TIBCO ActiveMatrix BusinessWorks™ Plug-in for Big Data User's Guide

Software Release 6.3
December 2016

Two-Second Advantage®

TIBCO®

Important Information

Contents

TIBCO Documentation and Support Services

Documentation for this and other TIBCO products is available on the TIBCO Documentation site. This site is updated more frequently than any documentation that might be included with the product. To ensure that you are accessing the latest available help topics, visit:

https://docs.tibco.com

Product-Specific Documentation

Documentation for TIBCO products is not bundled with the software. Instead, it is available on the TIBCO Documentation site. To directly access documentation for this product, double-click the following file:

TIBCO_HOME/release_notes/TIB_bwpluginbigdata_*version*_docinfo.html

where *TIBCO_HOME* is the top-level directory in which TIBCO products are installed. On Windows, the default *TIBCO_HOME* is `C:\tibco`. On UNIX systems, the default *TIBCO_HOME* is `/opt/tibco`.

The following documents for this product can be found on the TIBCO Documentation site:

- *TIBCO ActiveMatrix BusinessWorks Plug-in for Big Data Installation*
- *TIBCO ActiveMatrix BusinessWorks Plug-in for Big Data User's Guide*
- *TIBCO ActiveMatrix BusinessWorks Plug-in for Big Data Release Notes*

The following documents provide additional information and can be found on the TIBCO Documentation site:

- TIBCO ActiveMatrix BusinessWorks documentation
- *TIBCO Enterprise Administrator User's Guide*

How to Contact TIBCO Support

For comments or problems with this manual or the software it addresses, contact TIBCO Support:

- For an overview of TIBCO Support, and information about getting started with TIBCO Support, visit this site:

  http://www.tibco.com/services/support

- If you already have a valid maintenance or support contract, visit this site:

  https://support.tibco.com

  Entry to this site requires a user name and password. If you do not have a user name, you can request one.

How to Join TIBCO Community

TIBCO Community is an online destination for TIBCO customers, partners, and resident experts. It is a place to share and access the collective experience of the TIBCO community. TIBCO Community offers forums, blogs, and access to a variety of resources. To register, go to the following web address:

https://community.tibco.com

Product Overview

With TIBCO ActiveMatrix BusinessWorks™ Plug-in for Big Data, you can manage big data without coding.

TIBCO ActiveMatrix BusinessWorks™ is a leading integration platform that can integrate a wide variety of technologies and systems within enterprise and on cloud. TIBCO ActiveMatrix BusinessWorks includes an Eclipse-based graphical user interface (GUI) provided by TIBCO Business Studio™ for design, testing, and deployment. If you are not familiar with TIBCO ActiveMatrix BusinessWorks, see the TIBCO ActiveMatrix BusinessWorks documentation for more details.

TIBCO ActiveMatrix BusinessWorks Plug-in for Big Data plugs into TIBCO ActiveMatrix BusinessWorks, bridging TIBCO ActiveMatrix BusinessWorks with Hadoop. Using this plug-in, you can establish a non-code approach to integrate TIBCO ActiveMatrix BusinessWorks with Hadoop family projects, such as Hadoop Distributed File System (HDFS).

The plug-in provides the following common functions:

- Copies files between HDFS and a local file system
- Renames files in HDFS
- Deletes files from HDFS
- Lists the status of a file or directory in HDFS
- Reads data in a file in HDFS
- Writes data to a file in HDFS
- Creates and queues a Hive, MapReduce, or Pig job

Before using this plug-in to run operations, ensure that you have appropriate permissions to access HDFS.

Getting Started

This tutorial is designed for the beginners who want to use TIBCO ActiveMatrix BusinessWorks Plug-in for Big Data in TIBCO Business Studio.

All the operations are performed in TIBCO Business Studio. See TIBCO Business Studio Overview to get familiar with TIBCO Business Studio.

A basic procedure of using TIBCO ActiveMatrix BusinessWorks Plug-in for Big Data includes:

1. Creating a Project
2. Creating an HDFS Connection or Creating an HCatalog Connection
3. Configuring a Process
4. Testing a Process
5. Deploying an Application

Creating a Project

The first task using the plug-in is creating a project. After creating a project, you can add resources and processes.

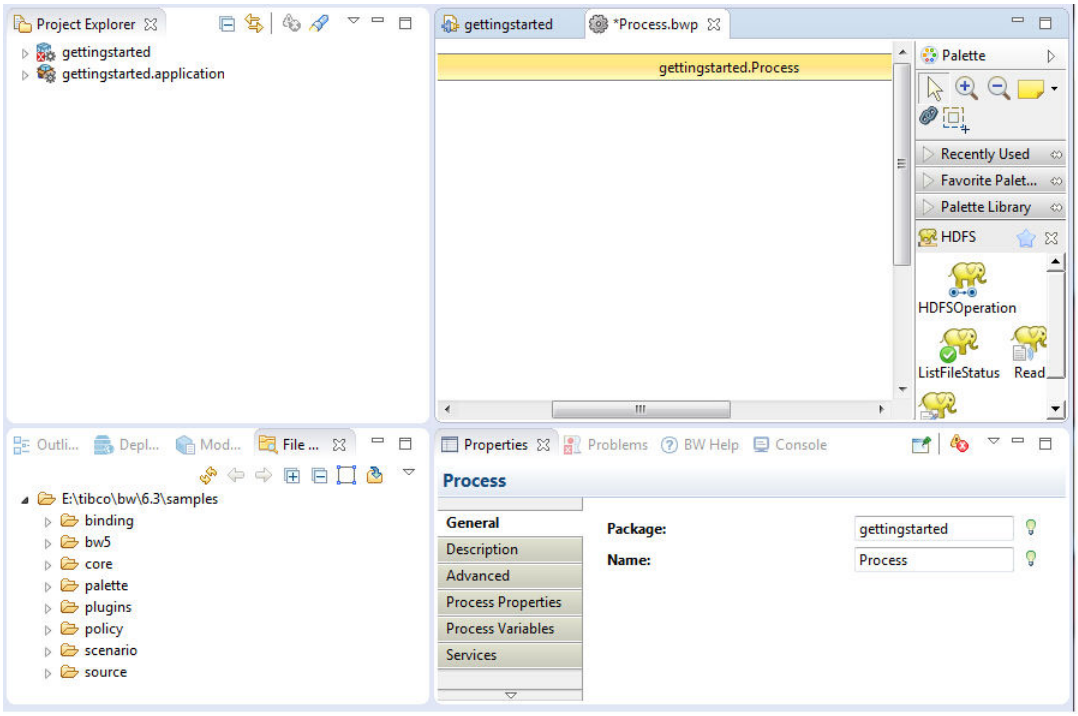
An Eclipse project is an application module configured for TIBCO ActiveMatrix BusinessWorks. An application module is the smallest unit of resources that is named, versioned, and packaged as part of an application.

Procedure

1. Start TIBCO Business Studio using one of the following ways:
   - Microsoft Windows: click **Start** > **All Programs** > **TIBCO** > *TIBCO_HOME* > **TIBCO Business Studio** *version_number* > **Studio for Designers**.
   - Mac OS and Linux: run the TIBCO Business Studio executable file that is located in the *TIBCO_HOME*/studio/*version_number*/eclipse directory.
2. From the menu, click **File** > **New** > **BusinessWorks Resources** to open the BusinessWorks Resource wizard.
3. In the "Select a wizard" dialog, click **BusinessWorks Application Module** and click **Next** to open the New BusinessWorks Application Module wizard.
4. In the Project dialog, configure the project that you want to create:
   a) In the **Project name** field, enter a project name.
   b) By default, the created project is located in the workspace current in use. If you do not want to use the default location for the project, clear the **Use default location** check box and click **Browse** to select a new location.
   c) Use the default version of the application module, or enter a new version in the **Version** field.
   d) Keep the **Create empty process** and **Create Application** check boxes selected to automatically create an empty process and an application when creating the project.
   e) Select the **Use Java configuration** check box if you want to create a Java module.
   A Java module provides the Java tooling capabilities.
   f) Click **Finish** to create the project.

