Apache Spark 3

Apache Spark Overview

Date of Publish: 2018-04-01

http://docs.hortonworks.com

Contents

Analyzing Data with Apache Spark

Apache Spark is a distributed, in-memory data processing engine designed for large-scale data processing and analytics.

Deep integration of Spark with YARN allows Spark to operate as a cluster tenant alongside Apache engines such as Hive, Storm, and HBase, all running simultaneously on a single data platform. Instead of creating and managing a set of dedicated clusters for Spark applications, you can store data in a single location, access and analyze it with multiple processing engines, and leverage your resources.

Spark on YARN leverages YARN services for resource allocation, runs Spark executors in YARN containers, and supports workload management and Kerberos security features. It has two modes:

- YARN-cluster mode, optimized for long-running production jobs
- YARN-client mode, best for interactive use such as prototyping, testing, and debugging

Spark shell and the Spark Thrift server run in YARN-client mode only.

HDP supports Spark, Livy for local and remote access to Spark through the Livy REST API, and Apache Zeppelin for browser-based notebook access to Spark. The Spark LLAP connector is not supported.

Table 1: Spark and Livy Feature Support by HDP Version

HDP Version(s)	**3.0**	**2.6.5**	**2.6.4**	**2.6.3**	**2.6.2**	**2.6.1**
Spark Version	**2.3.0**	**1.6.3, 2.3.0**	**1.6.3, 2.2.0**	**1.6.3, 2.2.0**	**1.6.3, 2.1.1**	**1.6.3, 2.1.1**
Support for Livy	0.5	0.3	0.3	0.3	0.3	0.3
Support for Hive	1.2.1	1.2.1	1.2.1	1.2.1	1.2.1	1.2.1
Spark Core	#	#	#	#	#	#
Spark on YARN	#	#	#	#	#	#
Spark on YARN for Kerberos-enabled clusters	#	#	#	#	#	#
Spark history server	#	#	#	#	#	#
Spark MLlib	#	#	#	#	#	#
ML Pipeline API	#	#	#	#	#	#
DataFrame API	#	#	#	#	#	#
Optimized Row Columnar (ORC) Files	#	#	#	#	#	#
PySpark	#	#	#	#	#	#
SparkR	#	#	#	#	#	#
Spark SQL	#	#	#	#	#	#
Spark SQL Thrift server for JDBC, ODBC access	1	#	#	#	#	#
Spark Streaming	#	#	#	#	#	#
Structured Streaming	2	TP	TP	TP		
Dynamic resource allocation	#*	#*	#*	#*	#*	#*
HBase connector	#	#	#	#	#	#
GraphX	TP	TP	TP	TP	TP	TP
DataSet API	TP	TP	TP	TP	TP	TP

* Dynamic Resource Allocation does not work with Spark Streaming.

1 Spark Thrift server for Spark 2 access from Hive beeline client is not supported in HDP-3.0. You can use the Spark version of the beeline client under /usr/hdp/current/spark2-client/bin/beeline. The JDBC interpreter to Spark Thrift server from Zeppelin is not supported.

2 Structured Streaming is supported in HDP-3.0+, with the exception of continuous processing. Continuous processing is an experimental streaming execution mode that is not currently supported.

TP: Technical Preview. Technical previews are considered under development. Do not use these features in production systems. If you have questions regarding these features, contact Support through the Hortonworks Support Portal, https://support.hortonworks.com.

The following features and associated tools are not officially supported by Hortonworks:

- Spark Standalone
- Spark on Mesos
- Jupyter Notebook (formerly IPython)