How to Configure Big Data Management 10.0 Update 1 for HDInsight 3.3

Abstract

Enable Big Data Management to run mappings on a Hadoop cluster on Microsoft Azure HDInsight.

Supported Versions

- Informatica Big Data Edition 10.0 Update 1

Table of Contents

Overview

Informatica Big Data Management 10.0 Update 1 adds support for Microsoft Azure HDInsight.

Before You Download and Install

Choose from among the following formats:

- Use Big Data Management in a Hadoop Environment
- Use Big Data Management in the Azure Cloud

Use Big Data Management in a Hadoop Environment

The Informatica Big Data Management installation is distributed to the Hadoop cluster as a Debian installation package. The Debian package includes the Informatica 10.0 engine, the Blaze engine, and the adapter for HDinsight. The Debian package and the binary files that you need to run the Big Data Management installation are compressed into a tar.gz file.

This guide primarily leads you through the tasks to implement Big Data Management in an on-premise Hadoop cluster environment.

To install Big Data Management on premise and enable support for Microsoft Azure HDInsight, perform the following tasks:

1. Perform installation tasks.
2. Perform post-installation tasks.
3. Update Hadoop cluster properties on the Data Integration Service Machine for MapReduce 2.
4. Configure mappings to run on the Hadoop cluster.
5. Update cluster configuration settings.

To begin, go to "Step 1. Installation" on page 4.

Use Big Data Management in the Azure Cloud

To use Big Data Management in the Azure cloud, perform the following steps:

1. Verify prerequisites.
2. Configure Big Data Management on the HDInsight cluster.
3. Configure and start Informatica services.

To see how to perform these tasks, skip to "Configuring Big Data Management in the Azure Cloud Environment" on page 32.

Step 1. Installation

To enable Azure HDInsight for Informatica 10.0 Update 1 in your cluster environment, perform the following steps:

1. Verify prerequisites.
2. Download and install EBF 17186.
   EBF 17186 enables you to install Big Data Management in an HDInsight environment. The EBF archive contains a Debian installer package that you can use to install the update in one of the following ways:
   - Install in a single node environment.
   - Install using the SCP protocol.
   - Install using another protocol.
   - Install in a cluster environment.
3. Download and install EBF 17167.
   EBF 17167 contains important updates to Big Data Management.

Verify Prerequisites

Before you download and apply the Big Data Management EBF, complete the following prerequisites, verify that you have the following environment:

- You can access a running Informatica domain that includes a Model Repository Service and a Data Integration Service.
- You have the Developer client installed on a machine in your cluster.
- Informatica Big Data Management 10.1 Debian packages are installed on your Hadoop cluster.

Download and Uncompress the Debian Package

In this task, you download and uncompress the EBF18186 archive file and use it to install Big Data Management in the cluster environment.

1. Open a browser.
2. In the address field, enter the following URL: https://tsftp.informatica.com.
3. Click **Browse** and navigate to the following directory: `/updates/Informatica10/10.0.0/EBF17186/`.
4. Download the following file to a temporary folder: `EBF17186.Linux64-X86.tar.gz`.
5. Extract the file to the machine from where you want to distribute the Debian package and run the Big Data Management installation.
6. Copy the following package to a shared directory based on the transfer protocol you are using: `InformaticaHadoop-<InformaticaForHadoopVersion>.deb`.

   For example,

   - HTTP: /var/www/html
   - FTP: /var/ftp/pub
   - NFS: <Shared location on the node>

     The file location must be accessible by all the nodes in the cluster.

   Note: The Debian package must be stored on a local disk and not on HDFS.

Installing Big Data Management in a Single Node Environment

You can install Big Data Management in a single node environment.

1. Log in to the machine.
2. Run the following command from the Big Data Management root directory to start the installation in console mode:

   ```
   sudo bash InformaticaHadoopInstall.sh
   ```
3. Press **y** to accept the Big Data Management terms of agreement.
4. Press **Enter**.
5. Press **1** to install Big Data Management in a single node environment.
6. Press **Enter**.

To get more information about the tasks performed by the installer, you can view the `informatica-hadoop-install.<DateTimeStamp>.log` installation log file.

Installing Big Data Management Using the SCP Protocol

You can install Big Data Management in a cluster environment from the primary namenode using the SCP protocol.

1. Log in to the primary namenode.
2. Run the following command to start the Big Data Management installation in console mode:

   ```
   sudo bash InformaticaHadoopInstall.sh
   ```
3. Press **y** to accept the Big Data Management terms of agreement.
4. Press **Enter**.
5. Press **2** to install Big Data Management in a cluster environment.
6. Press **Enter**.
7. Press **1** to install Big Data Management from the primary namenode.
8. Press **Enter**.

The installer installs Big Data Management in the HDInsight Hadoop cluster. The SCP utility copies the product binaries to every node on the cluster in the following directory: `/opt/Informatica`.

You can view the `informatica-hadoop-install.<DateTimeStamp>.log` installation log file to get more information about the tasks performed by the installer.

Installing Big Data Management Using NFS

You can install Big Data Management in a cluster environment from the primary NameNode using the NFS protocol.

1. Log in to the primary NameNode.
2. Run the following command to start the Big Data Management installation in console mode:

   ```
   sudo bash InformaticaHadoopInstall.sh
   ```
3. Press **y** to accept the Big Data Management terms of agreement.
4. Press **Enter**.
5. Press **2** to install Big Data Management in a cluster environment.
6. Press **Enter**.
7. Press **1** to install Big Data Management from the primary NameNode.
8. Press **Enter**.

You can view the `informatica-hadoop-install.<DateTimeStamp>.log` installation log file to get more information about the tasks performed by the installer.

Installing Big Data Management in a Cluster Environment

You can install Big Data Management in a cluster environment from any machine in the cluster that is not a name node.

1. Verify that the Big Data Management administrator has user root privileges on the node that will be running the Big Data Management installation.
2. Log in to the machine as the root user.
3. In the `HadoopDataNodes` file, add the IP addresses or machine host names of the nodes in the Hadoop cluster on which you want to install Big Data Management. The `HadoopDataNodes` file is located on the node from where you want to launch the Big Data Management installation. You must add one IP addresses or machine host names of the nodes in the Hadoop cluster for each line in the file.
4. Run the following command to start the Big Data Management installation in console mode:

   ```
   sudo bash InformaticaHadoopInstall.sh
   ```
5. Press **y** to accept the Big Data Management terms of agreement.
6. Press **Enter**.
7. Press **2** to install Big Data Management in a cluster environment.
8. Press **Enter**.

Download and Install EBF 17167

After you install the HDInsight Debian package from EBF 17186, download and install EBF 17167 to get important Big Data Management updates.

To see how to apply EBF 17167, see Knowledge Base article 284396 at the following location: https://kb.informatica.com/howto/6/Pages/15/284396.aspx.

Step 2. Post-Installation Tasks

After you install Big Data Management, perform the post-installation tasks to ensure that Big Data Management runs properly.

1. Populate the HDFS file system.
2. Configure settings on the Informatica domain.
3. If HBase is not already installed, install it.
4. Configure the Hadoop pushdown properties for the Data Integration Service.
5. Configure the Blaze engine log directories.
6. Configure environment variables in the Big Data Management properties file.
7. Edit Informatica Developer files and variables.
8. Open the required ports for the Blaze engine.
9. Enable the Blaze engine console.
10. Configure Hive CLI or HiveServer2 to run mappings.

Populate the HDFS File System

After you install Big Data Management, populate the HDFS file system.

Informatica supports read/write from the local HDFS location, but not the wasb location. In an HDInsight cluster, the default environment has a local HDFS location that is empty, and a wasb location populated with files. Perform the following steps to copy files from the wasb location to the local HDFS location:

1. Use the Ambari configuration tool to identify the wasb location and the HDFS location.

   You can find these locations as follows:
   wasb location

   The wasb location is a resource locator like:

   ```
   wasb://<cluster_name>@<domain_or_IP-address>/
   ```

   HDFS location

   The HDFS location is a resource locator like:

   ```
   hdfs://<headnode_IP_address>:<port_number>/
   ```

   Note: Make a note of these locations. You might need them later, during the configuration process, to define property values.

2. Copy the folder `/hdp/apps/<CurrentVersion>/` and all its contents from the `wasb` location to the `hdfs` location, with the same folder structure.

   The HDFS file system is populated.

3. Set the value of the fs.defaultFS property to the HDFS location.

   After you restart, the cluster will populate files in the HDFS location.
   Optionally, you can change the value of fs.defaultFS to the wasb location, and restart the affected components again.

   Note: If it is undesirable to restart cluster components, you can manually copy files from the wasb location to the local HDFS location.

Configure Settings on the Informatica Domain

After you install Big Data Management, edit the hdfs-site.xml file to support access to files in the HDInsight wasb location.

- To change the value of the fs.DefaultFS property to the wasb location, edit the following file:
  `<Informatica_installation_directory>/services/shared/<hadoop_distribution>/conf/hive-site.xml`

  You can get the wasb location from the `hdfs-site.xml` file on the Hadoop cluster, or through the Ambari cluster management tool.

