Open for Innovation
KNIME

KNIME Azure Integration User Guide

KNIME AG, Zurich, Switzerland
Version 5.3 (last updated on 2024-07-05)

Table of Contents

Overview

KNIME Analytics Platform includes a set of nodes to support Azure cloud services. The supported Azure cloud services that will be covered in this guide are Azure HDInsight, Azure Blob Storage, Azure Data Lake Storage Gen2, and Azure SQL.

The KNIME Azure Cloud Connectors extension is available on KNIME Hub.

Azure Storage

KNIME Azure Cloud Connectors extension provides nodes to connect to Azure storage (Blob storage and Data Lake Storage Gen2).

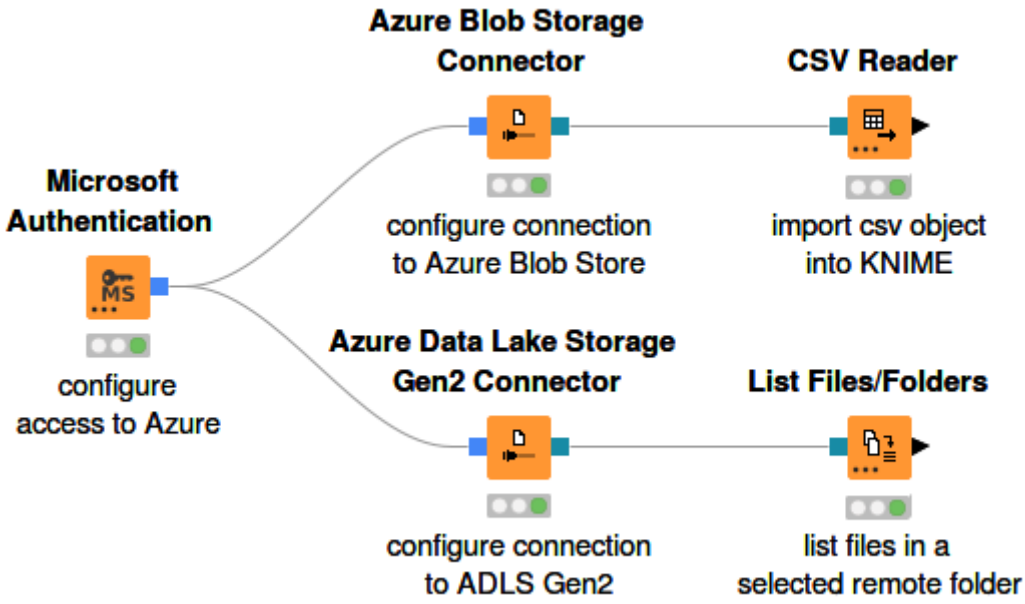

Figure 1. Connecting to and working with Azure storage

Figure 1 shows an example on how to connect to an Azure storage and work with the remote files.

Microsoft Authentication node provides authentication to access Microsoft Azure. The following authentication modes are supported:

- Interactive Authentication
- Username/password Authentication
- Shared key authentication (Azure Storage only)
- Shared access signature (SAS) authentication (Azure Storage only)

For more details on how to use the Microsoft Authentication node, please check out the KNIME File Handling Guide.

Azure Blob Storage

Azure Blob Store Connector node connects KNIME Analytics Platform with Azure Blob storage.

The node outputs a connection object that allows downstream nodes to access the Azure Blob Storage data as a file system, e.g. to read or write files and folders, or to perform other file system operations, such as browse/list files, copy, move.

Paths for Azure Blob Storage are specified with a UNIX-like syntax, `/mycontainer/myfolder/myfile`. An absolute path for Azure Blob Storage consists of:

- A leading slash (/)
- Followed by the name of a container (`mycontainer` in the above example), followed by a slash.
- Followed by the name of an object within the container (`myfolder/myfile` in the above example).

An example workflow on how to connect and work with remote files on Azure Blob Storage is available on KNIME Hub.

