Extensions and Custom Metrics

On this page:

Related pages:

- Build a Monitoring Extension Using Scripts
- Build a Monitoring Extension Using Java
- Standalone Machine Agent HTTP Listener

Watch the video:

Standalone Machine Agent - Report Metrics from Extensions

Using the Standalone Machine Agent, you can supplement the existing metrics in the AppDynamics Controller UI with your own custom metrics. There are many extensions currently available on the AppSphere Community site. Some are created by AppDynamics and some have been created by users.

Like built-in metrics, your custom metrics are subject to the following AppDynamics features:

- automatic baselines and anomaly detection
- availability for display on custom dashboards
- availability for use in policies
- visibility of all metrics in the Metric Browser and on the Infrastructure tab, where you can display external metrics along with AppDynamics metrics on the same graph

Add New Custom Metrics

To create custom metrics, you create a monitoring extension. In your extension, you define the name and path of your metric (where it appears in the metric browser tree), what type of metric it is (sum, average, and so on), and how the data for the metric should be rolled up as it ages. One agent can run many extensions, although you may need to increase the amount of memory for the agent JVM if you do. See the section "*JVM Memory Requirements* in Standalone Machine Agent Requirements and Supported Environments for details. You can have multiple copies of the same extension, if they are in different directories.

A custom metric can be common across nodes or associated with a specific tier. When you create a metric, you specify the path in which it will appear in the metric tree. To make a common custom metric, use the root tree path Custom Metrics in your metric declaration. To make a tier-specific metric, specify the metric path associated with that component. For details, see the topics on creating Java or script-based custom metrics listed below.

> Once you establish a metric, you can not delete the metric without access to the Controller database (on-prem) and cannot delete it at all (on SAAS). Therefore, when you want to add new custom metrics to your applications, try it first in one application in a test environment to confirm that everything is appearing satisfactorily. Then, when you are ready you can move to a production environment.

Types of Monitoring Extensions

You can implement custom metrics using the following mechanisms.

- **Using a script**: You can write a shell script (Linux and Unix-like systems) or batch file (Windows) to report custom metrics every minute to the Standalone Machine Agent. The Standalone Machine Agent passes these metrics on to the Controller. For more information, see Build a Monitoring Extension Using Scripts.
- **Using Java**: Your custom metrics may be too complicated to collect using a script. For example, you may need to perform complex calculations or call a third party API to get the metrics. In this case you can extend the JavaServersMonitor class to collect the metrics and report them to the Standalone Machine Agent. Your Java program extends the JavaServersMonitor class to provide your custom functionality.
  See Build a Monitoring Extension Using Java.
- **Using HTTP**: If you enable the agent HTTP listener, you can post HTTP requests to the Standalone Machine Agent to send it custom metrics every minute. This is done by starting the Standalone Machine Agent with a Jetty HTTP listener.
  See Standalone Machine Agent HTTP Listener for information on starting the HTTP listener and sending it metrics.