Result

The project with the specified settings is displayed in the Project Explorer view.



Creating an HDFS Connection

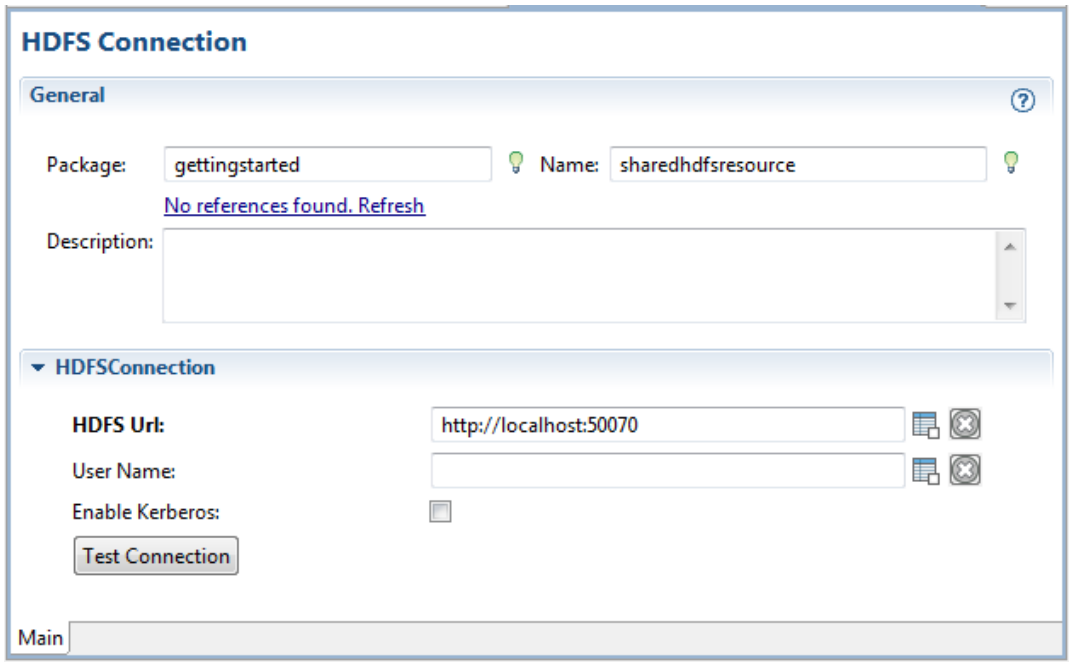
You can create an HDFS Connection shared resource if you want to build a connection between the plug-in and HDFS.

Prerequisites

The HDFS Connection shared resource is available at the **Resources** level. Ensure that you have created a project, as described in Creating a Project.

Procedure

1. Expand the created project in the Project Explorer view.
2. Right-click the **Resources** folder and click **New** > **HDFS Connection**.
3. The resource folder, package name, and resource name of the shared resource are provided by default. You can change these default configurations accordingly. Click **Finish** to open the HDFS Connection shared resource editor.



4. In the **HDFS Url** field, specify the WebHDFS URL or the HttpFS URL that is used to connect to HDFS.
5. In the **User Name** field, specify the unique user name that is used to connect to HDFS.
6. Click **Test Connection** to validate the connection.
7. Optional: If you want to connect to a Kerberized HDFS server, select the **Enable Kerberos** check box, and then set up a connection to the Kerberized HDFS server.

   If your server uses the AES-256 encryption, you must install Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy Files on your machine. For more details, see Installing JCE Policy Files.

   a) From the **Kerberos Method** list, select a Kerberos authentication method:
      - **Keytab**: specify a keytab file to authorize access to HDFS.
      - **Cached**: use the cached credentials to authorize access to HDFS.
      - **Password**: specify a password for the Kerberos principal.

   b) In the **Kerberos Principal** field, specify the Kerberos principal name that is used to connect to HDFS.
   c) If you select **Keytab** as the authentication method, specify the keytab that is used to connect to HDFS in the **Kerberos Keytab** field.
   d) Click **Test Connection** to validate the connection.

Creating an HCatalog Connection

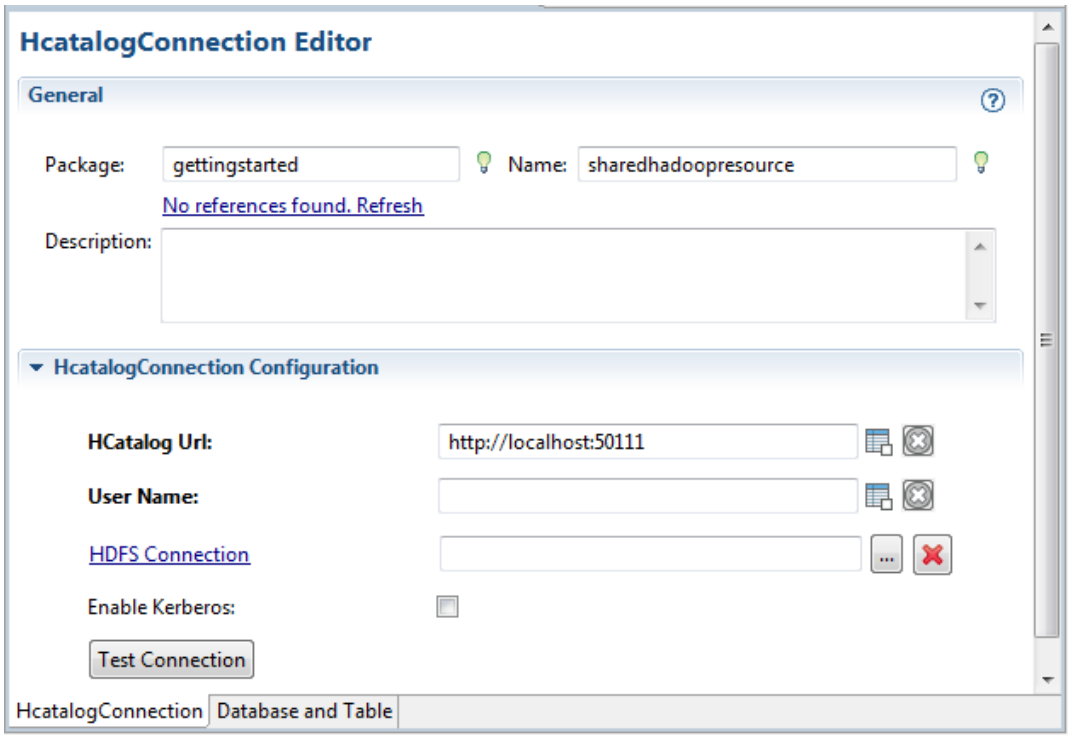
You can create an HCatalog Connection shared resource if you want to build a connection between the plug-in and HCatalog.

Prerequisites

The HCatalog Connection shared resource is available at the **Resources** level. Ensure that you have created a project, as described in Creating a Project.

Procedure

1. Expand the created project in the Project Explorer view.
2. Right-click the **Resources** folder and click **New** > **HCatalog Connection**.
3. The resource folder, package name, and resource name of the shared resource are provided by default. You can change these default configurations accordingly. Click **Finish** to open the HCatalog Connection shared resource editor.



4. In the **HCatalog Url** field, specify the WebHCat URL used to connect to HCatalog.
5. In the **User Name** field, specify the unique user name used to connect to HCatalog.
6. Next to the **HDFS Connection** field, click ... to select an HDFS connection.
7. Click **Test Connection** to validate the connection.
8. Optional: If you want to connect to a Kerberized WebHCat server, select the **Enable Kerberos** check box, and then set up a connection to the Kerberized WebHCat server.

