KNIME Azure Integration User Guide

KNIME AG, Zurich, Switzerland
Version 4.2 (last updated on 2020-10-26)

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, and Azure SQL.

The KNIME Azure Cloud Connectors extension 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

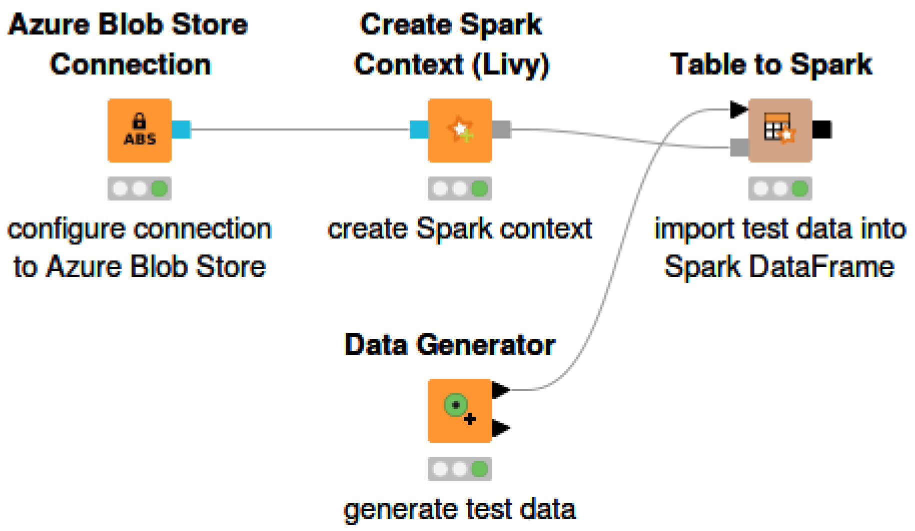

Azure Blob Store Connection
ABS
configure connection to Azure Blob Store
Create Spark Context (Livy)
create Spark context
Table to Spark
import test data into Spark DataFrame
Data Generator
generate test data

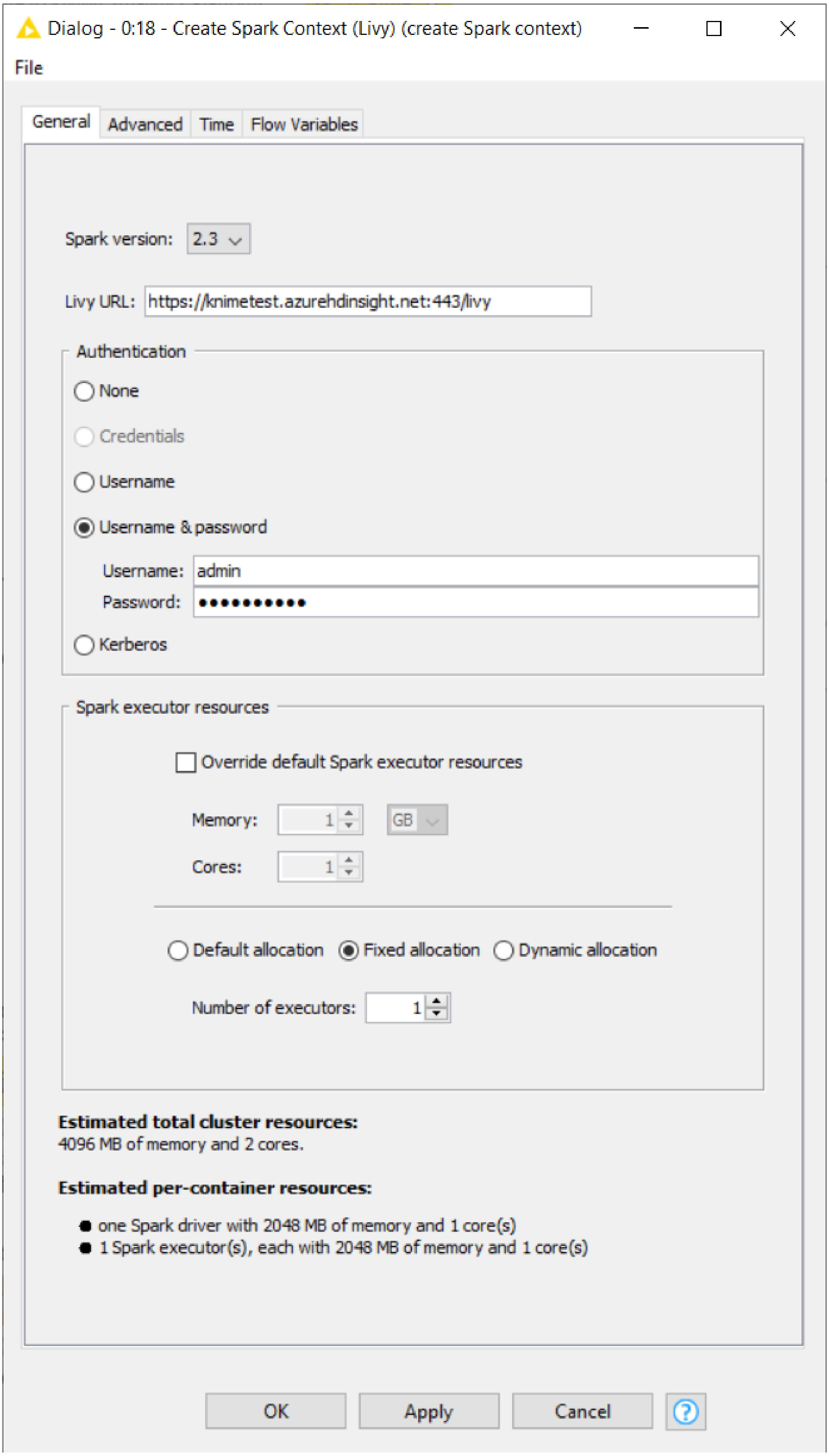
Figure 1. Connecting to HDInsight cluster

