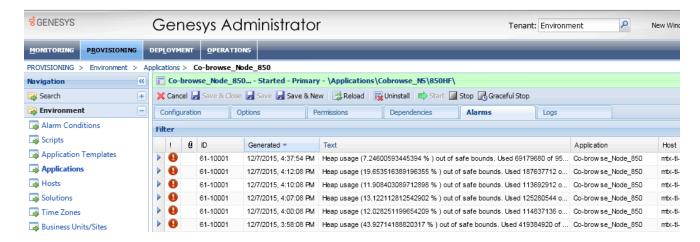
## Co-browse Alarm Reporting

Once you configure your Co-browse alarm reporting, you can monitor your Co-browse Server:

You can observe all monitoring alarms in the **Active Alarms** section of Genesys Administrator.



You can observe fatal alarms in the Genesys Administrator **Dashboard**.

You can also observe fatal alarms in the **Alarms** tab of your *Co-browse node's* application properties.

## Responding to Co-browse Alarms

Monitoring alarms detect problems in your application server. The table below lists possible actions to resolve problems detected.

Once you fix a problem, the server recalculates the metric after the monitoring time interval and deletes the alarm from the alarm monitoring view. At the same time, the appropriate message appears in the log and states that the metric value is back to normal.

Actions to Respond to Common Alarms

| Alarm name | Fatal? | Detect alarm message example | Problem description | Actions to fix the problem | Cancel alarm message |
|---|---|---|---|---|---|
| Heap Memory Usage | yes | [ERROR] HeapUsageThreshold - Heap usage (40.65 %) out of safe bounds. Used 388140568 of 954728448 bytes. | This alarm signals that Co-browse Server is working but at full capacity. | To prevent the application from overloading, you should extent the memory heap:<br><br>1. Open setenv.bat (Windows) or setenv.sh (UNIX) for editing.<br><br>2. Increase **Xmx\*** value in the **JAVA_OPTS** directive:<br><br>set JAVA_OPTS=%JAVA_OPTS | [ WARN] HeapUsageThreshold - Heap usage (30.05 %) is back to normal |

| Alarm name | Fatal? | Detect alarm message example | Problem description | Actions to fix the problem | Cancel alarm message |
|---|---|---|---|---|---|
|  |  |  |  | % ...<br><br>-Xmx1024m<br><br>...<br><br>3. Restart the Co-browse Server application. |  |
| GC Frequency | no | [ERROR] GcFrequencyThreshold - Garbage collection frequency (24,4718 per hour) is out of bounds (24,000000 per hour). | There might be several causes:<br><br>• The heap memory size is less than needed<br><br>• Too many created entities. It might happen due to log message overloading<br><br>• If this problem happened while the log level is high, the reason might be hyperactivity of sessions while the memory heap is small. | 1. You should increase heap size as described above.<br><br>2. Increasing the log level may resolve the problem.<br><br>If these solutions do not help, you should add key **Xmn\*** in the **JAVA_OPTS** directive in setenv.bat/sh file. | [ WARN] GcFrequencyThreshold - Garbage collection frequency (20.6773 per hour) is back to normal |
| GC Latency | no | [ERROR] GcLatencyThreshold - Garbage collection latency (<number> milliseconds) is out of the defined bounds (<number> milliseconds). | This alarm means that the GC processor is overloaded. | To resolve the problem, you should remove excessive load by either:<br><br>• replacing existent processor with a more | [ WARN] GcLatencyThreshold - Garbage collection latency (251 milliseconds) is back to normal |

| Alarm name | Fatal? | Detect alarm message example | Problem description | Actions to fix the problem | Cancel alarm message |
|---|---|---|---|---|---|
| | | | | powerful one<br><br>• replacing existent RAM with faster RAM<br><br>• doing both. | |
| Inactive Sessions | no | [ERROR] InactiveSessionsThreshold - Percent of inactive sessions 0,25 out of bounds 0,20. 10 are inactive from 40 | Shows how many Co-browse sessions were created but never joined by an agent. | | [ WARN] InactiveSessionsThreshold - Percent of inactive sessions is back to normal |
| Slave Render Latency | no | [ERROR] SlaveRenderLatencyThreshold - Average time of agent side rendering (14730,0 milliseconds) is out of bounds (10000 milliseconds) | This alarm alerts that the reported agent side rendering is too slow. | | [ WARN] SlaveRenderLatencyThreshold - Average time of agent side rendering is back to normal |
| Jetty Thread Pool Usage | no | [ERROR] JettyThreadPoolUsageThreshold - Jetty thread pool usage (0,06) is out of bounds (0,001). 11 busy threads from 200 | Too few free threads allowed to handle http requests. | | [ WARN] JettyThreadPoolUsageThreshol - Jetty thread pool usage is back to normal |
| Server Responce Time | no | [ERROR] ServerResponseTimeThreshold - Average response time (68,68280 milliseconds) is out o f bounds (0,50000 milliseconds) | Co-browse Server may have exceeded the threshold because:<br><br>• Co-browse Server is overloaded<br><br>• The disk is working very slow | 1. See the solution for the Heap Memory Usage alarm<br><br>2. Replace the disk with a more powerful one | [ WARN] ServerResponseTimeThreshold - Average response time is back to normal |

> **Important**
>
> To properly use the **Xmx**, **Xmn**, and **Xms** java options consult the Oracle documentation.

## Localizing Co-browse Alarms

You can localize alarm log messages using LMS files. You can have two types of LMS files, an LMS file that includes common log messages and a project specific LMS file. Default LMS files are embedded into the Co-browse Server code.

To change the log message text, use the custom LMS files shipped with the product in the **<Co-browse Server root>/server/launcher** directory:

- **GeneralAlarms_en.lms** is a common LMS file
- **CobrowseAlarms_en.lms** contains project-specific log messages

To add localization to your monitoring alarms, apply the following to each custom LMS file:

1. Copy the content of the file to a new file name which ends with a system locale abbreviation. For example, au for Australia and `fr` for France. The common LMS file name for Australia would be `GeneralAlarms_au.lms`.

2. Edit the new file to change the log message text. Save your changes.

3. After you have finished editing each custom LMS file, restart the application server.

> **Important**
>
> To avoid inconsistency in alarm logging, the only thing you can change in a custom LMS file is the log message text.