Azure Data Lake Storage Gen2

Azure ADLS Gen2 Connector node connects KNIME Analytics Platform with Azure Data Lake Storage Gen2.

Azure Data Lake Storage Gen2 enhances Azure Blob storage by adding hierarchical namespaces on top of the standard Blob storage. Operations such as renaming or deleting a directory, are now single atomic metadata operations on the directory, which provides more efficient data access. For more information on Azure Data Lake Storage Gen2, please check out the Azure Documentation.

The node outputs a connection object that allows downstream nodes to access Azure Data Lake Storage Gen2 data as a file system. Since Azure Data Lake Storage Gen2 is built on top of Azure Blob storage, both Connector nodes have similar configuration dialog. For more information, please check out the Azure Blob Storage section.

Azure SQL Database

KNIME Analytics Platform includes a set of database nodes to support connecting to and working with Azure SQL Database.

Setting up KNIME Analytics Platform for Azure SQL has the following prerequisites:

1. An active Azure subscription. For more information on how to create one, please check out the Azure Documentation.
2. Create the SQL database (e.g. single database). For more information on how to create a SQL database on the Azure portal, please follow the Azure Documentation.

Connect to Azure SQL Database

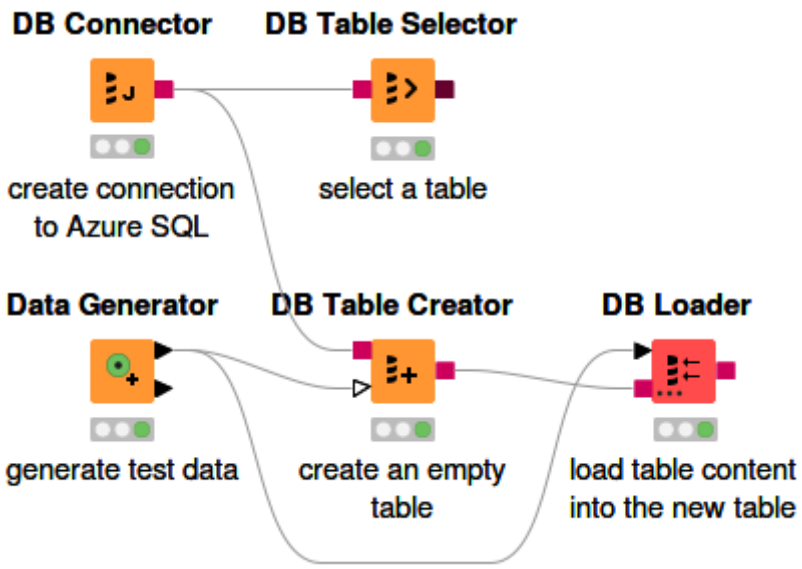

Figure 2. Connecting to Azure SQL database using DB Connector node

The DB Connector node or the Microsoft SQL Server Connector node can be used to connect to Azure SQL. The first step is to install the official driver for Microsoft SQL Server in KNIME Analytics Platform. Please follow the tutorial on how to install the Microsoft SQL Server JDBC driver in KNIME Analytics Platform in the KNIME Database documentation.

The default jTDS for Microsoft SQL Server driver that is bundled with the Microsoft SQL Server Connector node does not support some features, such as the DB Loader node.

Figure 2 shows how to connect to Azure SQL database using DB Connector node. The node configuration dialog is shown in Figure 3. The database URL should look as follow:


```
jdbc:sqlserver://<host>.database.windows.net:<port>;databaseName=<database_name>
```

where:

- `<host>` is the hostname or the name of the server that is created during database creation
- `<port>` is the database port. The default value is 1433.
- `<database_name>` is the name of the created database.