Figure 1 shows how to establish a connection to a running HDInsight Spark cluster via KNIME Analytics Platform. The Azure Blob Store Connection node is used to create a connection to Azure Blob storage. For more information on Azure Blob Store Connection node, please check out the Azure Blob 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.

© 2020 KNIME AG. All rights reserved.



Dialog - 0:18 - Create Spark Context (Livy) (create Spark context)

File

General | Advanced | Time | Flow Variables

Spark version: 2.3

Livy URL: https://knimetest.azurehdinsight.net:443/livy

Authentication

○ None

○ Credentials

○ Username

● Username & password

Username: admin

Password: ●●●●●●●●●●

○ Kerberos

Spark executor resources

□ Override default Spark executor resources

Memory: 1 GB

Cores: 1

○ Default allocation ● Fixed allocation ○ Dynamic allocation

Number of executors: 1

Estimated total cluster resources:
4096 MB of memory and 2 cores.

Estimated per-container resources:

- one Spark driver with 2048 MB of memory and 1 core(s)
- 1 Spark executor(s), each with 2048 MB of memory and 1 core(s)

Figure 2. Create Spark Context (Livy) node configuration dialog

© 2020 KNIME AG. All rights reserved.

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.

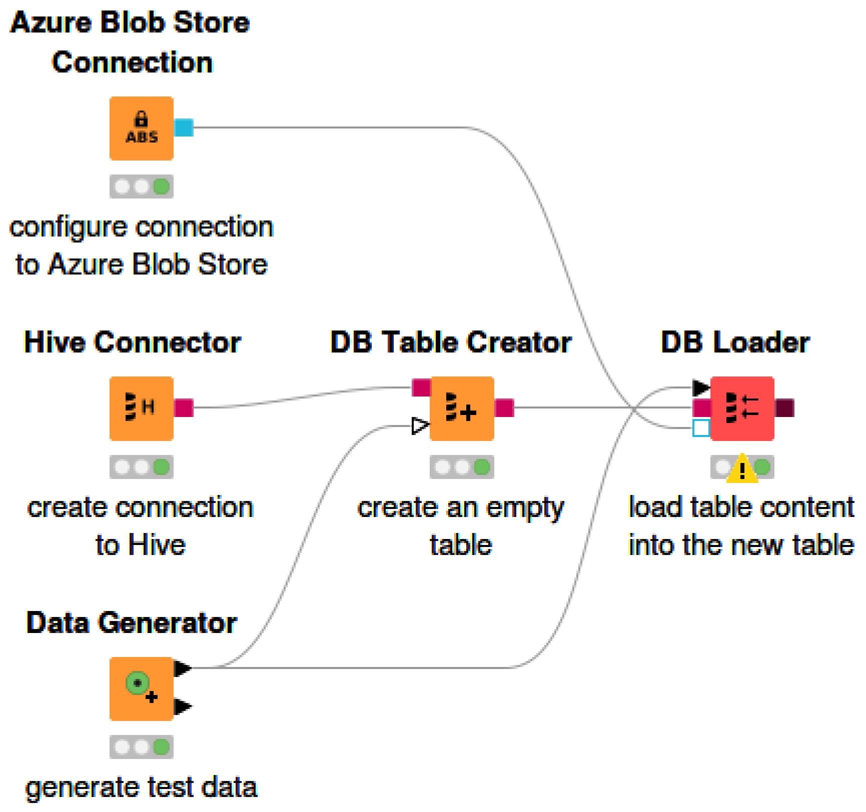

Azure Blob Store Connection
ABS
configure connection to Azure Blob Store
Hive Connector
H
create connection to Hive
DB Table Creator
create an empty table
DB Loader
load table content into the new table
Data Generator
generate test data

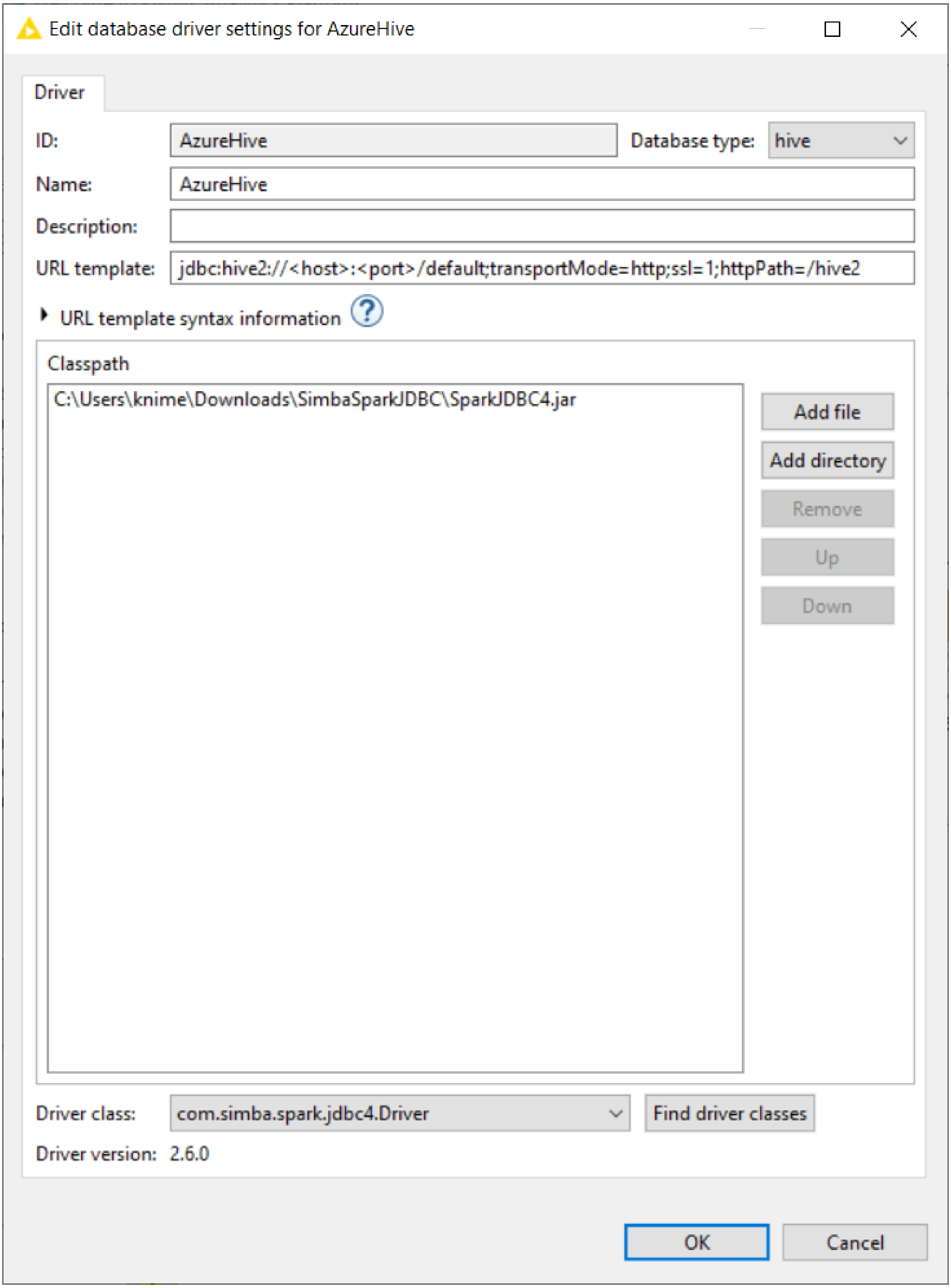
Figure 3. Connect to Hive and create a Hive table