Configure the Hadoop Pushdown Properties for the Data Integration Service

Configure Hadoop pushdown properties for the Data Integration Service to run mappings in a Hadoop environment.

You can configure Hadoop pushdown properties for the Data Integration Service in the Administrator tool.

The following table describes the Hadoop pushdown properties for the Data Integration Service:

Property	Description
Informatica Home Directory on Hadoop	The Big Data Management home directory on every data node created by the Hadoop RPM install. Type `/opt/Informatica`.
Hadoop Distribution Directory	The directory containing a collection of Hive and Hadoop JARS on the cluster from the RPM Install locations. The directory contains the minimum set of JARS required to process Informatica mappings in a Hadoop environment. Type `/opt/Informatica/services/shared/hadoop/hortonworks_2.3`.
Data Integration Service Hadoop Distribution Directory	The Hadoop distribution directory on the Data Integration Service node. Type `../../services/shared/hadoop/hortonworks_2.3`. The contents of the Data Integration Service Hadoop distribution directory must be identical to Hadoop distribution directory on the data nodes.

Configuring the Hadoop Distribution Directory

You can modify the Hadoop distribution directory on the data nodes.

When you modify the Hadoop distribution directory, you must copy the minimum set of Hive and Hadoop JARS, and the Snappy libraries required to process Informatica mappings in a Hadoop environment from your Hadoop install location. The actual Hive and Hadoop JARS can vary depending on the Hadoop distribution and version.

The Hadoop RPM installs the Hadoop distribution directories in the following path:
`<BigDataManagementInstallationDirectory>/Informatica/services/shared/hadoop`.

Configure the Blaze Engine Log Directories

The `hadoopEnv.properties` file lists the log directories that the Blaze engine uses on the node and on HDFS. You must grant the user account that starts the Blaze engine write permission on the log directories.

Grant the user account that starts the Blaze engine write permission for the directories specified in the following properties:

- infagrid.node.local.root.log.dir
- infacal.hadoop.logs.directory

For more information about user accounts for the Blaze engine, see the *Informatica Big Data Management Security Guide*.

Configure Environment Variables in the Big Data Management Hadoop Environment Properties File

To add environment variables or to extend existing ones, use the Hadoop environment properties file, `hadoopEnv.properties`.

You can optionally add third-party environment variables or extend the existing PATH environment variable in `hadoopEnv.properties`.

1. Go to the following location: `<Informatica installation directory>/services/shared/hadoop/hortonworks_<version_number>/infaConf`
2. Find the file `hadoopEnv.properties`.
3. Back up the file before you modify it.

4. Use a text editor to open the file and modify the properties.
5. Save the properties file with the name `hadoopEnv.properties`.

Edit Informatica Developer Files and Variables

Edit `developerCore.ini` to enable the Developer tool to communicate with the Hadoop cluster on a particular Hadoop distribution. After you edit the file, you must click `run.bat` to launch the Developer tool again.

`developerCore.ini` is located in the following directory: `<Informatica installation directory>\clients\DeveloperClient`

Add the following property to `developerCore.ini`

```
-DINFA_HADOOP_DIST_DIR=hadoop\hortonworks_<version_number>
```

Open the Required Ports for the Blaze Engine

You must open a range of ports for the Blaze engine to use to communicate with the Informatica domain.

Note: Skip this task if the Blaze engine does not support the distribution that the Hadoop cluster runs.

If the Hadoop cluster is behind a firewall, work with your network administrator to open the range of ports that the Blaze engine uses.

When you create the Hadoop connection, specify the port range that the Blaze engine can use with the minimum port and maximum port fields.

Enable the Blaze Engine Console

Enable the Blaze engine console in the `hadoopEnv.properties` file.

1. On the machine where the Data Integration Service runs, edit the `hadoopEnv.properties` file.

   You can find `hadoopEnv.properties` in the following directory: `<Informatica installation directory>/services/shared/hadoop/<Hadoop distribution>/infaConf`.
2. Set the `infagrid.blaze.console.enabled` property to `true`.
3. Save and close the `hadoopEnv.properties` file.

Configure Hive CLI or HiveServer2

You can choose to run mappings using the Hive Command Line Interface (CLI) or HiveServer2, a graphic user interface.

The default tool is Hive CLI.

Use Hive CLI to Run Mappings

When you choose the Hive Command Line Interface (CLI) to run mappings, no configuration is required.

Configuring Hadoop Cluster Properties and the Data Integration Service to Use HiveServer2

The `hadoopEnv.properties file` contains two entries for the `infapdo.aux.jars.path` property. By default, the entry for Hive CLI is uncommented, and the entry for HiveServer2 is commented out. Manually edit the entries to choose a method to run mappings.

1. Edit the Hadoop environment properties file to set HiveServer2 as the tool to run mappings.
   a. Browse to the `hadoopEnv.properties` file in the following directory: `<Informatica installation directory>/services/shared/hadoop/hortonworks_<version_number>/infaConf`.

      The `hadoopEnv.properties file` contains two entries for the `infapdo.aux.jars.path` property. The default value is Hive CLI, and the entry for HiveServer2 is commented out.
   b. To use HiveServer2, comment out the Hive CLI entry, and uncomment the HiveServer2 entry.
2. Assign the required permissions on the cluster to the user account specified in the Hive connection.

   For example, the user account testuser1 belongs to the "Other" user group. To use this account, verify that the "Other" user group has permissions on the Hive Warehouse Directory.

   Additionally, testuser1 must have the following permissions:

   - Full permission on the staging directory
   - Full permission on the `/tmp/hive-<username>` directory
   - Read and write permission on the `/tmp` directory
3. Use the Administrator tool in the Informatica domain to configure the Data Integration Service for HiveServer2.
   a. Log in to the Administrator tool.
   b. In the **Domain Navigator**, select the Data Integration Service.
   c. In the **Processes** tab, create the following custom property: `ExecutionContextOptions.hive.executor`.
   d. Set the value to `hiveserver2`.
   e. Recycle the Data Integration Service.
4. Disable SQL-based authorization for HiveServer2.
5. Optionally, enable storage-based authorization.

Troubleshooting HiveServer2

Consider the following troubleshooting tips when you configure HiveServer2.

A mapping fails with the following error: java.lang.OutOfMemoryError: Java heap space

Increase the heap size that mapReduce can use with HiveServer2 to run mappings.

To configure the heap size, you must edit `hadoopEnv.properties`. You can find `hadoopEnv.properties` in the following directory: `<Informatica installation directory>/services/shared/hadoop/hortonworks_<version>/infaConf`.

Find the `infapdo.java.opts` property in `hadoopEnv.properties`.

Add the following values: `-Xms<memory size>m -Xmx<memory size>m`.

The following sample text shows the infapdo.java.opts property with a modified heap size:

```
infapdo.java.opts=-Djava.library.path=$HADOOP_NODE_INFA_HOME/services/shared/bin:
$HADOOP_NODE_HADOOP_DIST/lib/native:$HADOOP_NODE_HADOOP_DIST/lib/*:
$HADOOP_NODE_HADOOP_DIST/lib/native -Djava.security.egd=file:/dev/./urandom -Xms3150m -
Xmx6553m -XX:MaxPermSize=512m
```

Step 3. Configure Hadoop Cluster Properties on the Data Integration Service Machine for MapReduce 2

If the HDInsight cluster uses MapReduce 2, you must configure the Hadoop cluster properties in `hive-site.xml` and `yarn-site.xml` on the machine where the Data Integration Service runs.

`hive-site.xml` and `yarn-site.xml` are located in the following directory on the machine where the Data Integration Service runs: `<Informatica installation directory>/services/shared/hadoop/HDinsight_<version>_yarn/conf/`.

In `hive-site.xml`, configure the following property:

yarn.app.mapreduce.am.staging-dir

Location of the staging directory for the Hadoop cluster.

The following sample code describes the property you can set in `hive-site.xml`:

```
<property>
 <name>yarn.app.mapreduce.am.staging-dir</name>
 <value><staging directory path></value>
</property>
```

In `yarn-site.xml`, configure the following properties:

mapreduce.jobhistory.address

Location of the MapReduce JobHistory Server. The default value is 10020.

Use the value in the following file: `/etc/hdaoop/<version>/0/mapred-site.xml`

mapreduce.jobhistory.webapp.address

Web address of the MapReduce JobHistory Server. The default value is 19888.

Use the value in the following file: `/etc/hdaoop/<version>/0/mapred-site.xml`

yarn.resourcemanager.scheduler.address

Scheduler interface address. The default value is 8030.

Use the value in the following file: `/etc/hdaoop/<version>/0/yarn-site.xml`

yarn.resourcemanager.webapp.address

Resource Manager web application address.

Use the value in the following file: `/etc/hdaoop/<version>/0/yarn-site.xml`

The following sample code describes the properties you can set in `yarn-site.xml`:

```
<property>
    <name>mapreduce.jobhistory.address</name>
     <value>hostname:port</value>
<description>MapReduce JobHistory Server IPC host:port</description>
</property>

<property>
    <name>mapreduce.jobhistory.webapp.address</name>
    <value>hostname:port</value>
<description>MapReduce JobHistory Server Web UI host:port</description>
</property>

<property>
 <name>yarn.resourcemanager.scheduler.address</name>
 <value>hostname:port</value>
 <description>The address of the scheduler interface</description>
</property>


<property>
 <name>yarn.resourcemanager.webapp.address</name>
```

```
 <value>hostname:port</value>
 <description>The address for the Resource Manager web application.</description>
</property>
```

Step 4. Configure Mappings to Run on the Hadoop Cluster

You can enable Informatica mappings to run on a Hadoop cluster on Hortonworks with HDinsight 3.3.