   If your server uses the AES-256 encryption, you must install Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy Files on your machine. For more details, see Installing JCE Policy Files.

   a) From the **Kerberos Method** list, select a Kerberos authentication method:
      - **Keytab**: specify a keytab file to authorize access to WebHCat.
      - **Cached**: use the cached credentials to authorize access to WebHCat.
      - **Password**: specify a password for the Kerberos principal.

   b) In the **Kerberos Principal** field, specify the Kerberos principal name used to connect to WebHCat.
   c) If you select **Keytab** as the authentication method, specify the keytab used to connect to WebHCat in the **Kerberos Keytab** field.
   d) Click **Test Connection** to validate the connection.

What to do next

After an HCatalog connection is established, you can click the **Database and Table** tab to introspect databases and tables existing in HCatalog. See Database and Table for more information.

Connecting to a Kerberos-Enabled Hadoop Server

You can configure the TIBCO ActiveMatrix BusinessWorks™ Plug-in for Big Data to connect to a Kerberos-enabled Hadoop server using any of the three Kerberos methods: **Keytab**, **Cached**, or **Password**.

Prerequisites

You must have the following files available when configuring your plug-in:

- A `keytab` file that is used to identify the principal
- **On Macintosh and Linux**: `krb5.conf` file. This file contains the information about your KDC server configuration such as realm, and the connection properties.
- **On Windows**: `krb5.ini` file

Generate the above files on your KDC server and keep them handy. Refer to the documentation from your vendor for details on how to generate the files.

Copying the keytab and krb5 Files to your system

You must copy the `krb5` files obtained from the KDC server to the following folder on the machine where the Big Data plug-in is installed:

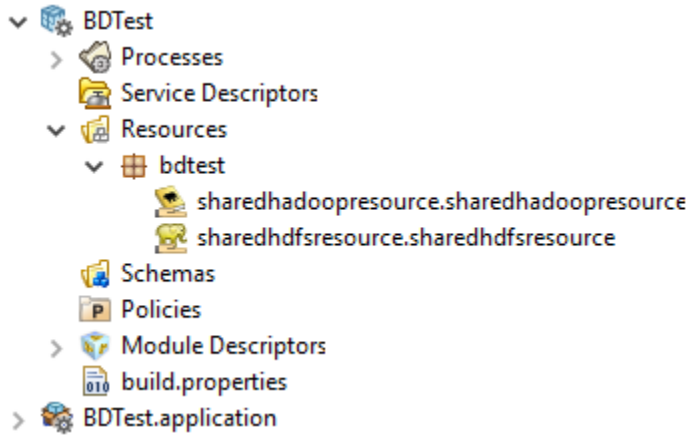
- **On Macintosh and Linux**: `/etc`
- **On Windows:** `C:\Windows`

Select one of the following Kerberos methods in TIBCO Business Studio™ to connect to a Kerberos-enabled Hadoop server.

Using the Keytab Method

Follow these steps to use the Keytab method:

1. In **Project Explorer**, fully expand the application module, and double-click the HCatalog Connection shared resource under the **Resources** folder to open the **HcatalogConnection Editor** in the right pane.



2. Select the **Enable Kerberos** checkbox.
3. Select **Keytab** from the **Kerberos Method** drop-down menu.
4. Enter the **Kerberos Principal** in its text box.
5. Enter the path to the Keytab file on your system or navigate to it using the button.
6. Test the connection using the **Test Connection** button.
7. Repeat the same steps for the HDFS Connection shared resource under the **Resources** folder.

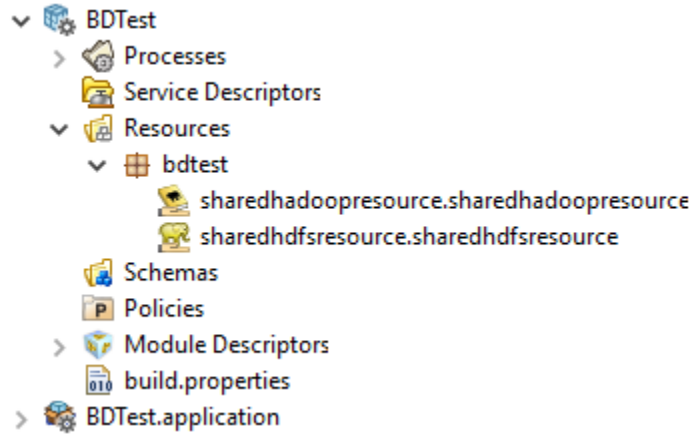
Using the Cached Method

Make sure that you have the keytab file handy before following this procedure. You will need to cache the keytab file in your environment on the machine where the Big Data plug-in is installed.

Follow these steps to use the Cached method:

1. Open a command prompt and run the following command:`kinit -kt <keytab-filename> <principal-name>`

   This utility caches the keytab file in your environment.
2. After the command completes, run `klist` to make sure that the `keytab` file has been properly cached. When successful, the `klist` command outputs the details of the principal associated with the `keytab` file.
3. In **Project Explorer**, fully expand the application module, and double-click the HCatalog Connection shared resource under the **Resources** folder to open the **HcatalogConnection Editor** in the right pane.

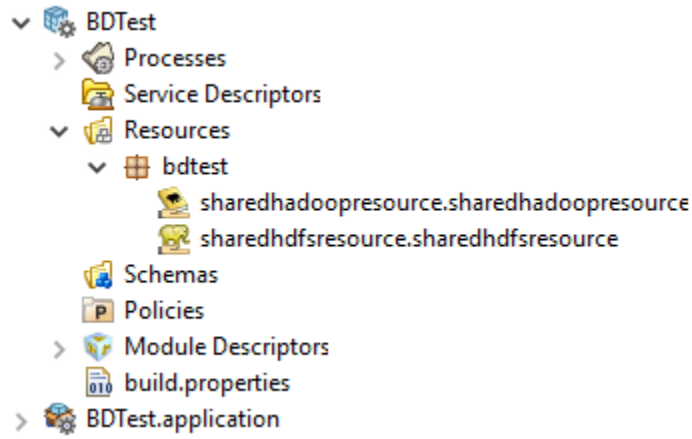


4. Select the **Enable Kerberos** checkbox.
5. Select **Cached** from the **Kerberos Method** drop-down menu.
6. Enter the **Kerberos Principal** in its text box.
7. Test the connection using the **Test Connection** button.
8. Repeat the same steps for the HDFS Connection shared resource under the **Resources** folder.

Using the Password Method

The **Password** method does not require the `keytab` file. Follow these steps to use the Password method:

1. In **Project Explorer**, fully expand the application module, and double-click the HCatalog Connection shared resource under the **Resources** folder to open the **HcatalogConnection Editor** in the right pane.



2. Select the **Enable Kerberos** checkbox.
3. Select **Password** from the **Kerberos Method** drop-down menu.
4. Enter the **Kerberos Principal** in its text box.
5. Enter the password for the Kerberos principal in the **Kerberos Password** text box.
6. Test the connection using the **Test Connection** button.
7. Repeat the same steps for the HDFS Connection shared resource under the **Resources** folder.