Figure 3 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 4).

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

© 2020 KNIME AG. All rights reserved.



Edit database driver settings for AzureHive

Driver

ID: AzureHive

Database type: hive

Name: AzureHive

Description:

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

URL template syntax information

Classpath

C:\Users\knime\Downloads\SimbaSparkJDBC\SparkJDBC4.jar

Driver class: com.simba.spark.jdbc4.Driver

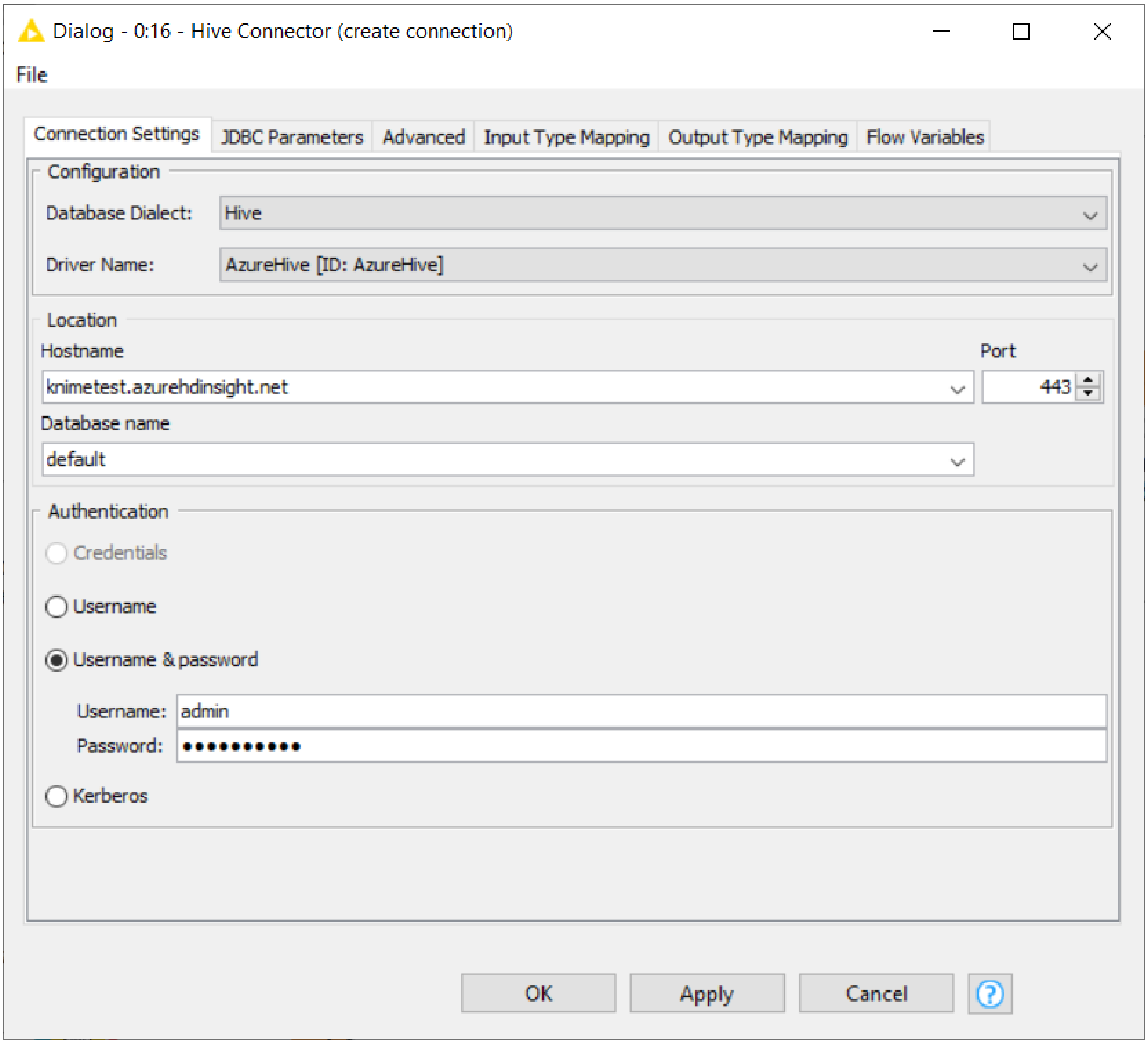
Driver version: 2.6.0

Figure 4. 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 5, 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.



Configuration

Database Dialect: Hive

Driver Name: AzureHive [ID: AzureHive]

Location

Hostname: knimetest.azurehdinsight.net

Port: 443

Database name: default

Authentication

○ Credentials

○ Username

● Username & password

Username: admin

Password: ●●●●●●●●●●

○ Kerberos

Figure 5. Hive Connector node configuration dialog

© 2020 KNIME AG. All rights reserved.