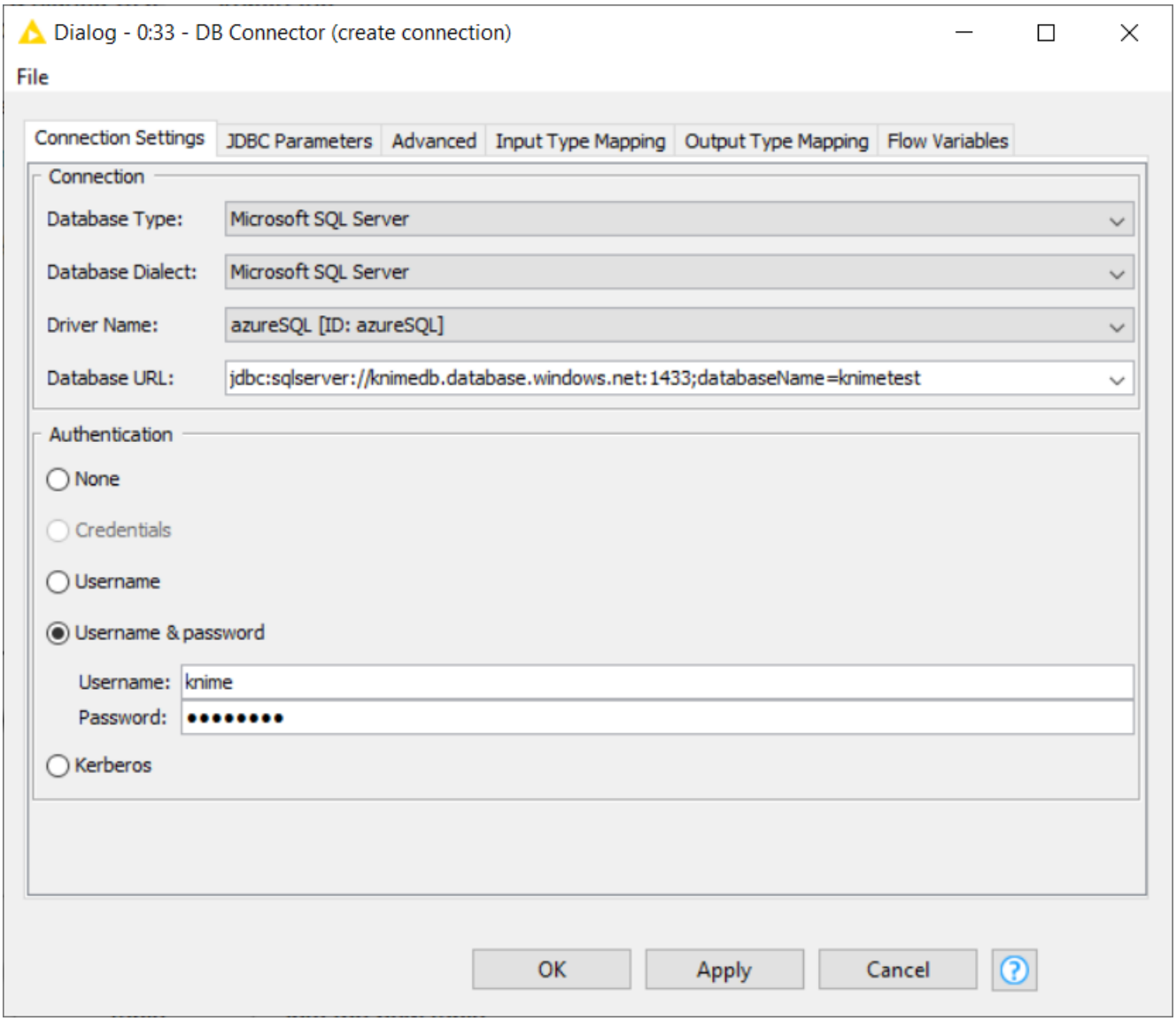

Figure 3. DB Connector node configuration dialog

The database authentication is the server credentials, i.e. server admin login and password.

For more information on the *JDBC parameters* tab or the *Advanced* tab in the node configuration dialog of DB Connector node, please check out the KNIME Documentation.

Executing this node will create a connection to the Azure SQL database and you can use any KNIME database nodes to visually assemble your SQL statements.

For more information on KNIME database nodes, please check out the KNIME Database documentation.