Informatica supports Hortonworks HDInsight clusters that are deployed on-premise on Microsoft Azure.

Note: If you do not use HiveServer2 to run mappings, skip steps 5-9.

To enable Informatica mappings to run on a Hortonworks HDInsight cluster, complete the following steps:

1. Configure Hadoop cluster properties for the Data Integration Service.
2. Configure the mapping logic pushdown method.
3. Add hbase_protocol.jar to the Hadoop classpath.
4. Enable HBase support.
5. Configure the Hadoop cluster for the Blaze engine.
6. Create the HiveServer2 environment variables and configure the HiveServer2 environment.
7. Disable SQL standard based authorization to run mappings with HiveServer2.
8. Enable storage based authorization with HiveServer2.
9. Enable support for HBase with HiveServer2.
10. Configure HiveServer2 for DB2 partitioning.

1. Enable the Data Integration Service to use Hive CLI to run mappings.
2. Configure Hadoop cluster properties in hive-site.xml and yarn-site.xml.
3. Configure the mapping logic pushdown method.
4. Add hbase_protocol.jar to the Hadoop classpath.
5. Configure the Hadoop cluster for the Blaze engine.
6. Create the HiveServer2 environment variables and configure the HiveServer2 environment.
7. Disable SQL standard based authorization to run mappings with HiveServer2.
8. Enable storage based authorization with HiveServer2.
9. Enable support for HBase with HiveServer2.
10. Configure HiveServer2 for DB2 partitioning.

Enable the Data Integration Service to Use Hive CLI to Run Mappings

Perform the following tasks to enable the Data Integration Service to use Hive CLI to run mappings:

1. Copy the following files from the Hadoop cluster to the following location on the machine that hosts the Data Integration Service: `<Informatica_installation_directory>/hortonworks_2.3/lib`
   - `/usr/hdp/<CurrentVersion>/hadoop/hadoop-azure-2.7.1.2.3.3.1-7.jar`
   - `/usr/hdp/<CurrentVersion>/hadoop/lib/azure-storage-2.2.0.jar`
   - `/usr/hdp/<CurrentVersion>/hadoop/lib/jetty-util-6.1.26.hwx.jar`

2. Copy the following files from the Hadoop cluster to the following location on the machine that hosts the Data Integration Service: `<Informatica_installation_directory>/usr/lib/python2.7/dist-packages/hdinsight_common`
   - `/usr/lib/python2.7/dist-packages/hdinsight_common/decrypt.sh`
   - `/usr/lib/python2.7/dist-packages/hdinsight_common/key_decryption_cert.prv`

Configure Hadoop Cluster Properties in hive-site.xml and yarn-site.xml

If you use Hive CLI to run mappings, configure Hadoop cluster properties in files that the Data Integration Service uses when it runs mappings on a HDInsight cluster.

Configure hive-site.xml for the Data Integration Service

Configure the Hortonworks cluster properties in the `hive-site.xml` file that the Data Integration Service uses when it runs mappings in a Hadoop cluster.

1. Open the `hive-site.xml` file in the following directory on the node on which the Data Integration Service runs:

   ```
   <Informatica installation directory>/services/shared/hadoop/hortonworks_<version>/conf/
   ```

   To run a mapping in HiveServer2, configure the following property in the `hive-site.xml` file:
   hive.metastore.uris

   URI for the metastore host.

   The following sample text shows the property you can configure in the `hive-site.xml` file:

   ```
   <property>
   <name>hive.metastore.uris</name>
   <value>thrift://<HOSTNAME>:9083</value>
   </property>
   ```

2. Use the Ambari cluster management configuration tool to get the values of the following properties from the cluster, and add the properties to `hive-site.xml`:
   - `<name>fs.azure.account.key.ilabsstoragevpn.blob.core.windows.net</name>`
   - `<name>fs.azure.account.keyprovider.ilabsstoragevpn.blob.core.windows.net</name>`
   - `<name>fs.azure.shellkeyprovider.script</name>`

Configure yarn-site.xml for the Data Integration Service

You need to configure the Hortonworks cluster properties in the `yarn-site.xml` file that the Data Integration Service uses when it runs mappings in a Hadoop cluster. If you use the Big Data Management Configuration Utility to configure Big Data Management, the `yarn-site.xml` file is automatically configured.

Open the `yarn-site.xml` file in the following directory on the node on which the Data Integration Service runs:

```
<Informatica installation directory>/services/shared/hadoop/hortonworks_<version>/conf/
```

Configure the following properties in the `yarn-site.xml` file:
yarn.resourcemanager.scheduler.address

Scheduler interface address.

Use the value in the following file: `/etc/hadoop/conf/yarn-site.xml`

yarn.resourcemanager.webapp.address

Web application address for the Resource Manager.

Use the value in the following file: `/etc/hadoop/conf/yarn-site.xml`.

The following sample text shows the properties you can set in the `yarn-site.xml` file:

```
<property>
  <name>yarn.resourcemanager.scheduler.address</name>
  <value>hostname:port</value>
  <description>The address of the scheduler interface</description>
</property>

<property>
 <name>yarn.resourcemanager.webapp.address</name>
 <value>hostname:port</value>
 <description>The address for the Resource Manager web application.</description>
</property>
```

Configure mapred-site.xml for the Data Integration Service

You need to configure the Hortonworks cluster properties in the `mapred-site.xml` file that the Data Integration Service uses when it runs mappings in a Hadoop cluster.

Open the `mapred-site.xml` file in the following directory on the node on which the Data Integration Service runs:

```
<Informatica installation directory>/services/shared/hadoop/hortonworks_<version>/conf/
```

Configure the following properties in the `mapred-site.xml` file:

mapreduce.jobhistory.intermediate-done-dir

Directory where the MapReduce jobs write history files.

Use the value in the following file: `/etc/hadoop/conf/mapred-site.xml`

mapreduce.jobhistory.done-dir

Directory where the MapReduce JobHistory server manages history files.

Use the value in the following file: `/etc/hadoop/conf/mapred-site.xml`

The following sample text shows the properties you must set in the mapred-site.xml file:

```
<property>
    <name>mapreduce.jobhistory.intermediate-done-dir</name>
    <value>/mr-history/tmp</value>
    <description>Directory where MapReduce jobs write history files.</description>
</property>

<property>
    <name>mapreduce.jobhistory.done-dir</name>
    <value>/mr-history/done</value>
    <description>Directory where the MapReduce JobHistory server manages history files.</
description>
</property>
```

If you use the Big Data Management Configuration Utility to configure Big Data Management, the following properties are automatically configured in `mapred-site.xml`. If you do not use the utility, configure the following properties in `mapred-site.xml`:

mapreduce.jobhistory.address

Location of the MapReduce JobHistory Server.

Use the value in the following file:`/etc/hadoop/conf/mapred-site.xml`

mapreduce.jobhistory.webapp.address

Web address of the MapReduce JobHistory Server.

Use the value in the following file: `/etc/hadoop/conf/mapred-site.xml`

The following sample text shows the properties you can set in the `mapred-site.xml` file:

```
<property>
  <name>mapreduce.jobhistory.address</name>
  <value>hostname:port</value>
```

```
  <description>MapReduce JobHistory Server IPC host:port</description>
</property>

<property>
  <name>mapreduce.jobhistory.webapp.address</name>
  <value>hostname:port</value>
  <description>MapReduce JobHistory Server Web UI host:port</description>
</property>
```

Configure Rolling Upgrades for HDInsight

To enable support for rolling upgrades for HDInsight, you must configure the following properties in `mapred-site.xml` on the machine where the Data Integration Service runs:

mapreduce.application.classpath

Classpaths for MapReduce applications.

Use the following value:

```
$PWD/mr-framework/hadoop/share/hadoop/mapreduce/*:$PWD/mr-framework/hadoop/share/hadoop/
mapreduce/lib/*:$PWD/mr-framework/hadoop/share/hadoop/common/*:$PWD/mr-framework/hadoop/
share/hadoop/common/lib/*:$PWD/mr-framework/hadoop/share/hadoop/yarn/*:$PWD/mr-framework/
hadoop/share/hadoop/yarn/lib/*:$PWD/mr-framework/hadoop/share/hadoop/hdfs/*:$PWD/mr-
framework/hadoop/share/hadoop/hdfs/lib/*:$PWD/mr-framework/hadoop/share/hadoop/
tools/lib/*:/usr/hdp/${hdp.version}/hadoop/lib/hadoop-lzo-0.6.0.${hdp.version}.jar:/etc/
hadoop/conf/secure
```

Replace *${hdp.version}* with the version number of the Hortonworks HDInsights cluster.

mapreduce.application.framework.path

Path for the MapReduce framework archive.