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

© 2020 KNIME AG. All rights reserved.

Azure Blob Storage

KNIME Azure Cloud Connectors extension provides nodes to connect to Azure Blob storage.

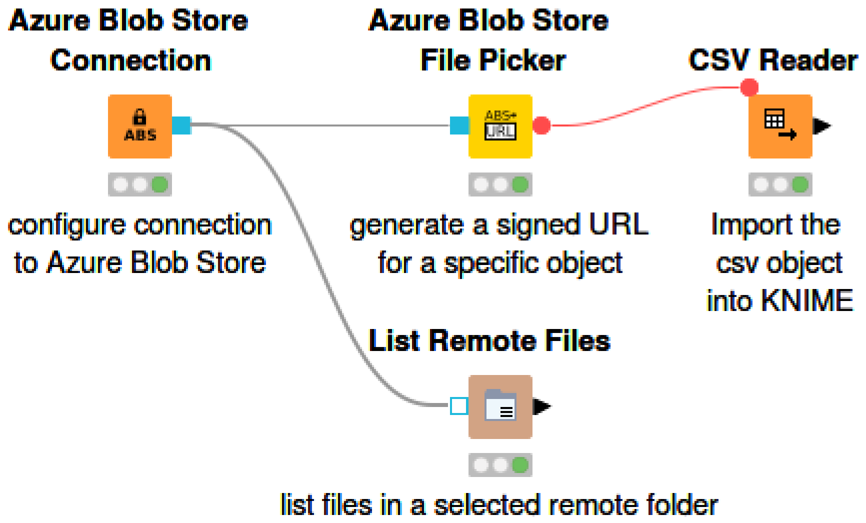

Azure Blob Store Connection
ABS
configure connection to Azure Blob Store
Azure Blob Store File Picker
ABS+ URL
generate a signed URL for a specific object
CSV Reader
Import the csv object into KNIME
List Remote Files
list files in a selected remote folder

Figure 6. Connecting to and working with Azure Blob storage

Figure 6 shows an example on how to connect to Azure Blob storage and work with the remote files.

Azure Blob Store Connection

Azure Blob Store Connection node connects KNIME Analytics Platform with Azure Blob storage. The output of this node is a Azure Blob storage connection information and it allows you to work with your files inside a certain project using the remote file handling nodes. For example, you can create directory, list, delete, download and upload files from and to Azure Blob storage.