An example workflow to demonstrate the usage of the Microsoft SQL Server Connector node to connect to AzureSQL from within KNIME Analytics Platform is available on KNIME Hub.

Azure HDInsight

Cluster Setup

To create an Azure HDInsight cluster using the Azure portal, follow the step-by-step guide provided by Azure documentation. During cluster creation, the following settings are important:

- Cluster credentials. In this section, you have to give login credentials to access and administer the cluster. Please remember the cluster login username and password, which will be needed later to connect to it via KNIME Analytics Platform.
- Storage Account. A storage account contains all of your Azure Storage objects. The storage account provides a unique namespace for your Azure Storage data that is accessible from anywhere, including a HDInsight cluster.
- Cluster type. The cluster type defines the services that will be provisioned for your cluster. For example, select Apache Spark to enable Spark processing on the cluster.

HDInsight clusters only expose three ports publicly: 22, 23, and 443. For more information on the ports used by Apache Hadoop services running HDInsight clusters, please check out the Azure documentation.

Connect to HDInsight cluster

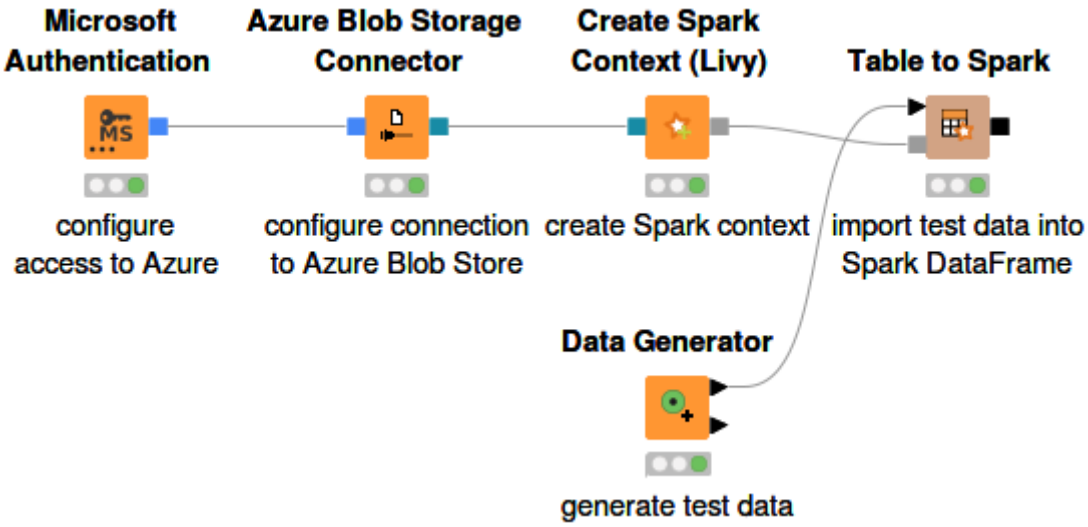

Figure 4. Connecting to HDInsight cluster

Figure 4 shows how to establish a connection to a running HDInsight Spark cluster via KNIME Analytics Platform. The Microsoft Authentication node provides authentication to access

Azure services. The Azure Blob Storage Connector node is used to create a connection to Azure Blob storage. For more details, please check out the Azure Storage section of this guide.

The Create Spark Context (Livy) node creates a Spark context via Apache Livy. Inside the node configuration dialog, the most important settings are:

- Spark version. Please make sure the Spark version is equivalent to the Spark version on the cluster.
- The Livy URL. It has the format `https://<cluster-name>.azurehdinsight.net:443/livy` where `<cluster-name>` is the name of the HDInsight cluster.
- Authentication. Enter the cluster login username and password in this field. Please check the Cluster Setup section to find more information on the cluster credentials.
- Under *Advanced* tab, it is mandatory to set the *staging area for Spark jobs*. The staging area, which is located in the connected Azure Blob storage system, will be used to exchange temporary files between KNIME Analytics Platform and the Spark context.