Configuring a Process

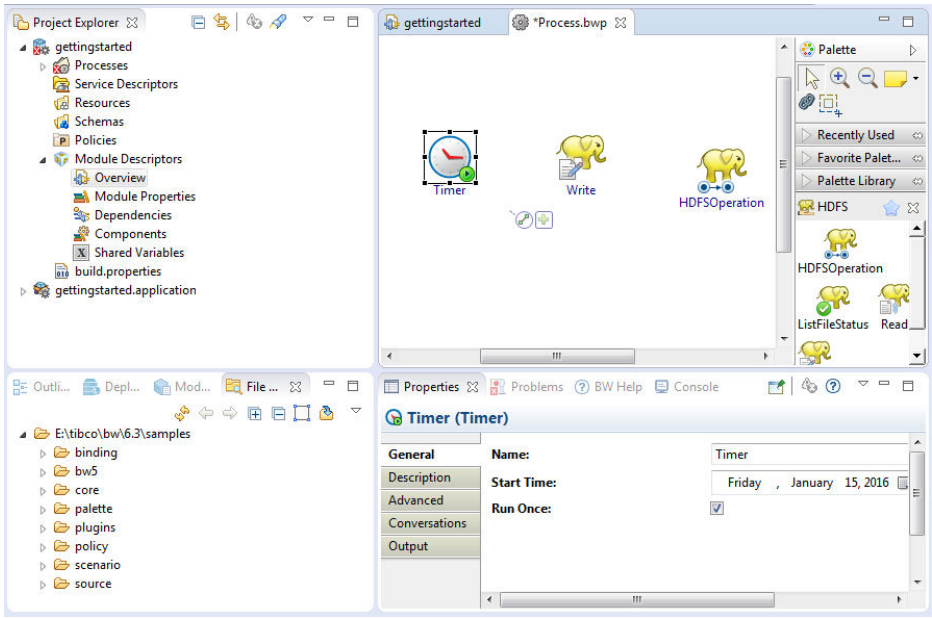
After creating a project, an empty process is created. You can add activities to the empty process to complete a task, for example, writing data to a specified file in HDFS.

Prerequisites

Ensure that you have created an empty process after Creating a Project.

Procedure

1. In the Project Explorer view, click the created project and open the empty process from the **Processes** folder.
2. Select an activity from the Palette view and drop it in the Process editor.
   For example, select and drop the Timer activity from the General Activities palette and the Write and HDFSOperation activities from the HDFS palette.



3. Drag the icon to create a transition between the added activities.
4. Configure the added activities, as described in HDFS Palette or Hadoop Palette.

- An HDFS connection is required when configuring the HDFS activities. See Creating an HDFS Connection for details about how to create an HDFS connection.
- An HCatalog connection is required when configuring the HCatalog activities. See Creating an HCatalog Connection for details about how to create an HCatalog connection.

5. Click **File** > **Save** to save the project.

Testing a Process

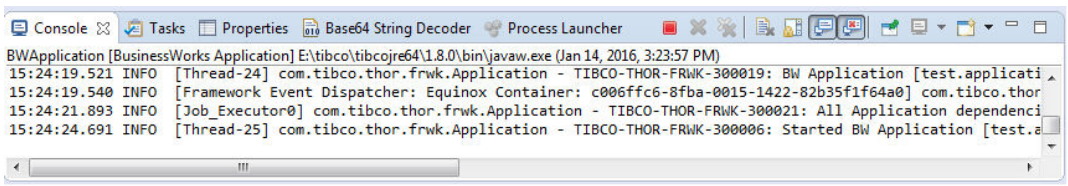
After configuring a process, you can test the process to check if the process completes your task.

Prerequisites

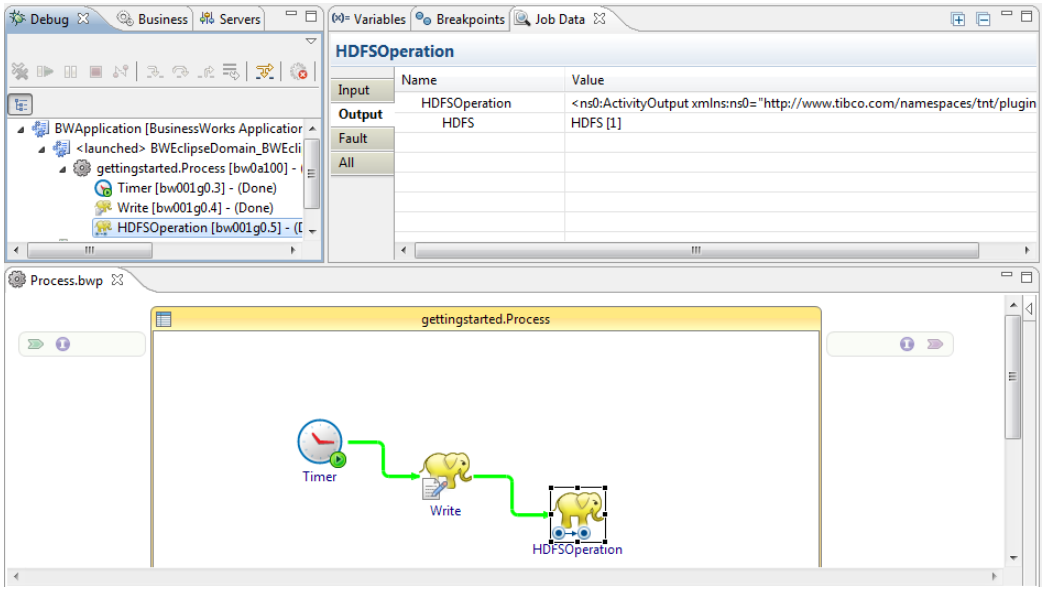
Ensure that you have configured a process, as described in Configuring a Process.

Procedure

1. On the toolbar, click **Debug** > **Debug Configurations**.
2. Click **BusinessWorks Application** > **BWApplication** in the left panel.
   By default, all the applications in the current workspace are selected in the **Applications** tab. Ensure that only the application you want to debug is selected in the **Applications** tab in the right panel.
3. Click **Debug** to test the process in the selected application.
   TIBCO Business Studio changes to the Debug perspective. The debug information is displayed in the Console view.



4. In the **Debug** tab, expand the running process and click an activity.
5. In the upper-right corner, click the **Job Data** tab, and then click the **Output** tab to check the activity output.



Deploying an Application

After testing, if the configured process works as expected, you can deploy the application that contains the configured process into a runtime environment, and then use the **`bwadmin`** utility to manage the deployed application.

Before deploying an application, you must generate an application archive, which is an enterprise archive (EAR) file that is created in TIBCO Business Studio. For more information about TIBCO Business Studio, see TIBCO Business Studio Overview.

Deploying an application involves the following tasks:

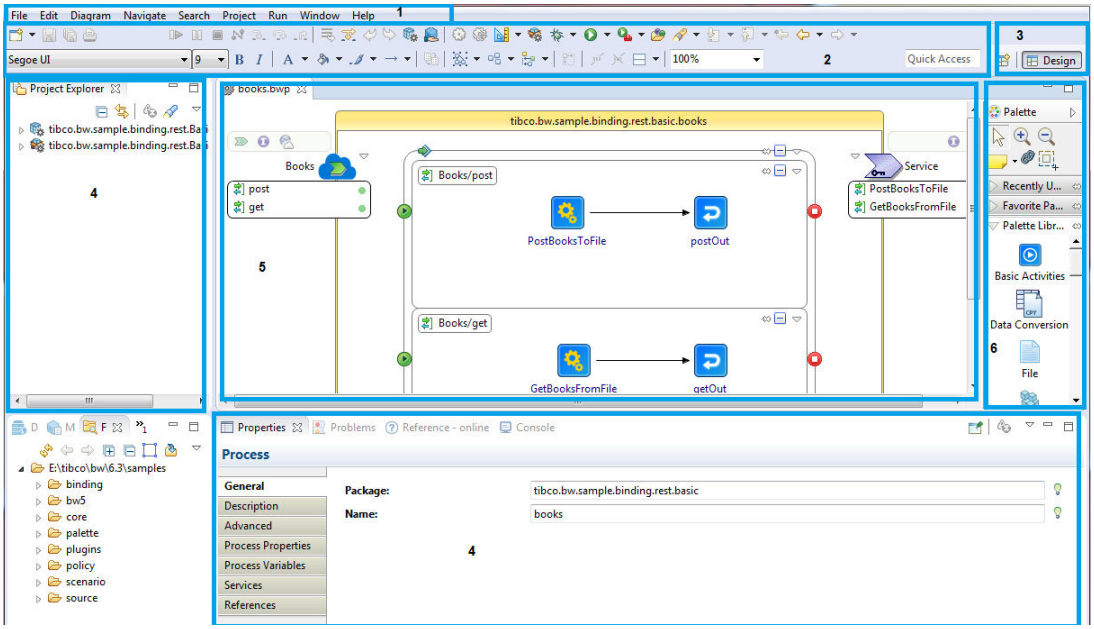
1. Uploading an application archive.
2. Deploying an application archive.
3. Starting an application.