Inside the node configuration dialog of the Azure Blob Store Connection node, you need to enter the authentication credentials, i.e. the storage account and its corresponding access key. For more information about storage account, please check out the Storage Account and Access Key section of the Azure documentation.

The connection information is encoded in the format `abs://<storage-account>@blob.core.windows.net`. The protocol is `abs`. The first folder path is the container name and the rest is the blob name, i.e. `abs://<storage-account>@blob.core.windows.net/<container>/<path-to-blob>`.

© 2020 KNIME AG. All rights reserved.

To access Azure Data Lake Storage Gen2 (ADLS Gen2), it should be possible to establish a connection via the Azure Blob Store Connection node. An alternative is to mount the Azure Data Lake Storage Gen2 to Databricks File System (DBFS) and access it using the Databricks File System Connection node.

Azure Blob Store File Picker

The Azure Blob Store File Picker node creates a pre-signed URL for a specific file. This URL can then be used by any of the reader nodes in KNIME Analytics Platform to read the selected file directly from Azure Blob storage, or share it with other users without the need for authentication. Please note that the generated URL is only valid for a specific period of time. Upon expiration, the URL will no longer remain active and an attempt to access the URL will generate an error.

In the node configuration dialog of the Azure Blob Store File Picker node, you have to select:

- The remote file, for which the signed URL should be created
- The expiration time. This defines the duration or UTC time after the signed URL becomes invalid.

The output of this node is a flow variable containing the signed URL pointing to the selected file. You can connect this flow variable to any KNIME file handling node, e.g the CSV Reader node (as shown in Figure 6). Inside the node configuration dialog of the CSV Reader node, simply set the input location using the value from the flow variable.

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

© 2020 KNIME AG. All rights reserved.

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

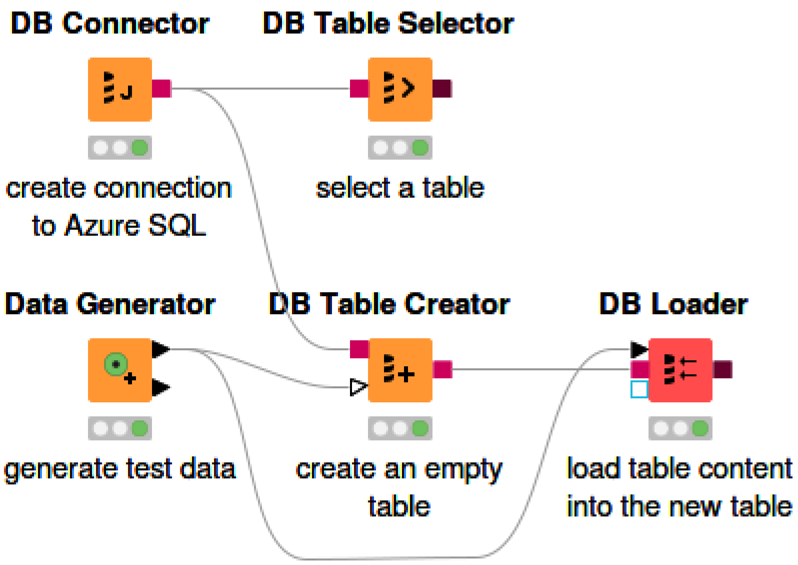


Figure 7. Connecting to Azure SQL database using DB Connector node

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.