The remaining settings can be configured according to your needs. For more information on the Create Spark Context (Livy) node, please check out the KNIME Amazon Web Services Integration User Guide.

Once the Spark context is created, you can use any number of the KNIME Spark nodes from the KNIME Extension for Apache Spark to visually assemble your Spark analysis flow to be executed on the cluster.

Apache Hive in Azure HDInsight

This section describes how to establish a connection to Apache Hive™ on Azure HDInsight in KNIME Analytics Platform.

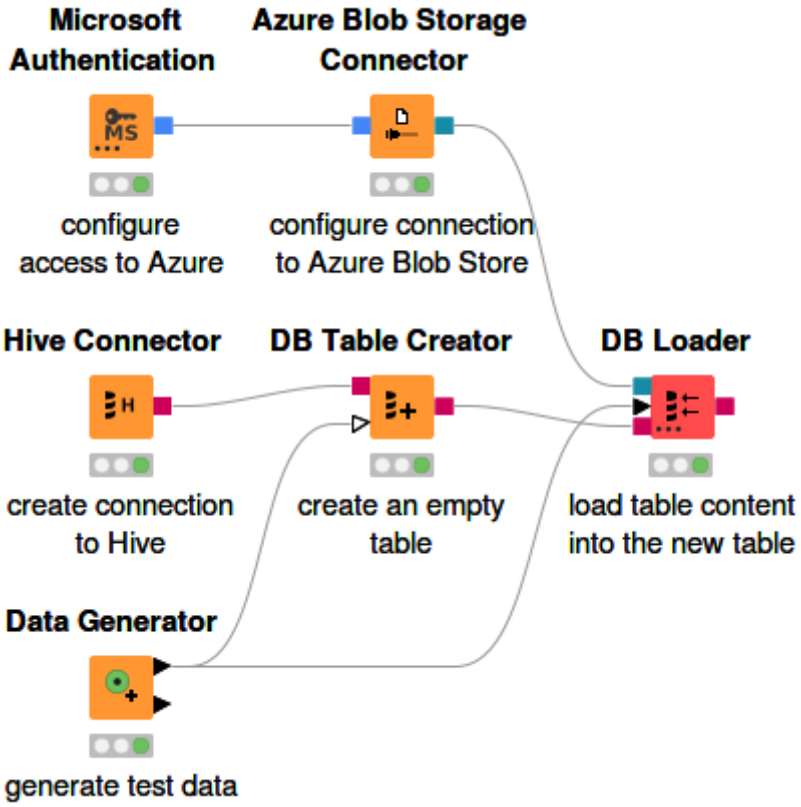

Figure 5. Connect to Hive and create a Hive table

Figure 5 shows how to connect to Hive running on a HDInsight cluster and how to create a Hive table.

The first step is to register the Hive JDBC driver with a custom JDBC URL. Follow the guide on how to register a Hive JDBC driver in KNIME Documentation. However for Hive on Azure HDInsight, enter the following URL template (see Figure 6).

```
jdbc:hive2://<host>:<port>/default;transportMode=http;ssl=1;httpPath=/hive2
```