Use the following value:

```
/hdp/apps/${hdp.version}/mapreduce/mapreduce.tar.gz#mr-framework
```

Replace *${hdp.version}* with the version of the hortonworks HDInsights cluster.

The following sample text shows the properties you can set in the `mapred-site.xml` file:

```
<property>
  <name>mapreduce.jobhistory.address</name>
  <value>hostname:port</value>
  <description>MapReduce JobHistory Server IPC host:port</description>
</property>

<property>
  <name>mapreduce.jobhistory.webapp.address</name>
  <value>hostname:port</value>
  <description>MapReduce JobHistory Server Web UI host:port</description>
</property>
```

Configure the Mapping Logic Pushdown Method

You can use MapReduce or Tez to push mapping logic to the Hadoop cluster. You enable MapReduce or Tez for the Data Integration Service or for a connection.

When you enable MapReduce or Tez for the Data Integration Service, that execution engine becomes the default execution engine to push mapping logic to the Hadoop cluster. When you enable MapReduce or Tez for a connection, that engine takes precedence over the execution engine set for the Data Integration Service.

Choose MapReduce or Tez as the Execution Engine for the Data Integration Service

To use MapReduce or Tez as the default execution engine to push mapping logic to the Hadoop cluster, perform the following steps:

1. Open `hive-site.xml` in the following directory on the node on which the Data Integration Service runs:

   ```
   <Informatica installation directory>/services/shared/hadoop/hortonworks_<version>/conf/
   ```

2. Edit the hive.execution.engine property.
   The following sample text shows the property in `hive-site.xml`:

   ```
   <property>
    <name>hive.execution.engine</name>
    <value>tez</value>
    <description>Chooses execution engine. Options are: mr (MapReduce, default) or tez
   (Hadoop 2 only)</description>
   </property>
   ```

   Set the value of the property as follows:

   - *mr* -- Sets MapReduce as the execution engine.
   - *tez* -- Sets Tez as the execution engine.

Enable Tez for a Hadoop or Hive Connection

When you enable Tez for a connection, the Data Integration Service uses Tez to push mapping logic to the Hadoop cluster regardless of what is set for the Data Integration Service.

1. Open the Developer tool.
2. Click **Window** > **Preferences**.
3. Select **Informatica** > **Connections**.
4. Expand the domain.
5. Expand the **Databases** and select the **Hadoop** or **Hive** connection.
6. Edit the connection and configure the Environment SQL property on the **Database Connection** tab.
   Use the following value: `set hive.execution.engine=tez;`

If you enable Tez for the Data Integration Service but want to use MapReduce, you can use the following value for the Environment SQL property: `set hive.execution.engine=mr;`.

Configure Tez

If you use Tez as the execution engine, you must configure properties in `tez-site.xml`.

You can find `tez-site.xml` in the following directory on the machine where the Data Integration Service runs: `<Informatica installation directory>/services/shared/hadoop/hortonworks_<version>/conf`.

Configure the following properties:

tez.lib.uris

Specifies the location of `tez.tar.gz` on the Hadoop cluster.

Use the value specified in `tez-site.xml` on the cluster. You can find `tez-site.xml` in the following directory on any node in the cluster: `/etc/tez/conf`.

tez.am.launch.env

Specifies the location of Hadoop libraries.

Use the following syntax when you configure `tez-site.xml`:

```
<property>
 <name>tez.lib.uris</name>
 <value><file system default name>://<directory of tez.tar.gz></value>
```

```
 <description>The location of tez.tar.gz. Set tez.lib.uris to point to the tar.gz uploaded to
HDFS.</description>
</property>

<property>
 <name>tez.am.launch.env</name>
 <value>LD_LIBRARY_PATH=<HDInsight directory>/<HDInsight version>/hadoop/lib/native</value>
 <description>The location of Hadoop libraries.</description>
</property>
```

The following example shows the properties if `tez.tar.gz` is in the `/apps/tez/lib` directory on HDFS:

```
<property>
 <name>tez.lib.uris</name>
 <value>hdfs://<Active_Name_Node>:8020/hdp/apps/<version>/tez/tez.tar.gz</value>
 <description>The location of tez.tar.gz. Set tez.lib.uris to point to the tar.gz uploaded to
HDFS.</description>
</property>

<property>
 <name>tez.am.launch.env</name>
 <value>LD_LIBRARY_PATH=/usr/hdp/<hadoop_version>/hadoop/lib/native</value>
 <description>The location of Hadoop libraries.</description>
</property>
```

Configure Tez for HiveServer2

If you use HiveServer2 to run mappings, open the `tez-site.xml` file. Verify that the following properties are commented out:

- tez.am.launch.cmd-opts
- tez.task.launch.env
- tez.am.launch.env

Enable HBase Support

To use HBase as a source or target when you run a mapping in the Hadoop environment, you must add `hbase-site.xml` to a distributed cache.

Perform the following steps:

1. On the machine where the Data Integration Service runs, go to the following directory: `<Informatica installation directory>/services/shared/hadoop/hortonworks_<version>/infaConf`.
2. Edit `hadoopEnv.properties`.
3. Verify the HBase version specified in `infapdo.env.entry.mapred_classpath` uses the correct HBase version for the Hadoop cluster.

   The following sample text shows `infapdo.env.entry.mapred_classpath` for a Hadoop cluster that uses HBase version 1.1.1.2.3.0.0-2504:

   ```
   infapdo.env.entry.mapred_classpath=INFA_MAPRED_CLASSPATH=$HADOOP_NODE_HADOOP_DIST/lib/
   hbase-server-1.1.1.2.3.0.0-2504.jar:$HADOOP_NODE_HADOOP_DIST/lib/htrace-core.jar:
   $HADOOP_NODE_HADOOP_DIST/lib/htrace-core-2.04.jar:$HADOOP_NODE_HADOOP_DIST/lib/protobuf-
   java-2.5.0.jar:$HADOOP_NODE_HADOOP_DIST/lib/hbase-client-1.1.1.2.3.0.0-2504.jar:
   $HADOOP_NODE_HADOOP_DIST/lib/hbase-common-1.1.1.2.3.0.0-2504.jar:
   $HADOOP_NODE_HADOOP_DIST/lib/hive-hbase-handler-1.2.1.2.3.0.0-2504.jar:
   $HADOOP_NODE_HADOOP_DIST/lib/hbase-protocol-1.1.1.2.3.0.0-2504.jar
   ```

4. Add the following entry to the `infapdo.aux.jars.path` variable: `file://$DIS_HADOOP_DIST/conf/hbase-site.xml`.

   The following sample text shows `infapdo.aux.jars.path` with the variable added:

   ```
   infapdo.aux.jars.path=file://$DIS_HADOOP_DIST/infaLib/hive0.14.0-infa-boot.jar,file://
   $DIS_HADOOP_DIST/infaLib/hive-infa-plugins-interface.jar,file://$DIS_HADOOP_DIST/infaLib/
   ```

```
profiling-hive0.14.0-udf.jar,file://$DIS_HADOOP_DIST/infaLib/hadoop2.2.0-
avro_complex_file.jar,file://$DIS_HADOOP_DIST/conf/hbase-site.xml
```

5. On the machine where the Data Integration Service runs, go to the following directory: `<Informatica installation directory>/services/shared/hadoop/hortonworks_<version>/conf`.
6. In `hbase-site.xml` and `hive-site.xml`, verify that the`zookeeper.znode.parent` property exists and matches the property set in `hbase-site.xml` on the cluster.

   By default, the ZooKeeper directory on the cluster is `/usr/hdp/current/hbase-client/conf`.
7. On the machine where the Developer tool runs, go to the following directory: <Informatica installation directory>\clients\DeveloperClient\hadoop\hortonworks_<version>/conf.
8. In `hbase-site.xml` and `hive-site.xml`, verify that the`zookeeper.znode.parent` property exists and matches the property set in `hbase-site.xml` on the cluster.

   By default, the ZooKeeper directory on the cluster is `/usr/hdp/current/hbase-client/conf`.
9. Edit the Hadoop classpath on every node on the Hadoop cluster to point to the `hbase-protocol.jar` file. Then, restart the Node Manager for each node in the Hadoop cluster.

   `hbase-protocol.jar` is located in the HBase installation directory on the Hadoop cluster. For more information, refer to the following link: https://issues.apache.org/jira/browse/HBASE-10304

Configure the Hadoop Cluster for the Blaze Engine

To use the Blaze engine, you must configure the Hadoop cluster.

Complete the following tasks:

- Configure the `yarn-site.xml` file on every node in the Hadoop cluster.
- Start the Application Timeline Server.
- Enable the Blaze Engine console.

Configure the Hadoop cluster for the Application Timeline Server

You must configure properties in the `yarn-site.xml` file on the Hadoop cluster for the Application Timeline Server.

You can use Ambari to configure the required properties in the `yarn-site.xml` file. Alternatively, configure the `yarn-site.xml` file on every node in the Hadoop cluster.

You can find the `yarn-site.xml` file in the following directory on every node in the Hadoop cluster: `/etc/hadoop/conf`.

Configure the following properties:

yarn.timeline-service.webapp.address

The HTTP address for the Application Timeline service web application.