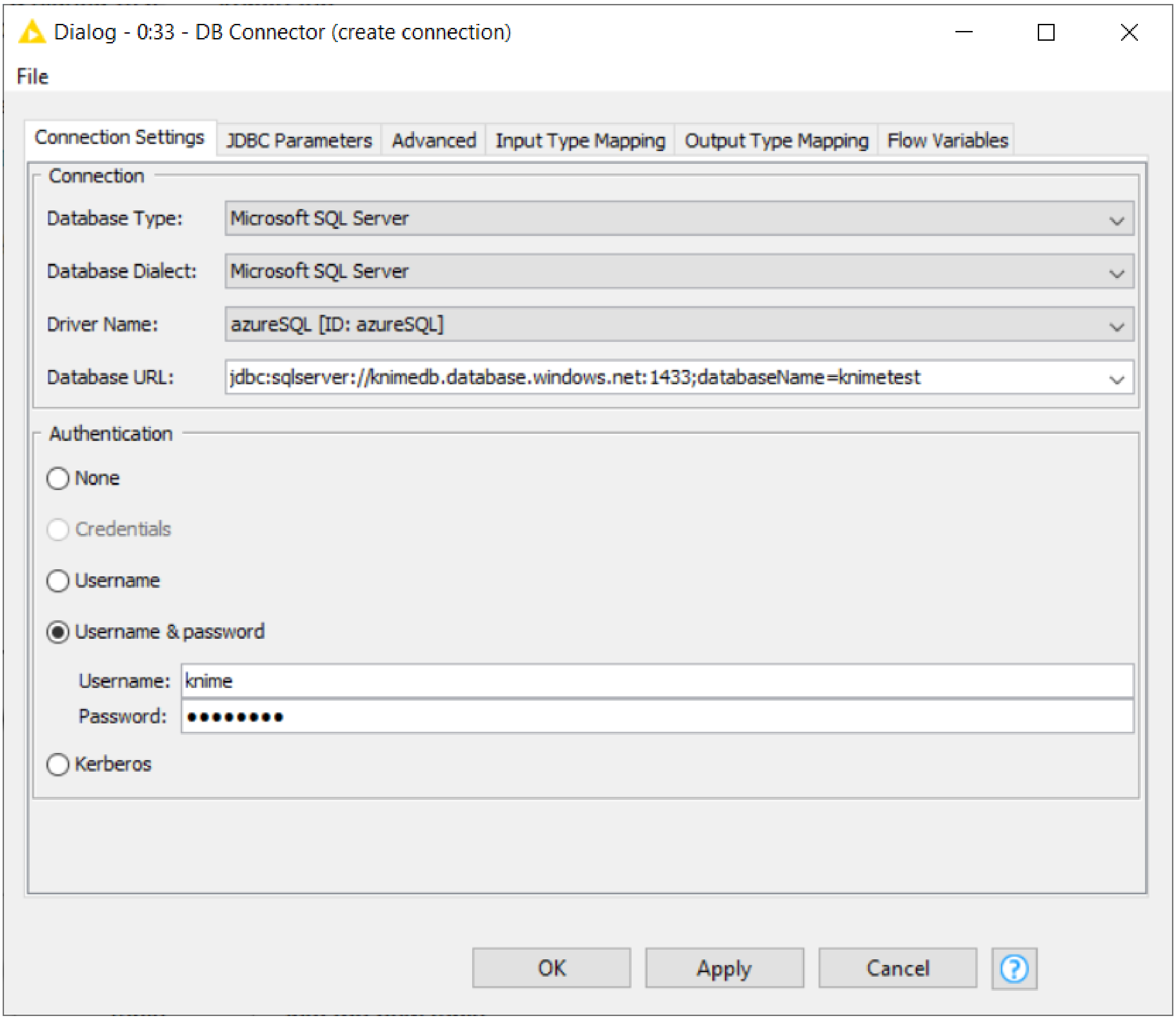
Figure 7 shows how to connect to Azure SQL database using DB Connector node. The node configuration dialog is shown in Figure 8. The database URL should look as follow:

© 2020 KNIME AG. All rights reserved.

```
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.



Connection

Database Type: Microsoft SQL Server

Database Dialect: Microsoft SQL Server

Driver Name: azureSQL [ID: azureSQL]

Database URL: jdbc:sqlserver://knimedb.database.windows.net:1433;databaseName=knimetest

Authentication

○ None

○ Credentials

○ Username

● Username & password

Username: knime

Password: ••••••••

○ Kerberos

Figure 8. DB Connector node configuration dialog

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

© 2020 KNIME AG. All rights reserved.

ℹ 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.

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

ℹ 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.

© 2020 KNIME AG. All rights reserved.

Open for Innovation
KNIME

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

The KNIME® trademark and logo and OPEN FOR INNOVATION® trademark are used by KNIME AG under license from KNIME GmbH, and are registered in the United States. KNIME® is also registered in Germany.