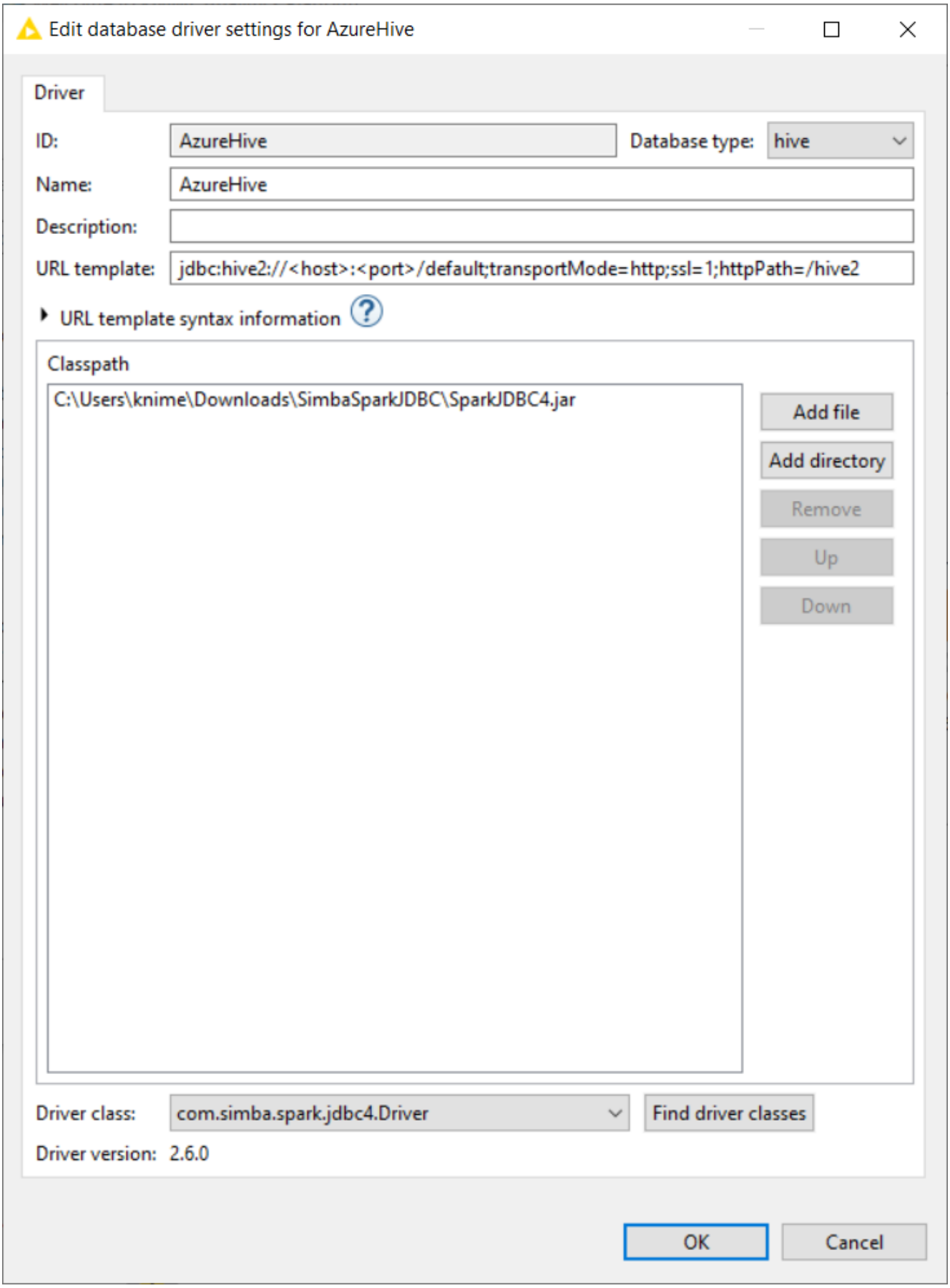

Figure 6. Hive JDBC URL Template

Once the Hive JDBC driver is registered, you can configure the Hive Connector node. The

node configuration dialog is shown in Figure 7, where the hostname is the HDInsight cluster URL, and the credentials are the cluster login username and password (see Connect to HDInsight cluster section for more details). It is very important here to set the port to 443, instead of the usual Hive port 10000 or 10001.

For more information on how to configure the settings in the node configuration dialog, please refer to the KNIME Documentation. Executing the node will create a connection to Apache Hive and you can use any KNIME database nodes to visually assemble your SQL statements.

Please make sure that you set the port to 443, because all connections to the cluster are managed via a secure gateway. This means, you cannot connect directly to Hive server on ports 10001 or 10000, because they are not exposed to the outside of Azure virtual network.