See *TIBCO ActiveMatrix BusinessWorks Administration* for more details about how to deploy an application.

TIBCO Business Studio Overview

TIBCO Business Studio is an Eclipse-based integration development environment that is used to design, develop, and test ActiveMatrix BusinessWorks applications.

TIBCO Business Studio provides a workbench in which you can create, manage, and navigate resources in your workspace. A *workspace* is the central location on your machine where all data files are stored.



The workbench consists of:

1. **Menu**: contains menu items such as File, Edit, Diagram, Navigate, Search, Project, Run, Window, and Help.
2. **Toolbar**: contains buttons for frequently used commands such as New, Save, Enable/Disable Business Studio Capabilities, Create a new BusinessWorks Application Module, Create a new BusinessWorks Shared Module, Debug, Run, and so on.
3. **Perspective**: contains an initial set and layout of views that are required to perform a certain task. TIBCO Business Studio launches the Modeling perspective by default. You can change the perspective from the menu **Window** > **Open Perspective** > *Perspective_Name*.
4. **View**: displays resources. For example, the Project Explorer view displays the ActiveMatrix BusinessWorks applications, modules, and other resources in your workspace; the Properties view displays the properties for the selected resource. You can open a view from the menu **Window** > **Show View** > *View_Name*.
5. **Editor**: provides a canvas to configure, edit, or browse a resource. Double-click a resource in a view to open the appropriate editor for the selected resource. For example, double-click an ActiveMatrix BusinessWorks process (`DemoWorkflow.bwp`) in the Project Explorer view to open the process in the editor.
6. **Palette**: contains a set of widgets and a palette library. A *palette* groups activities that perform similar tasks and provides quick access to activities when configuring a process.

Shared Resources

You can use the HDFS Connection shared resource to connect the plug-in with HDFS, and use the HCatalog Connection shared resource to connect the plug-in with HCatalog.

HDFS Connection

The HDFS Connection shared resource contains all necessary parameters to connect to HDFS. It can be used by the HDFS Operation, ListFileStatus, Read, Write activities, and the HCatalog Connection shared resource.

General

In the General panel, you can specify the package that stores the HDFS Connection shared resource and the shared resource name.

The following table lists the fields in the General panel of the HDFS Connection shared resource:

Field	Module Property?	Description
Package	No	The name of the package where the shared resource is located.
Name	No	The name as the label for the shared resource in the process.
Description	No	A short description for the shared resource.

HDFSConnection

In the HDFSConnection Configuration panel, you can provide necessary information to connect the plug-in with HDFS. You can also connect to a Kerberized HDFS server.

The following table lists the fields in the HDFSConnection panel of the HDFS Connection shared resource:

Field	Module Property?	Description

HDFS Url	Yes	The WebHDFS URL that is used to connect to HDFS. The default value is `http://localhost:50070`.
		The plug-in supports HttpFS and HttpFS with SSL. You can enter an HttpFS URL with HTTP or HTTPS in this field. For example:
		`http://httpfshostname:14000`
		`https://httpfshostname:14000`
		  To set up high availability for your cluster, enter two comma-separated URLs in this field. Make sure that there are no spaces in between the comma and the second URL. The plug-in designates the first entry to be the primary node and the second entry to be the secondary node.

User Name	Yes	The unique user name that is used to connect to HDFS.

Field	Module Property?	Description
Enable Kerberos	No	If you want to connect to a Kerberized WebHCat server, you can select this check box.  If your server uses the AES-256 encryption, you must install Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy Files on your machine. For more details, see Installing JCE Policy Files.
Kerberos Method	No	The Kerberos authentication method that is used to authorize access to HDFS. Select an authentication method from the list: • **Keytab**: specify a keytab file to authorize access to HDFS. • **Cached**: use the cached credentials to authorize access to HDFS. • **Password**: enter the name of the Kerberos principal and a password for the Kerberos principal. This field is displayed only when you select the **Enable Kerberos** check box.
Kerberos Principal	Yes	The Kerberos principal name that is used to connect to HDFS. This field is displayed only when you select the **Enable Kerberos** check box.
Kerberos Password	Yes	The password for the Kerberos principal. This field is displayed only when you select the **Password** from the **Kerberos Method** list.
Kerberos Keytab	Yes	The keytab that is used to connect to HDFS. This field is displayed only when you select **Keytab** from the **Kerberos Method** list.

Test Connection

You can click **Test Connection** to test whether the specified configuration fields result in a valid connection.

Setting up High Availability

You can set up high availability for your cluster in this panel. To do so, enter two URLs as comma-separated values (no space between the comma and the second URL) in the **HDFS Url** field under the **HDFS Connection** section of this panel. The plug-in designates the first entry to be the primary node and the second entry to be the secondary node. The plug-in automatically connects and routes the request to the secondary node in the event that the primary node goes down.

To check the status of a node, use the API, *<HDFS URL>*`/jmx?qry=Hadoop:service=NameNode,name=NameNodeStatus`, For example, `http://cdh571.na.tibco.com:50070/jmx?qry=Hadoop:service=NameNode,name=NameNodeStatus`

HCatalog Connection

The HCatalog Connection shared resource contains all the necessary parameters to connect to HCatalog. It can be used by the Hive, MapReduce, Pig, and WaitForJobCompletion activities.

General

In the General panel, you can specify the package that stores the HCatalog Connection shared resource and the shared resource name.

The following table lists the fields in the General panel of the HCatalog Connection shared resource:

Field	Module Property?	Description
Package	No	The name of the package where the shared resource is located.
Name	No	The name as the label for the shared resource in the process.
Description	No	A short description for the shared resource.

HCatalogConnection Configuration

In the HCatalogConnection Configuration panel, you can provide necessary information to connect the plug-in with HCatalog. You can also connect to a Kerberized WebHCat server.

The following table lists the fields in the HCatalogConnection Configuration panel of the HCatalog Connection shared resource:

Field	Module Property?	Description
HCatalog Url	Yes	The WebHCat URL that is used to connect to HCatalog. The default value is `http://localhost:50111`.
User Name	Yes	The unique user name that is used to connect to HCatalog.
HDFSConnection	No	The HDFS Connection shared resource that is used to create a connection between the plug-in and HDFS. Click ... to select an HDFS Connection shared resource. If no matching HDFS Connection shared resources are found, click **Create Shared Resource** to create one. For more details, see Creating an HDFS Connection.
Enable Kerberos	No	If you want to connect to a Kerberized WebHCat server, you can select this check box.  If your server uses the AES-256 encryption, you must install Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy Files on your machine. For more details, see Installing JCE Policy Files.