Use the host name of the machine that starts the Application Timeline Server for the host name.

yarn.timeline-service.enabled

Whether the Timeline service is enabled.

Set this value to `true`.

yarn.timeline-service.address

Address for the Application Timeline Server to start the RPC server.

Use the host name of the machine that starts the Application Timeline Server for the host name.

yarn.timeline-service.hostname

The host name for the Application Timeline Service web application.

Use the host name of the machine that starts the Application Timeline Server for the host name.

yarn.timeline-service.ttl-ms

The time to live in milliseconds for data in the timeline store.

Use 3600000.

yarn.nodemanager.resource.memory-mb

Amount of physical memory that can be allotted for containers.

The Blaze engine requires at least 6144 MB.

yarn.nodemanager.local-dirs

List of directories to store localized files in.

The Blaze engine uses local directories for a distributed cache.

The following sample text shows the properties you configure in the `yarn-site.xml` file:

```
<property>
 <name>yarn.timeline-service.webapp.address</name>
 <value><ATSHostname>:8188</value>
</property>

<property>
 <name>yarn.timeline-service.enabled</name>
 <value>true</value>
</property>

<property>
 <name>yarn.timeline-service.address</name>
 <value><ATSHostname>:10200</value>
</property>

<property>
 <name>yarn.timeline-service.hostname</name>
 <value><ATSHostname></value>

<property>
 <name>yarn.timeline-service.ttl-ms</name>
 <value>3600000</value>
</property>

<property>
 <name>yarn.nodemanager.resource.memory-mb</name>
 <value>6144</value>
 <description>Amount of physical memory that can be allotted for containers.</description>
</property>

<property>
 <name>yarn.nodemanager.local-dirs</name>
 <value>/<local directory>,/<local directory></value>
 </property>
```

Start the Hadoop Application Timeline Server

The Blaze engine uses the Hadoop Application Timeline Server to store the Job monitor status.

To start the Hadoop Application Timeline Server, run the following command on any node in the Hadoop cluster:

```
sudo yarn timelineserver &
```

Enable the Blaze Engine Console

Enable the Blaze engine console in the `hadoopEnv.properties` file.

1. On the machine where the Data Integration Service runs, edit the `hadoopEnv.properties` file.

   You can find `hadoopEnv.properties` in the following directory: `<Informatica installation directory>/services/shared/hadoop/<Hadoop distribution>/infaConf`.

2. Set the `infagrid.blaze.console.enabled` property to `true`.
3. Save and close the `hadoopEnv.properties` file.

Create the HiveServer2 Environment Variables to Configure the HiveServer2 Environment

Before you can configure the HiveServer2 environment, create the required environment variables. Then configure the HiveServer2 environment with Ambari or the `hive-env.sh` file.

You can run the Big Data Management Configuration Utility and select HiveServer2 to generate the `HiveServer2_EnvInfa.txt` file. Alternatively, you can modify a template to create the required environment variables.

Modify the following template:

```
export LD_LIBRARY_PATH=/opt/Informatica/services/shared/bin:/opt/Informatica/services/shared/
hadoop/hortonworks_2.3/lib/native:$LD_LIBRARY_PATH
export INFA_HADOOP_DIST_DIR=/opt/Informatica/services/shared/hadoop/hortonworks_2.3
export INFA_PLUGINS_HOME=/opt/Informatica/plugins

export TMP_INFA_AUX_JARS=$INFA_HADOOP_DIST_DIR/infaLib/hadoop2.4.0-hdfs-native-impl.jar:
$INFA_HADOOP_DIST_DIR/infaLib/hadoop2.7.1.hw23-native-impl.jar:$INFA_HADOOP_DIST_DIR/infaLib/
hbase1.1.2-infa-plugins.jar:$INFA_HADOOP_DIST_DIR/infaLib/hive0.14.0-infa-boot.jar:
$INFA_HADOOP_DIST_DIR/infaLib/hive0.14.0-infa-plugins.jar:$INFA_HADOOP_DIST_DIR/infaLib/
hive0.14.0-infa-storagehandler.jar:$INFA_HADOOP_DIST_DIR/infaLib/hive0.14.0-native-impl.jar:
$INFA_HADOOP_DIST_DIR/infaLib/hive1.1.0-avro_complex_file.jar:$INFA_HADOOP_DIST_DIR/infaLib/
hive-infa-plugins-interface.jar:$INFA_HADOOP_DIST_DIR/infaLib/infa-hadoop-hdfs.jar:
$INFA_HADOOP_DIST_DIR/infaLib/profiling-hive0.14.0-udf.jar:/opt/Informatica/infa_jars.jar:
$INFA_HADOOP_DIST_DIR/lib/parquet-avro-1.6.0rc3.jar

if [ "${HIVE_AUX_JARS_PATH}" != "" ]; then
  export HIVE_AUX_JARS_PATH=$HIVE_AUX_JARS_PATH:$TMP_INFA_AUX_JARS
else
  export HIVE_AUX_JARS_PATH=$TMP_INFA_AUX_JARS
fi

export JAVA_LIBRARY_PATH=/opt/Informatica/services/shared/bin
export INFA_RESOURCES=/opt/Informatica/services/shared/bin
export INFA_HOME=/opt/Informatica
export IMF_CPP_RESOURCE_PATH=/opt/Informatica/services/shared/bin
export
INFA_MAPRED_OSGI_CONFIG='osgi.framework.activeThreadType:false&:org.osgi.framework.storage.clean
:none&:eclipse.jobs.daemon:true&:infa.osgi.enable.workdir.reuse:true&:infa.osgi.parent.workdir::
/tmp/infa&:infa.osgi.workdir.poolsize:4'
```

In the template text above, replace the following text:

- Replace <HADOOP_NODE_INFA_HOME> with the Informatica installation directory on the HDInsight 3.3 cluster.
- Replace <HADOOP_DISTRIBUTION> with the Informatica Hadoop installation directory on the HDInsight 3.3 cluster.

Note: If you use Ambari with CSH as the default shell, you must change the `export` command to `set`.

After you create the environment variables, configure the HiveServer2 environment with Ambari or the `hive-env.sh` file.

Configure the HiveServer2 Environment with Ambari

After you create the HiveServer2 environment variables, configure the HiveServer2 environment.

1. Open the modified template.
2. Copy the contents of the file.

   Note: If you use Ambari with CSH as the default shell, you must change the `export` command to `set`.
3. Log in to Ambari.
4. Click **Hive** > **Configs** > **Advanced**.
5. Search for the "hive-env template" property.
6. Paste the contents of the modified template.
7. Save the changes.
8. Restart the HiveServer2 services.

Configure the HiveServer2 Environment with hive-env

After you create the HiveServer2 environment variables with the modified template, configure the HiveServer2 environment with the `hive-env.sh` file.

1. Open the `hive-env.sh` file.

   You can find `hive-env.sh` in the following directory: `/etc/hive/conf/hive-env.sh`.
2. Copy and paste the contents of the modified template at the end of `hive-env.sh`.
3. Restart HiveServer2 services.

Disable SQL Standard Based Authorization to Run Mappings with HiveServer2

If the Hadoop cluster uses SQL standard based authorization, you must disable it to run mappings with HiveServer2.

1. Log in to Ambari.
2. Select **Hive** > **Configs**.
3. In the Security section, set **Hive Security Authorization** to **None**.
4. Navigate to the Advanced tab for hiveserver2-site.
5. Set **Enable Authorization** to **false**.
6. Restart Hive Services.

Enable Storage Based Authorization with HiveServer2

Optionally, you can use storage-based authorization with HiveServer2.

1. Log in to Ambari.
2. Click **Hive** > **Configs**.
3. In the Security section, set the Hive Security Authorization to **SQLStdAuth**.
4. Navigate to **Advanced Configs**.
5. In the **General** section, verify that the Hive Authorization Manager property is set to the following value:

Hive Authorization Manager

Set this property to the following value:

```
org.apache.hadoop.hive.ql.security.authorization.StorageBasedAuthorizationProvider,or
g.apache.hadoop.hive.ql.security.authorization.MetaStoreAuthzAPIAuthorizerEmbedOnly
```

hive.security.authorization.manager

Set this property to the following value:

```
org.apache.hadoop.hive.ql.security.authorization.StorageBasedAuthorizationProvider
```

By default, this property is set to the following value:

```
org.apache.hadoop.hive.ql.security.authorization.plugin.sqlstd.SQLStdConfOnlyAuthoriz
erFactory
```

Enable Authorization

Set this property to True.

6. In the **Advanced hiveserver2-site** section, configure the following properties:

Enable Authorization

Set this value to True.

hive.security.authorization.manager

Set this property to the following value:

```
org.apache.hadoop.hive.ql.security.authorization.StorageBasedAuthorizationProvider
```

By default, this property is set to the following value:

```
org.apache.hadoop.hive.ql.security.authorization.plugin.sqlstd.SQLStdHiveAuthorizerFa
ctory
```

7. Restart all Hive services.

Enable Support for HBase with HiveServer2

You must configure Big Data Management to run a mapping that uses an HBase source or target with HiveServer2.