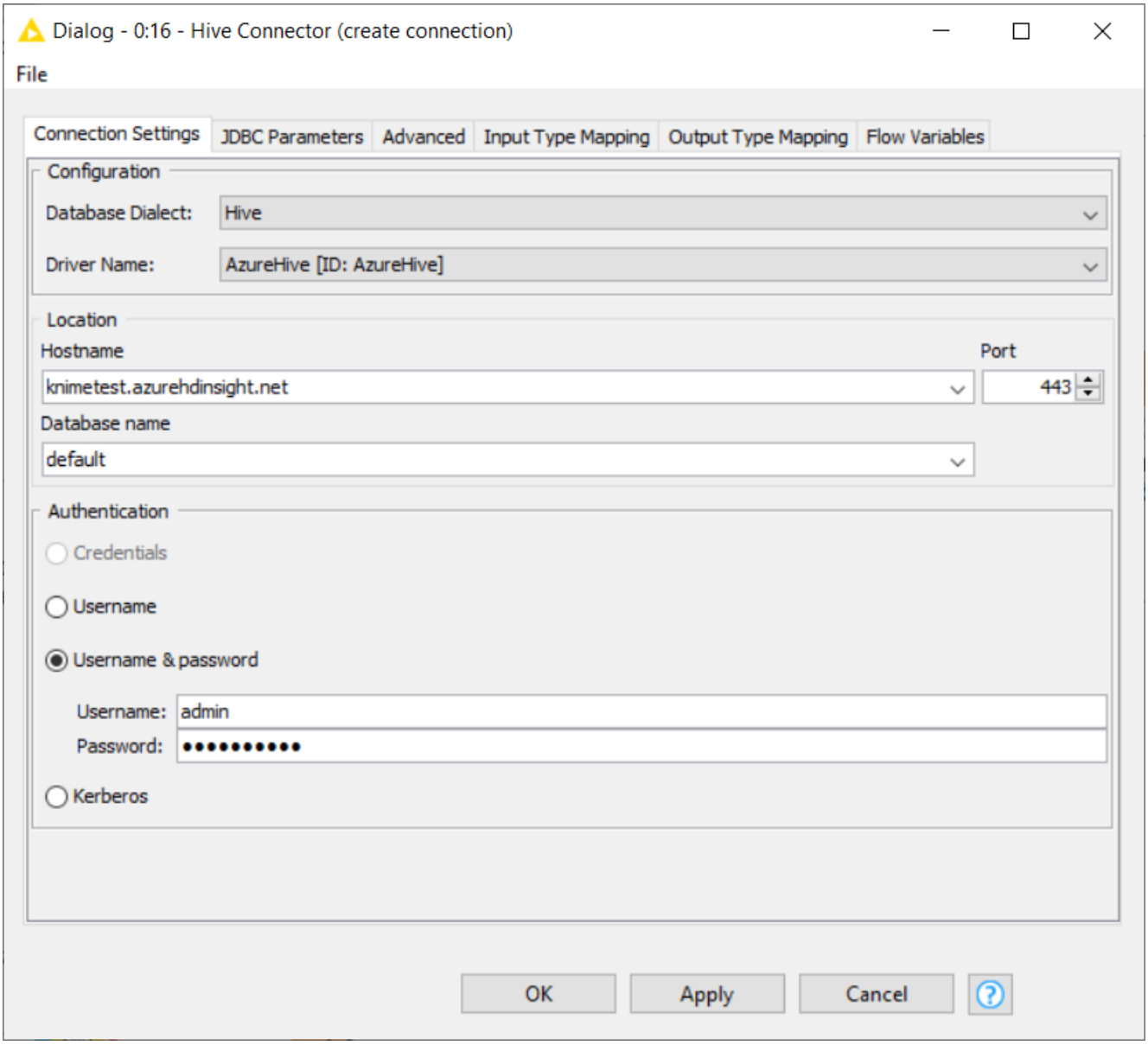

Figure 7. Hive Connector node configuration dialog

An example workflow to demonstrate the usage of HDInsight from within KNIME Analytics Platform is available on KNIME Hub.

KNIME AG
Talacker 50
8001 Zurich, Switzerland
www.knime.com
info@knime.com