Field	Module Property?	Description
Kerberos Method	No	The Kerberos authentication method that is used to authorize access to WebHCat. Select an authentication method from the list: • **Keytab**: specify a keytab file to authorize access to WebHCat. • **Cached**: use the cached credentials to authorize access to WebHCat. • **Password**: enter the name of the Kerberos principal and a password for the Kerberos principal. This field is displayed only when you select the **Enable Kerberos** check box.
Kerberos Principal	Yes	The Kerberos principal name that is used to connect to WebHCat. This field is displayed only when you select the **Enable Kerberos** check box.
Kerberos Password	Yes	The password for the Kerberos principal. This field is displayed only when you select the **Password** from the **Kerberos Method** list.
Kerberos Keytab	Yes	The keytab that is used to connect to WebHCat. This field is displayed only when you select **Keytab** from the **Kerberos Method** list.

Test Connection

You can click **Test Connection** to test whether the specified configuration fields result in a valid connection.

Database and Table

In **Database and Table Editor**, you can introspect the databases and tables existing in HCatalog.

Creating or modifying databases or tables are not supported by the plug-in in this release.

You can perform the following operations on databases or tables:

Introspecting Databases

If you want to introspect all the databases that currently exist in the specified HCatalog server, click **Introspect Database**. When you select one of them from the database list, the plug-in generates a database with definitions. See Database for more information about the introspected database.

Removing Databases or Tables

If you want to remove a selected database or table from the **Database** list, click **Remove**.

Introspecting Tables

If you want to introspect all the tables that currently exist in a specified database, click **Introspect Table**. When you select one of them from the table list, the plug-in generates a table with definitions. See Table for more information about the introspected table.

Generating Table Schema

If you want to generate table schema, click **Generate Schema**. You can then enter a name for the schema file, which is stored in the Schemas folder in your project.

Database

Database is an administrative container for a set of tables.

Configuration

In the Configuration panel, you can view detailed information of an introspected database.

The following table lists the fields in the Configuration panel of a database:

Field	Description
Name	The name of the database.
Description	A short description of the database.
Database Location	The directory where the database is located.
Database Comment	The comment for the database.
Database Properties	Properties of the database. A property is associated with a name and a value.

Table

Table provides shared virtual storage for data. It is a shared entity that can be accessed by multiple applications concurrently, each one of these has the same coherent view of the data contained in the table.

Configuration

In the Configuration panel, you can view detailed information of an introspected table.

The following table lists the fields in the Configuration panel of a table:

Field	Description
Name	The name of the table.
Description	A short description of the table.
Comment	The comment for the table.
Location	The directory where the table is located.

Field	Description
Table Columns	The columns of the table. A column is associated with a name, a type, an advanced type, and a comment.  The AdvancedType column indicates a complex data type.

Advanced

In the Advanced panel, you can view advanced configurations of the introspected table.

The following table lists the fields in the Advanced panel of the table:

Field	Description
Partitioned	If this check box is selected, the table is partitioned. Otherwise, the table is not partitioned.
Output Format	The output format of the table.
Owner	The owner of the table.
Input Format	The input format of the table.
Permission	The permission of the table.
Group	The group that is associated with the table.
Partition Columns	The partition columns of the table. A partition column is associated with a name, a type, and a comment.
Table Properties	The properties of the table. A table property is associated with a name and a value.

Palettes

A palette groups together the activities that connect to the same external applications. You can use the activities that are contained in a palette to design your business processes.

The following two palettes are available after installing TIBCO ActiveMatrix BusinessWorks Plug-in for Big Data:

- HDFS Palette
- Hadoop Palette

HDFS Palette

With the HDFS palette, you can perform some operations on the files in HDFS.

You can use the following activities in the HDFS palette:

- HDFSOperation
- ListFileStatus
- Read
- Write

HDFSOperation

You can use the HDFSOperation activity to do basic operations on files in HDFS, including copying files between HDFS and a local system, renaming files in HDFS, and deleting files from HDFS.

General

In the **General** tab, you can specify the activity name in the process, establish a connection to HDFS, and select the specific HDFS operation that you want to perform.

The following table lists the configurations in the **General** tab of the HDFSOperation activity:

Field	Module Property?	Description
Name	No	The name to be displayed as the label for the activity in the process.
HDFSConnection	Yes	The HDFS Connection shared resource that is used to create a connection between the plug-in and HDFS. Click to select an HDFS Connection shared resource. If no matching HDFS Connection shared resources are found, click **Create Shared Resource** to create one. For more details, see Creating an HDFS Connection.

Field	Module Property?	Description
HDFSOperation	No	The HDFS operation that you want to perform. Select a HDFS operation from the list: • **PUT_LOCAL_TO_HDFS**: copy local files or folders to HDFS. • **GET_HDFS_TO_LOCAL**: copy files or folders from HDFS to a local file system. • **RENAME_HDFS**: rename files in HDFS. • **DELETE_FROM_HDFS**: delete files from HDFS.

Description

In the **Description** tab, you can enter a short description for the HDFSOperation activity.

Input

In the **Input** tab, you can configure the HDFS operation that you select in the **General** tab. The input elements of the HDFSOperation activity vary depending on the HDFS operation that you select in the **General** tab.

The following table lists all the possible input elements in the **Input** tab of the HDFSOperation activity:

For the **PUT_LOCAL_TO_HDFS** and **GET_HDFS_TO_LOCAL** operations, you can optionally copy all the files in a source folder to a destination folder. To do so, provide the full folder path in both **sourceFilePath** and **destinationFilePath**.

Input Item	Data Type	Description
`HDFS`	Complex	The HDFS operation configuration. This element contains the elements from `sourceFilePath` to `recursive` that are listed in this table.
`sourceFilePath`	String	The path of the source file. Alternatively, to copy multiple files to a destination folder, you can provide the path to the folder that contains the source file(s). The plug-in will automatically copy all the files in the source folder to the destination folder that you specify in **destinationFilePath**.
`destination FilePath`	String	The path of the destination file or folder into which you want the source files copied.
`overwrite`	Boolean	If a file that has the same name already exists in the specified destination path, yo can specify whether you want to overwrite the existing file , `1` (true) or `0` (false).
`blockSize`	Long	The block size of the file. The value in this field must be greater than 0.
`replication`	Short	The number of replications of the file. The value in this field must be greater than 0.

Input Item	Data Type	Description
permission	Integer	The permission of the file. The value in this field must be in the range 0 - 777.
offset	Long	The starting byte position. The value in this field must be 0 or greater.
length	Long	The number of bytes to be processed.
bufferSize	Integer	The size of the buffer that is used in transferring data. The value in this field must be greater than 0.
recursive	Boolean	You can specify whether you want to operate on the content in the subdirectories, 1 (true) or 0 (false).
timeout	Long	The amount of time, in milliseconds, to wait for this activity to complete. By default, this activity does not time out if you do not specify a value.

Output

In the **Output** tab, you can view whether the execution is successfully.

The following table lists the output elements in the **Output** tab of the HDFSOperation activity:

Output Item	Data Type	Description
HDFS	Complex	The execution of HDFS operation. This element contains the **status** and the **msg** elements.
status	Integer	A standard HTTP status code that indicates whether the execution is successful.
msg	String	The execution message.

Fault

In the **Fault** tab, you can view the error code and error message of the HDFSOperation activity. See Error Codes for more detailed explanation of errors.

The following table lists the error schema elements in the **Fault** tab of the HDFSOperation activity:

Error Schema Element	Data Type	Description
msg	String	The error message description that is returned by the plug-in.
msgCode	String	The error code that is returned by the plug-in.
exception	String	The exception occurs when the plug-in has internal errors.

Error Schema Element	Data Type	Description
`message`	String	The error message that is returned by the server.
`javaClassName`	String	The name of the Java class where an error occurs.

ListFileStatus

You can use the ListFileStatus activity to list the status of a specified file or directory in HDFS.

Regarding a specified directory, the activity returns the status of all the files and directories in the specified directory.

General

In the **General** tab, you can specify the activity name in the process and establish a connection to HDFS.

