Azure Install Base Configure Platform for HDI

The Trifacta platform can be configured to integrate with supported versions of HDInsight clusters to run jobs in Spark.

> **NOTE:** Before you attempt to integrate, you should review the limitations around this integration. For more information, see *Configure for HDInsight*.

Specify running environment options:

1. You can apply this change through the *Admin Settings Page* (recommended) or

   `trifacta-conf.json`

   . For more information, see *Platform Configuration Methods*.

2. Configure the following parameters to enable job execution on the specified HDI cluster:

```
"webapp.runInDatabricks": false,
"webapp.runWithSparkSubmit": true,
```

Parameter	Description
webapp.runInDatabricks	Defines if the platform runs jobs in Azure Databricks. Set this value to `false`.
webapp.runWithSparkSubmit	For HDI deployments, this value should be set to `true`.

Specify Trifacta user:

Set the Hadoop username for the Trifacta platform to use for executing jobs [`hadoop.user` (default=`trifacta`)] :

```
"hdfs.username": "[hadoop.user]",
```

Specify location of client distribution bundle JAR:

The Trifacta platform ships with client bundles supporting a number of major Hadoop distributions. You must configure the jarfile for the distribution to use. These distributions are stored in the following directory:

`/trifacta/hadoop-deps`

Configure the bundle distribution property (`hadoopBundleJar`):

```
  "hadoopBundleJar": "hadoop-deps/hdp-2.6/build/libs/hdp-2.6-bundle.jar"
```

Configure component settings:

For each of the following components, please explicitly set the following settings.

1. You can apply this change through the *Admin Settings Page* (recommended) or

   `trifacta-conf.json`

   . For more information, see *Platform Configuration Methods*.

2. Configure Batch Job Runner:

```
  "batch-job-runner": {
   "autoRestart": true,
    ...
    "classpath": "%(topOfTree)s/services/batch-job-runner/build/install/batch-job-runner/batch-job-
runner.jar:%(topOfTree)s/services/batch-job-runner/build/install/batch-job-runner/lib/*:%(topOfTree)s/%
(hadoopBundleJar)s:/etc/hadoop/conf:%(topOfTree)s/conf/hadoop-site:/usr/lib/hdinsight-datalake/*:/usr
/hdp/current/hadoop-client/client/*:/usr/hdp/current/hadoop-client/*"
  },
```

3. Configure the following environment variables:

```
"env.PATH": "${HOME}/bin:$PATH:/usr/local/bin:/usr/lib/zookeeper/bin",
"env.TRIFACTA_CONF": "/opt/trifacta/conf"
"env.JAVA_HOME": "/usr/lib/jvm/java-1.8.0-openjdk-amd64",
```

4. Configure the following properties for various Trifacta components:

```
  "ml-service": {
   "autoRestart": true
  },
  "monitor": {
   "autoRestart": true,
    ...
   "port": <your_cluster_monitor_port>
  },
  "proxy": {
   "autoRestart": true
  },
  "udf-service": {
   "autoRestart": true
  },
  "webapp": {
    "autoRestart": true
  },
```

5. Disable S3 access:

```
"aws.s3.enabled": false,
```

6. Configure the following Spark Job Service properties:

```
"spark-job-service.classpath": "%(topOfTree)s/services/spark-job-server/server/build/install/server/lib
/*:%(topOfTree)s/conf/hadoop-site/:%(topOfTree)s/services/spark-job-server/build/bundle/*:/usr/hdp
/current/hadoop-client/hadoop-azure.jar:/usr/hdp/current/hadoop-client/lib/azure-storage-2.2.0.jar",
"spark-job-service.env.SPARK_DIST_CLASSPATH": "/usr/hdp/current/hadoop-client/*:/usr/hdp/current
/hadoop-mapreduce-client/*",
```

7. Save your changes.