

# Monitoring Azure Databricks

03/26/2019 • 2 minutes to read • Contributors 

## In this article

### [Next steps](#)

[Azure Databricks](#) is a fast, powerful [Apache Spark](#)-based analytics service that makes it easy to rapidly develop and deploy big data analytics and artificial intelligence (AI) solutions. Many users take advantage of the simplicity of notebooks in their Azure Databricks solutions. For users that require more robust computing options, Azure Databricks supports the distributed execution of custom application code.

Monitoring is a critical part of any production-level solution, and Azure Databricks offers robust functionality for monitoring custom application metrics, streaming query events, and application log messages. Azure Databricks can send this monitoring data to different logging services.

The following articles show how to send monitoring data from Azure Databricks to [Azure Monitor](#), the monitoring data platform for Azure. You should follow these articles in order.

1. [Configure Azure Databricks to send metrics to Azure Monitor](#)
2. [Send Azure Databricks application logs to Azure Monitor](#)
3. [Use dashboards to visualize Azure Databricks metrics](#)

The code library that accompanies these articles extends the core monitoring functionality of Azure Databricks to send Spark metrics, events, and logging information to Azure Monitor.

The audience for these articles and the accompanying code library are Apache Spark and Azure Databricks solution developers. The code must be built into Java Archive (JAR) files and then deployed to an Azure Databricks cluster. The code is a combination of [Scala](#) and Java, with a corresponding set of [Maven](#) project object model (POM) files to build the output JAR files. Understanding of Java, Scala, and Maven are recommended as prerequisites.

## Next steps

Start by building the code library and deploying it to your Azure Databricks cluster.

[Configure Azure Databricks to send metrics to Azure Monitor](#)