The following table lists the configurations in the **General** tab of the ListFileStatus activity:

Field	Module Property?	Description
Name	No	The name to be displayed as the label for the activity in the process.
HDFSConnection	Yes	The HDFS Connection shared resource that is used to create a connection between the plug-in and HDFS. Click to select an HDFS Connection shared resource. If no matching HDFS Connection shared resources are found, click **Create Shared Resource** to create one. For more details, see Creating an HDFS Connection.

Description

In the **Description** tab, you can enter a short description for the ListFileStatus activity.

Input

In the **Input** tab, you can specify the path of the file or directory that you want to list status for.

The following table lists the input elements in the **Input** tab of the ListFileStatus activity:

Input Item	Data Type	Description
`path`	String	The path of a file or a directory. If you specify an empty folder in this field, the **Output** tab is empty.
`timeout`	Long	The amount of time, in milliseconds, to wait for this activity to complete. By default, this activity does not time out if you do not specify a value.

Output

In the **Output** tab, you can view detailed file status.

The following table lists the output elements in the **Output** tab of the ListFileStatus activity:

Output Item	Data Type	Description
fileinfo	Complex	The file information. This element contains the elements from **accessTime** to **pathSuffix** that are listed in this table.
accessTime	Long	The access time of the file in milliseconds.
blockSize	Long	The block size of the file.
length	Long	The number of bytes in a file.
modification Time	Long	The modification time of the file in milliseconds.
replication	Long	The number of replications of the file.
owner	String	The user name of the file owner.
type	String	The type of the path object, FILE or DIRECTORY.
group	String	The group that is associated with the file.
permission	String	The permission of the file.
pathSuffix	String	The path suffix of the file.

Fault

In the **Fault** tab, you can view the error code and error message of the ListFileStatus activity. See Error Codes for more detailed explanation of errors.

The following table lists the error schema elements in the **Fault** tab of the ListFileStatus activity:

Error Schema Element	Data Type	Description
msg	String	The error message description that is returned by the plug-in.
msgCode	String	The error code that is returned by the plug-in.
exception	String	The exception occurs when the plug-in has internal errors.
message	String	The error message that is returned by the server.
javaClassName	String	The name of the Java class where an error occurs.

Read

You can use the Read activity to read data from a file in HDFS and place its content into the **Output** tab of the activity.

Because of the limitations of TIBCO ActiveMatrix BusinessWorks, this activity cannot read more than 2 GB data at one time. You can use the group to iteratively read the data in a file of 2 GB or more than 2 GB.

General

In the **General** tab, you can specify the activity name in the process, establish a connection to HDFS, and specify which format you want the file to be read in.

The following table lists the configurations in the **General** tab of the Read activity:

Field	Module Property?	Description
Name	No	The name to be displayed as the label for the activity in the process.
HDFSConnection	Yes	The HDFS Connection shared resource that is used to create a connection between the plug-in and HDFS. Click to select an HDFS Connection shared resource. If no matching HDFS Connection shared resources are found, click **Create Shared Resource** to create one. For more details, see Creating an HDFS Connection.
ReadAs	No	The format that you want the file to be read in. Select a format from the list: • **text** • **binary**

Description

In the **Description** tab, you can enter a short description for the Read activity.

Input

In the **Input** tab, you can specify how you want the file to be read.

The following table lists the input elements in the **Input** tab of the Read activity:

Input Item	Data Type	Description
`fileName`	String	The path of the file to be read.
`offset`	Long	The starting byte position to be read. The value in this field must be 0 or greater.
`length`	Long	The number of bytes to be read.

Input Item	Data Type	Description
bufferSize	Integer	The size of the buffer used in transferring data. The value in this field must be greater than 0.
timeout	Long	The amount of time, in milliseconds, to wait for this activity to complete. By default, this activity does not time out if you do not specify a value.

Output

In the **Output** tab, you can view the file content. The output elements of the Read activity vary depending on the file format that you select in the **General** tab.

The following table lists the output elements in the **Output** tab of the Read activity:

Output Item	Data Type	Description
fileContent	Complex	The file content. This element contains the **textContent** or **binaryContent** element.
text **Content**	String	The file content in text format. This item is displayed only when you select **text** in the **ReadAs** field in the **General** tab.
binary **Content**	Base64 Binary	The file content in binary format. This item is displayed only when you select **binary** in the **ReadAs** field in the **General** tab.
end	Boolean	You can view whether the file has been read to the end.

Fault

In the **Fault** tab, you can view the error code and error message of the Read activity. See Error Codes for more detailed explanation of errors.

The following table lists the error schema elements in the **Fault** tab of the Read activity:

Error Schema Element	Data Type	Description
msg	String	The error message description that is returned by the plug-in.
msgCode	String	The error code that is returned by the plug-in.
exception	String	The exception when the plug-in has internal errors.
message	String	The error message that is returned by the server.

Error Schema Element	Data Type	Description
`javaClassName`	String	The name of the Java class where an error occurs.

Write

You can use the Write activity to write data to a specified file in HDFS.

To append data to the same file or overwrite data in the same file in HDFS while running multiple processes simultaneously, you have to turn on the single thread by adding the following properties to the VM arguments in TIBCO Business Studio:

```
-Dcom.tibco.plugin.bigdata.write.singlethread=true
```

```
-Dbw.application.job.flowlimit.project_name.application=1
```

To access the VM arguments in TIBCO Business Studio, click **Run** > **Run Configurations**, expand **BusinessWorks Application** > **BWApplication** in the left panel of the Run Configurations dialog, and then click the **Argument** tab in the right panel of the dialog.

Also, for deployment of the Write activity, you have to add the following properties to the `Config.ini` file:

```
com.tibco.plugin.bigdata.write.singlethread=true
```

```
bw.application.job.flowlimit.project_name.application=1
```

The `Config.ini` file is located in the *`TIBCO_HOME`*`/bw/`*`version_number`*`/domains/`*`domain_name`*`/appnodes/`*`appspace_name`*`/`*`appnode_name`* directory.

General

In the **General** tab, you can specify the activity name in the process, establish a connection to HDFS, and decide wether to append data to an existing file or overwrite data in an existing file.

The following table lists the configurations in the **General** tab of the Write activity:

Field	Module Property?	Description
Name	No	The name to be displayed as the label for the activity in the process.
HDFSConnection	Yes	The HDFS Connection shared resource that is used to create a connection between the plug-in and HDFS. Click  to select an HDFS Connection shared resource. If no matching HDFS Connection shared resources are found, click **Create Shared Resource** to create one. For more details, see Creating an HDFS Connection.
Append	No	If you want to append the data to an existing file, you can select this check box.

Field	Module Property?	Description
WriteType	No	The format of the data to be written. Select a format from the list: • **text** • **binary** • **file** • **StreamObject**
Overwrite	Yes	If you want to overwrite the existing data in the specified file, you can select this check box. This check box is displayed only when you clear the **Append** check box.

Description

In the **Description** tab, you can enter a short description for the Write activity.

Input

In the **Input** tab, you can configure the append or overwrite operation that you select in the **General** tab. The input elements of the Write activity vary depending on the data format that you select in the **General** tab.

The following table lists all the possible input elements in the **Input** tab of the Write activity:

Input Item	Data Type	Description
`fileName`	String	The path of the file to be written in.
`fileContent`	String	The content to be written in the file. This element is displayed only when you select **text** from the **WriteType** list in the **General** tab.
`binaryData`	Base64 Binary	The binary data to be written in the file. This element is displayed only when you select **binary** from the **WriteType** list in the **General** tab.
`sourceFile` `Path`	String	The path of the file to be written. This element is displayed only when you select **file** from the **WriteType** list in the **General** tab.
`inputStream` `Object`	Object	The streaming object to be written. This element is displayed only when you select **StreamObject** from the **WriteType** list in the **General** tab.
`overwrite`	Boolean	You can specify whether you want to overwrite the existing data in the source file, `1` (true) or `0` (false).