Perform the following steps:

1. Verify that the value for the `zookeeper.znode.parent` property in the `hbase-site.xml` file on the machine where the Data Integration Service runs matches the value on the Hadoop cluster.

   The default value is `/hbase-unsecure`.

   You can find the `hbase-site.xml` file in the following directory on the machine where the Data Integration Service runs: `<Informatica installation directory>/services/shared/hadoop/hortonworks_<version>/conf`.

   You can find the `hbase-site.xml` file in the following directory on the Hadoop cluster: `<Informatica installation directory>/services/shared/hadoop/hortonworks_<version>`.

2. Verify that the `infapdo.aux.jars.path` property contains the path to the `hbase-site.xml` file.

   The following sample text shows the `infapdo.aux.jars.path` property with the path for hbase-site.xml:

```
infapdo.aux.jars.path=file://$HADOOP_NODE_HADOOP_DIST/infaLib/hive0.14.0-infa-
boot.jar,file://$HADOOP_NODE_HADOOP_DIST/infaLib/hive-infa-plugins-interface.jar,file://
$HADOOP_NODE_HADOOP_DIST/infaLib/profiling-hive0.14.0-udf.jar,file://
$HADOOP_NODE_HADOOP_DIST/infaLib/hadoop2.2.0-avro_complex_file.jar,file://
$HADOOP_NODE_HADOOP_DIST/conf/hbase-site.xml,file://$HADOOP_NODE_HADOOP_DIST/infaLib/
infa_jars.jar
```

Configure HiveServer2 for DB2 Partitioning

To use HiveServer2 with DB2 database partitioning, use Ambari to configure the to configure the LD_LIBRARY_PATH for HiveServer2.

Note: If the Hadoop cluster uses RPMs, you must manually edit the `hive-env.sh` file to add the `<DB2_HOME>/lib64` directory to LD_LIBRARY_PATH. You can find `hive-env.sh` in the following directory: `/etc/hive/conf`

1. Open Ambari.
2. Click **Hive** > **Configs** > **Advanced**.
3. Search for the `hive-env template` property.
4. Add the following directory to the LD_LIBRARY_PATH property: `<DB2_HOME>/lib64`.
5. Restart the Hive services.

Step 5. Update Cluster Configuration Settings

Perform the following steps to update the Hadoop cluster to enable support for Azure HDInsight.

Note: You can use the Ambari cluster configuration tool to view and edit cluster properties. After you change property values, the Ambari tool displays the affected cluster components. Restart the affected components for the changes to take effect.

1. Choose where scripts will be executed.
   - Edit each of the files in the following directory on each of the nodes where you installed Big Data Management: `<Informatica installation home>/services/shared/hadoop/hortonworks_2.3/scripts` to change the value of `/bin/sh` from `dash` to `bash`.
   - If you cannot change the value of `/bin/sh`, then change scripts to point to `/bin/bash` instead of `/bin/sh`.

   Place scripts in the following path:

   ```
   <Cluster_Installation_directory>/services/shared/hadoop/hortonworks_2.3/scripts
   ```

   Note: Restart the affected components for the changes to take effect.
2. Set the value of the fs.defaultFS property to the HDFS location you want.

   After you restart, the cluster will populate files in the HDFS location.
   Optionally, you can reset fs.defaultFS to the wasb location, and restart the affected components again.

   Note: If you are not able to reset the value of the fs.defaultFS property, you can manually copy the entire folder structure from the wasb location to the local HDFS location.

Connections

Define the connections that you want to use to access data in HBase, HDFS, or Hive, or run a mapping on a Hadoop cluster. You can create the connections using the Developer tool, Administrator tool, and infacmd.

You can create the following types of connections:

Hadoop connection

Create a Hadoop connection to run mappings on the Hadoop cluster. Select the Hadoop connection if you select the Hadoop run-time environment. You must also select the Hadoop connection to validate a mapping to run on the Hadoop cluster. Before you run mappings in the Hadoop cluster, review the information in this guide about rules and guidelines for mappings that you can run in the Hadoop cluster.

HDFS connection

Create an HDFS connection to read data from or write data to the HDFS file system on the Hadoop cluster.

HBase connection

Create an HBase connection to access HBase. The HBase connection is a NoSQL connection.

Hive connection

Create a Hive connection to access Hive as a source or target. You can access Hive as a source if the mapping is enabled for the native or Hadoop environment. You can access Hive as a target only if the mapping uses the Hive engine.

Note: For information about creating connections to other sources or targets such as social media web sites or Teradata, see the respective PowerExchange adapter user guide for information.

HDFS Connection Properties

Use a Hadoop File System (HDFS) connection to access data in the Hadoop cluster. The HDFS connection is a file system type connection. You can create and manage an HDFS connection in the Administrator tool, Analyst tool, or the Developer tool. HDFS connection properties are case sensitive unless otherwise noted.

Note: The order of the connection properties might vary depending on the tool where you view them.

The following table describes HDFS connection properties:

Property	Description
Name	Name of the connection. The name is not case sensitive and must be unique within the domain. The name cannot exceed 128 characters, contain spaces, or contain the following special characters: `~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /`
ID	String that the Data Integration Service uses to identify the connection. The ID is not case sensitive. It must be 255 characters or less and must be unique in the domain. You cannot change this property after you create the connection. Default value is the connection name.
Description	The description of the connection. The description cannot exceed 765 characters.
Location	The domain where you want to create the connection. Not valid for the Analyst tool.
Type	The connection type. Default is Hadoop File System.
User Name	User name to access HDFS.
NameNode URI	The URI to access HDinsight-FS. Use the following URI: `hdfs://`

HBase Connection Properties

Use an HBase connection to access HBase. The HBase connection is a NoSQL connection. You can create and manage an HBase connection in the Administrator tool or the Developer tool. HBase connection properties are case sensitive unless otherwise noted.

The following table describes HBase connection properties:

Property	Description
Name	The name of the connection. The name is not case sensitive and must be unique within the domain. You can change this property after you create the connection. The name cannot exceed 128 characters, contain spaces, or contain the following special characters: `~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /`
ID	String that the Data Integration Service uses to identify the connection. The ID is not case sensitive. It must be 255 characters or less and must be unique in the domain. You cannot change this property after you create the connection. Default value is the connection name.
Description	The description of the connection. The description cannot exceed 4,000 characters.
Location	The domain where you want to create the connection.
Type	The connection type. Select HBase.
ZooKeeper Host(s)	Name of the machine that hosts the ZooKeeper server.
ZooKeeper Port	Port number of the machine that hosts the ZooKeeper server. Use the value specified for `hbase.zookeeper.property.clientPort` in hbase-site.xml. You can find `hbase-site.xml` on the Namenode machine in the following directory: `/opt/HDinsight/hbase/hbase-0.98.7/conf`
Enable Kerberos Connection	Enables the Informatica domain to communicate with the HBase master server or region server that uses Kerberos authentication.
HBase Master Principal	Service Principal Name (SPN) of the HBase master server. Enables the ZooKeeper server to communicate with an HBase master server that uses Kerberos authentication. Enter a string in the following format: `hbase/<domain.name>@<YOUR-REALM>` Where: - domain.name is the domain name of the machine that hosts the HBase master server. - YOUR-REALM is the Kerberos realm.
HBase Region Server Principal	Service Principal Name (SPN) of the HBase region server. Enables the ZooKeeper server to communicate with an HBase region server that uses Kerberos authentication. Enter a string in the following format: `hbase_rs/<domain.name>@<YOUR-REALM>` Where: - domain.name is the domain name of the machine that hosts the HBase master server. - YOUR-REALM is the Kerberos realm.

Hive Connection Properties

Use the Hive connection to access Hive data. A Hive connection is a database type connection. You can create and manage a Hive connection in the Administrator tool, Analyst tool, or the Developer tool. Hive connection properties are case sensitive unless otherwise noted.

Note: The order of the connection properties might vary depending on the tool where you view them.

The following table describes Hive connection properties:

Property	Description
Name	The name of the connection. The name is not case sensitive and must be unique within the domain. You can change this property after you create the connection. The name cannot exceed 128 characters, contain spaces, or contain the following special characters: ~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /
ID	String that the Data Integration Service uses to identify the connection. The ID is not case sensitive. It must be 255 characters or less and must be unique in the domain. You cannot change this property after you create the connection. Default value is the connection name.
Description	The description of the connection. The description cannot exceed 4000 characters.
Location	The domain where you want to create the connection. Not valid for the Analyst tool.
Type	The connection type. Select Hive.
Connection Modes	Hive connection mode. Select at least one of the following options: - Access Hive as a source or target. Select this option if you want to use the connection to access the Hive data warehouse. If you want to use Hive as a target, you must enable the same connection or another Hive connection to run mappings in the Hadoop cluster. - Use Hive to run mappings in Hadoop cluster. Select this option if you want to use the connection to run mappings in the Hadoop cluster. You can select both the options. Default is **Access Hive as a source or target**.

Property	Description
User Name	User name of the user that the Data Integration Service impersonates to run mappings on a Hadoop cluster. Use the user name of an operating system user that is present on all nodes on the Hadoop cluster.
Common Attributes to Both the Modes: Environment SQL	SQL commands to set the Hadoop environment. In native environment type, the Data Integration Service executes the environment SQL each time it creates a connection to a Hive metastore. If you use the Hive connection to run mappings in the Hadoop cluster, the Data Integration Service executes the environment SQL at the beginning of each Hive session. The following rules and guidelines apply to the usage of environment SQL in both connection modes: - Use the environment SQL to specify Hive queries. - Use the environment SQL to set the classpath for Hive user-defined functions and then use environment SQL or PreSQL to specify the Hive user-defined functions. You cannot use PreSQL in the data object properties to specify the classpath. The path must be the fully qualified path to the JAR files used for user-defined functions. Set the parameter hive.aux.jars.path with all the entries in infapdo.aux.jars.path and the path to the JAR files for user-defined functions. - You can use environment SQL to define Hadoop or Hive parameters that you want to use in the PreSQL commands or in custom queries. If you use the Hive connection to run mappings in the Hadoop cluster, the Data Integration service executes only the environment SQL of the Hive connection. If the Hive sources and targets are on different clusters, the Data Integration Service does not execute the different environment SQL commands for the connections of the Hive source or target.

Properties to Access Hive as Source or Target

The following table describes the connection properties that you configure to access Hive as a source or target:

Property	Description
Metadata Connection String	The JDBC connection URI used to access the metadata from the Hadoop server. You can use PowerExchange for Hive to communicate with a HiveServer service or HiveServer2 service. To connect to HiveServer2, specify the connection string in the following format: `jdbc:hive2://<hostname>:<port>/<db>;transportMode=<mode>` Where - <hostname> is name or IP address of the machine on which HiveServer2 runs. - <port> is the port number on which HiveServer2 listens. - <db> is the database to which you want to connect. If you do not provide the database name, the Data Integration Service uses the default database details. - <mode> is the value of the hive.server2.transport.mode property in the Hive tab of the Ambari tool.
Bypass Hive JDBC Server	JDBC driver mode. Select the check box to use the embedded JDBC driver mode. To use the JDBC embedded mode, perform the following tasks: - Verify that Hive client and Informatica services are installed on the same machine. - Configure the Hive connection properties to run mappings in the Hadoop cluster. If you choose the non-embedded mode, you must configure the Data Access Connection String. Informatica recommends that you use the JDBC embedded mode.
Data Access Connection String	The JDBC connection URI used to access data from the Hadoop server. You can use PowerExchange for Hive to communicate with a HiveServer service or HiveServer2 service. To connect to HiveServer2, specify the connection string in the following format: `jdbc:hive2://<hostname>:<port>/<db>;transportMode=<mode>` Where - <hostname> is name or IP address of the machine on which HiveServer2 runs. - <port> is the port number on which HiveServer2 listens. - <db> is the database to which you want to connect. If you do not provide the database name, the Data Integration Service uses the default database details. - <mode> is the value of the hive.server2.transport.mode property in the Hive tab of the Ambari tool.

Properties to Run Mappings in Hadoop Cluster

The following table describes the Hive connection properties that you configure when you want to use the Hive connection to run Informatica mappings in the Hadoop cluster:

Property	Description
Database Name	Namespace for tables. Use the name `default` for tables that do not have a specified database name.

Default FS URI	The URI to access the default HDinsight File System.
	Use the connection URI that matches the storage type. The storage type is configured for the cluster in the fs.defaultFS property.
	If the cluster uses HDFS storage, use the following string to specify the URI:
	`hdfs://<cluster_name>`
	Example:
	`hdfs://my-cluster`
	If the cluster uses wasb storage, use the following string to specify the URI:
	`wasb://<container_name>@<account_name>.blob.core.windows.net/<path>`
	where:
	- `<container_name>` identifies a specific Azure Blob storage container.
	Note: `<container_name>` is optional.
	- `<account_name>` identifies the the Azure storage object.
	Example:
	`wasb://infabdmoffering1storage.blob.core.windows.net/infabdmoffering1cluster/mr-history`

Yarn Resource Manager URI	The service within Hadoop that submits the MapReduce tasks to specific nodes in the cluster.
	For HDInsight 3.3 with YARN, use the following format:
	`<hostname>:<port>`
	Where
	- <hostname> is the host name or IP address of the JobTracker or Yarn resource manager.
	- <port> is the port on which the JobTracker or Yarn resource manager listens for remote procedure calls (RPC).
	Use the value specified by `yarn.resourcemanager.address` in `yarn-site.xml`. You can find `yarn-site.xml` in the following directory on the NameNode: `/etc/hive/<version>/0/`.
	For HDInsight 3.3 with MapReduce 2, use the following URI:
	`hdfs://host:port`

Hive Warehouse Directory on HDFS	The absolute HDFS file path of the default database for the warehouse that is local to the cluster. For example, the following file path specifies a local warehouse: `/user/hive/warehouse` If the Metastore Execution Mode is remote, then the file path must match the file path specified by the Hive Metastore Service on the hadoop cluster. Use the value specified for the `hive.metastore.warehouse.dir` property in `hive-site.xml`. You can find `yarn-site.xml` in the following directory on the node that runs HiveServer2: `/etc/hive/<version>/0/`.

Property	Description

Advanced Hive/Hadoop Properties	Configures or overrides Hive or Hadoop cluster properties in hive-site.xml on the machine on which the Data Integration Service runs. You can specify multiple properties.
	Use the following format:
	`<property1>=<value>`
	Where
	- <property1> is a Hive or Hadoop property in hive-site.xml.
	- <value> is the value of the Hive or Hadoop property.
	To specify multiple properties use `&:` as the property separator.
	The maximum length for the format is 1 MB.
	If you enter a required property for a Hive connection, it overrides the property that you configure in the Advanced Hive/Hadoop Properties.
	The Data Integration Service adds or sets these properties for each map-reduce job. You can verify these properties in the JobConf of each mapper and reducer job. Access the JobConf of each job from the Jobtracker URL under each map-reduce job.
	The Data Integration Service writes messages for these properties to the Data Integration Service logs. The Data Integration Service must have the log tracing level set to log each row or have the log tracing level set to verbose initialization tracing.
	For example, specify the following properties to control and limit the number of reducers to run a mapping job:
	`mapred.reduce.tasks=2&:hive.exec.reducers.max=10`

Temporary Table Compression Codec	Hadoop compression library for a compression codec class name.
Codec Class Name	Codec class name that enables data compression and improves performance on temporary staging tables.
Metastore Execution Mode	Controls whether to connect to a remote metastore or a local metastore. By default, local is selected. For a local metastore, you must specify the Metastore Database URI, Driver, Username, and Password. For a remote metastore, you must specify only the Remote Metastore URI.

Metastore Database URI	The JDBC connection URI used to access the data store in a local metastore setup. Use the following connection URI:
	`jdbc:<datastore type>://<node name>:<port>/<database name>`
	where
	- <node name> is the host name or IP address of the data store.
	- <data store type> is the type of the data store.
	- <port> is the port on which the data store listens for remote procedure calls (RPC).
	- <database name> is the name of the database.
	For example, the following URI specifies a local metastore that uses MySQL as a data store:
	`jdbc:mysql://hostname23:3306/metastore`
	Use the value specified for the `javax.jdo.option.ConnectionURL` property in `hive-site.xml`. You can find `hive-site.xml` in the following directory on the node that runs HiveServer2: `/etc/hive/<version>/0/hive-site.xml`.

Metastore Database Driver	Driver class name for the JDBC data store. For example, the following class name specifies a MySQL driver: Use the value specified for the `javax.jdo.option.ConnectionDriverName` property in `hive-site.xml`. You can find `hive-site.xml` in the following directory on the node that runs HiveServer2: `/etc/hive/<version>/0/hive-site.xml`.

Property	Description
Metastore Database Username	The metastore database user name. Use the value specified for the `javax.jdo.option.ConnectionUserName` property in `hive-site.xml`. You can find `hive-site.xml` in the following directory on the node that runs HiveServer2: `/etc/hive/<version>/0/hive-site.xml`.
Metastore Database Password	Required if the Metastore Execution Mode is set to local. The password for the metastore user name. Use the value specified for the `javax.jdo.option.ConnectionPassword` property in `hive-site.xml`. You can find `hive-site.xml` in the following directory on the node that runs HiveServer2: `/etc/hive/<version>/0/hive-site.xml`.
Remote Metastore URI	The metastore URI used to access metadata in a remote metastore setup. For a remote metastore, you must specify the Thrift server details. Use the following connection URI: `thrift://<hostname>:<port>` Where - <hostname> is name or IP address of the Thrift metastore server. - <port> is the port on which the Thrift server is listening. Use the value specified for the `hive.metastore.uris` property in `hive-site.xml`. You can find `hive-site.xml` in the following directory on the node that runs HiveServer2: `/etc/hive/<version>/0/hive-site.xml`.
Hive Connection String	The JDBC connection URI used to access the metadata from the Hadoop server. You can use PowerExchange for Hive to communicate with a HiveServer service or HiveServer2 service. To connect to HiveServer2, specify the connection string in the following format: `jdbc:hive2://<hostname>:<port>/<db>;transportMode=<mode>` Where - <hostname> is name or IP address of the machine on which HiveServer2 runs. - <port> is the port number on which HiveServer2 listens. - <db> is the database to which you want to connect. If you do not provide the database name, the Data Integration Service uses the default database details. - <mode> is the value of the hive.server2.transport.mode property in the Hive tab of the Ambari tool.

Creating a Connection to Access Sources or Targets

Create an HBase, HDFS, or Hive connection before you import data objects, preview data, and profile data.

1. Click **Window** > **Preferences**.
2. Select **Informatica** > **Connections**.
3. Expand the domain in the **Available Connections** list.
4. Select the type of connection that you want to create:
   - To select an HBase connection, select **NoSQL** > **HBase**.
   - To select an HDFS connection, select **File Systems** > **Hadoop File System**.
   - To select a Hive connection, select **Database** > **Hive**.
5. Click **Add**.
6. Enter a connection name and optional description.
7. Click **Next**.

8. Configure the connection properties. For a Hive connection, you must choose the Hive connection mode and specify the commands for environment SQL. The SQL commands apply to both the connection modes. Select at least one of the following connection modes:

Option	Description
Access Hive as a source or target	Use the connection to access Hive data. If you select this option and click **Next**, the **Properties to Access Hive as a source or target** page appears. Configure the connection strings.
Run mappings in a Hadoop cluster.	Use the Hive connection to validate and run profiles in the Hadoop cluster. If you select this option and click **Next**, the **Properties used to Run Mappings in the Hadoop Cluster** page appears. Configure the properties.

9. Click **Test Connection** to verify the connection.

   You can test a Hive connection that is configured to access Hive data. You cannot test a Hive connection that is configured to run Informatica mappings in the Hadoop cluster.

10. Click **Finish**.

Configuring Big Data Management in the Azure Cloud Environment

You can choose to run Big Data Management for Azure HDInsight in the Azure cloud environment.

Perform the following steps to create an implementation of Big Data Management in the Azure cloud.

1. Verify prerequisites.
2. Configure Big Data Management on the HDInsight cluster.
3. Download and apply EBF 17167 to the Informatica domain server.
4. Configure and start Informatica services.

Prerequisites

Before you launch the process to configure Big Data Management in the Azure cloud environment, check that you have fulfilled the following prerequisites.

- You have an instance of HDInsight in a Linux cluster that uses Hortonworks 2.3 up and running on the Azure environment.
- You have permission to access and administer the HDInsight instance, and to get the names and addresses of cluster resources and other information from cluster configuration pages.
- You have purchased a license for Informatica Big Data Management.

Configure Big Data Management on the HDInsight Cluster

Enter a short description of the task here (optional).

1. Select Big Data Management for setup.
   a. In the Azure marketplace, click the + button to create a new resource.
   b. Search on "Informatica" to find Informatica offerings in the Azure marketplace.

c. Select Big Data Management Enterprise Edition 10.0U1 BYOL.

   The "Create Big Data Management Enterprise Edition" tab opens. It displays all the steps necessary to configure and launch Big Data Management on the cluster.

2. Supply information in the **Basics** panel, and then click **OK**.

   Subscription

   Select the Azure subscription account that you want to use for Big Data Management.

   Charges for this instance of Big Data Management will go to this subscription.

   Resource Group

   Select a resource group to contain the BDM implementation.

   Usually, you select an existing resource group where you have a running HDInsight cluster.

   Location

   Location of the resource group.

   Accept the location that is already associated with the resource group.

3. Supply information in the **Node Settings** panel, and then click **OK**.

   This tab allows you to configure details of the Informatica domain. Azure deploys the domain on an Ubuntu Linux box.

   Number of nodes in the domain.

   Default is 2.

   Machine prefix

   Type an alphanumeric string that will be a prefix on the name of each virtual machine in the Informatica domain.

   For example, if you use the prefix "infa" then Azure will identify virtual machines in the domain with this string at the beginning of the name.

   Username

   Username that you use to log in to the virtual machine that hosts the Informatica domain.

   Authentication type

   Authentication protocol you use to communicate with the Informatica domain.

   Default is SSH Public Key.

   Password

   Password to use to log in to the virtual machine that hosts the Informatica domain.

   Machine size

   Select from among the available preconfigured VMs. The default is 2x Standard D11.

4. Supply information in the **Domain Settings** panel, and then click **OK**.

   This tab allows you to configure additional details of the Informatica domain.

   Informatica Domain Name

   Create a name for the Informatica domain.

   Informatica domain administrator name

   Login to use to administer the Informatica domain.

Password

Password for the Informatica administrator.

Keyphrase for encryption key

Create a keyphrase to create an encryption key.

5. Supply information in the **Database Settings** panel, and then click **OK**.

This tab allows you to configure settings for the storage where Informatica metadata will be stored.

Database type

Select SQL Server 2014.

Database machine name

Name for the virtual machine that hosts the domain database.

Database machine size

Select a size from among the available preconfigured virtual machines. The default is 1x Standard D3.

Username

Username for the administrator of the virtual machine host of the database.

These credentials to log into the virtual machine where the database is hosted.

Password

Password for the database machine administrator.

Informatica Domain DB User

Name of the database user.

The Informatica domain uses this account to communicate with the domain database.

Informatica Domain DB Password

Password for the database user.

6. Supply information in the **Informatica Big Data Management Configuration** panel, and then click OK.

This tab allows you to configure credentials that allow the Informatica domain to communicate with the HDInsight cluster. Get the information for these settings from HDInsight cluster settings panels and the Ambari cluster management tool.

HDInsight Cluster Hostname

Name of the HDInsight cluster where you want to create the Informatica domain.

HDInsight Cluster Login Username

User login for the cluster. This is usually the same login you use to log in to the Ambari cluster management tool.

Password

Password for the HDInsight cluster user.

HDInsight Cluster SSH Hostname

Name of the cluster SSH host.

HDInsight Cluster SSH Username

Account name you use to log in to the cluster head node.

Password

Password to access the cluster SSH host.

The panel requires you to input values for the following additional addresses. Get these addresses from the Ambari cluster management tool:

- mapreduce.jobhistory.address
- mapreduce.jobhistory.webapp.address
- yarn.resourcemanager.scheduler.address
- yarn.resourcemanager.webapp.address

7. Supply information in the **Infrastructure Settings** panel, and then click **OK**.

   Use this tab to set up cluster resources for the Big Data Management implementation.

   Storage account

   Storage resource that the virtual machines that run the Big Data Management implementation will use for data storage.

   Select an existing storage resource, or create a new one.

   When you select an existing storage resource, verify that it belongs to the resource group you want. It is not essential to select the same resource group as the group that the Big Data Management implementation belongs to.

   Virtual network

   Virtual network for the Big Data Management implementation to belong to. Select the same network as the one that you used to create the HDInsight cluster..

   Subnets

   The subnet that the virtual network contains.

   Accept the default subnet.

8. Verify the choices in the **Summary** panel, and then click **OK**.
9. Read the terms of use in the **Buy** panel, and then click **Create**.

   When you click **Create**, Azure deploys Big Data Management and creates resources in the environment that you configured.

Apply EBF 17167 to the Informatica Domain Server

To enable communication between the HDInsight cluster and the Informatica domain, you must download and apply EBF 17167 to the Informatica domain server.

The Azure portal installs a full Informatica domain instance on a virtual machine. To use it, you download and apply an EBF patch before starting the domain.

1. If the Informatica domain server is running, stop it.
2. Download and apply EBF 17167.

   To see how to apply the EBF, see Knowledge Base article 284396 at the following location: https://kb.informatica.com/howto/6/Pages/15/284396.aspx.
3. Restart the Informatica domain.

Configure and Start Informatica Services

After Azure finishes deploying the Informatica domain, use the Administrator tool to configure and start Informatica services.

The Administrator tool is a browser-enabled utility that allows you to create, configure and run different services on the Informatica domain.

To see more about Informatica services, see the *Informatica Application Service Guide*. You can download this and all other documentation from the Informatica portal.

To access the Administrator tool, perform the following steps:

1. Get the host name, IP address, and port of the virtual machine where Azure deployed the Informatica domain.
2. Add an entry for this domain host to your hosts file.

   In the entry, type the host name and IP address. For example:

   ```
   Informatica_host_name    1.2.3.4
   ```
3. Optionally, use a secure copy program to copy the license file to the domain host machine.
4. In a new browser tab, enter the following URL to start the Administrator tool:

   ```
   https://<Informatica_host_name>:<port_number>
   ```

   The default port number for the Administrator tool is 6008.
5. In the Administrator tool, upload the Big Data Management license file.

   The Big Data management license file is required to use Big Data Management on HDInsight. Perform the following steps:

   a. Click **Manage** > **Services and Nodes**.

      The Domain Navigator opens.

   b. Select the Domain and click **New** > **License**.

      The Create License window opens.

   c. Supply a name for the license, then browse to select the license file.
6. Use the Administrator tool to create, configure and run application services on the Informatica domain.

Troubleshooting

Job History details are not visible when wasb location is set at cluster and domain level.

Change *fs.defaultFS* from *wasb* location to *hdfs* location in both cluster and domain site-xml's. Restart the required cluster components and re-run the job.

Author

Big Data Management Team