Input Item	Data Type	Description
blockSize	Long	The block size of the file. The value in this field must be greater than 0.
replication	Short	The number of replications of the file. The value in this field must be greater than 0.
permission	Integer	The permission of the file. The value in this field must be in the range 0 - 777.
buffersize	Integer	The size of the buffer that is used in transferring data. The value in this field must be greater than 0.
timeout	Long	The amount of time, in milliseconds, to wait for this activity to complete. By default, this activity does not time out if you do not specify a value.

Output

In the **Output** tab, you can view the operation status of the Write activity.

The following table lists the output elements in the **Output** tab of the Write activity:

Output Item	Data Type	Description
status	Integer	A standard HTTP status code that indicates whether the execution is successful.
msg	String	The execution message.

Fault

In the **Fault** tab, you can view the error code and error message of the Write activity. See Error Codes for more detailed explanation of errors.

The following table lists the error schema elements in the **Fault** tab of the Write activity:

Error Schema Element	Data Type	Description
msg	String	The error message description that is returned by the plug-in.
msgCode	String	The error code that is returned by the plug-in.
exception	String	The exception occurs when the plug-in has internal errors.
message	String	The error message that is returned by the server.
javaClassName	String	The name of the Java class where an error occurs.

Hadoop Palette

With the Hadoop palette, you can use the benefits of Hive, MapReduce, and Pig based on Hadoop.

You can use the following activities in the Hadoop palette:

- Hive
- MapReduce
- Pig
- WaitForJobCompletion

Hive

You can use the Hive activity to facilitate querying and managing large datasets located in a distributed storage.

- If you run this activity on the Red Hat platform (except version 7.0), you have to upgrade XML User Interface Language (XUL) Runner to version 1.8 or later. After the upgrading, reinstall Mozilla Firefox. To run this activity on Red Hat Enterprise Linux 7.0, see the "Known Issues" part in *TIBCO ActiveMatrix BusinessWorks Plug-in for Big Data Release Notes* for more details.
- The Hive activity does not support uploading Hive data from local clusters.

The latest Hive, 0.12 or later, is supported by default. If you want to use the Hive activity based on previous Hive versions (earlier than Hive 0.12), you have to add the following property to the VM arguments in TIBCO Business Studio:

```
-Dcom.tibco.plugin.bigdata.oldhive.active=true
```

To access the VM arguments in TIBCO Business Studio, click **Run** > **Run Configurations**, expand **BusinessWorks Application** > **BWApplication** in the left panel of the Run Configurations dialog, and then click the **Argument** tab in the right panel of the dialog.

Also, for deployment of the WaitForJobCompletion activity, you have to add the following property to the `Config.ini` file:

```
com.tibco.plugin.bigdata.oldhive.active=true
```

The `Config.ini` file is located in the *`TIBCO_HOME`*`/bw/`*`version_number`*`/domains/`*`domain_name`*`/appnodes/`*`appspace_name`*`/`*`appnode_name`* directory.

General

In the **General** tab, you can specify the activity name in the process, establish a connection to HCatalog, and specify Hive scripts and other Hive options to query data.

The following table lists the configurations in the **General** tab of the Hive activity:

Field	Module Property?	Description
Name	No	The name to be displayed as the label for the activity in the process.

Field	Module Property?	Description
HCatalog Connection	Yes	The HCatalog Connection shared resource that is used to create a connection between the plug-in and HCatalog. Click  to select an HCatalog Connection shared resource. If no matching HCatalog Connection shared resources are found, click **Create Shared Resource** to create one. For more details, see Creating an HCatalog Connection.
IsFileBase	No	If the Hive scripts are from a file, you can select this check box
Hive Script File	Yes	The path of the file that contains the Hive scripts. This field is displayed only when you select the **IsFileBase** check box.
Hive Editor	No	If the Hive scripts are not from a file, you can enter the Hive scripts directly in the **Hive Editor** field. The keywords of the scripts are highlighted automatically. This field is displayed only when you clear the **IsFileBase** check box.
Define	No	In this field, you can define the Hive configuration variables. A variable is associated with a name and a value.
Status Directory	Yes	The directory where the status of the Hive job is located.
WaitForResult	Yes	If you want the process to wait for the Hive operation to complete, you can select this check box .  When you select this check box, the Hive activity does not support querying more than 2 GB result data at one time because of the limitations of TIBCO ActiveMatrix BusinessWorks.

Description

In the **Description** tab, you can enter a short description for the Hive activity.

Input

The values that you specify in the **Input** tab override the ones that you specify in the corresponding fields in the **General** tab.

The following table lists all the possible input elements in the **Input** tab of the Hive activity:

Input Item	Data Type	Description
`HiveFile`	String	The path of the HDFS file that contains commands. This element is displayed only when you select the **IsFileBase** check box in the **General** tab.

Input Item	Data Type	Description
`HiveScript`	String	The Hive scripts directly. This element is displayed only when you clear the **IsFileBase** check box in the **General** tab.
`Defines`	String	You can define the Hive configuration variables. Each variable is associated with a name and a value.
`Status Directory`	String	The directory where the status of the Hive job is located.
`timeout`	Long	The amount of time, in milliseconds, to wait for this activity to complete. By default, this activity does not time out if you do not specify a value.

Output

In the **Output** tab, you can view the job ID of the Hive operation or the result of the job depending on whether you select the **WaitForResult** check box in the **General** tab.

The following table lists the output elements in the **Output** tab of the Hive activity:

Output Item	Data Type	Description
`jobId`	String	The job ID of the Hive operation. You can use the WaitForJobCompletion activity to wait for the job to complete. The `exitValue` output element in the **Output** tab of the WaitForJobCompletion activity The exit value of Hive SQL execution. This element is displayed only when you clear **WaitForResult** check box in the **General** tab.
`content`	String	The result of the job. This element is displayed only when you select the **WaitForResult** check box in the **General** tab.

Fault

In the **Fault** tab, you can view the error code and error message of the Hive activity. See Error Codes for more detailed explanation of errors.

The following table lists the error schema elements in the **Fault** tab of the Hive activity:

Error Schema Element	Data Type	Description
`msg`	String	The error message description that is returned by the plug-in.
`msgCode`	String	The error code that is returned by the plug-in.

MapReduce

You can use the MapReduce activity to create and queue a standard or streaming MapReduce job.

General

In the **General** tab, you can specify the activity name in the process, establish a connection to HCatalog, and create and queue a standard or streaming MapReduce job.

The following table lists the configurations in the **General** tab of the MapReduce activity:

Field	Module Property?	Description
Name	No	The name to be displayed as the label for the activity in the process.
HCatalog Connection	Yes	The HCatalog Connection shared resource that is used to create a connection between the plug-in and HCatalog. Click  to select an HCatalog Connection shared resource. If no matching HCatalog Connection shared resources are found, click **Create Shared Resource** to create one. For more details, see Creating an HCatalog Connection.
Streaming	No	If you want to create and run streaming MapReduce jobs, you can select this check box.
Input	Yes	The path of the input data in Hadoop. This field is displayed only when you select the **Streaming** check box.
Output	Yes	The path of the output data. This field is displayed only when you select the **Streaming** check box.
Mapper	Yes	The path of the mapper program in Hadoop. This field is displayed only when you select the **Streaming** check box.
Reducer	Yes	The path of the reducer program in Hadoop. This field is displayed only when you select the **Streaming** check box.
Jar Name	Yes	The name of the .jar file for the MapReduce activity to use. This field is displayed only when you clear the **Streaming** check box.
Main Class	Yes	The name of the class for the MapReduce activity to use. This field is displayed only when you clear the **Streaming** check box.

Field	Module Property?	Description
Lib Jars	Yes	The comma-separated .jar file to be included in the classpath. This field is displayed only when you clear the **Streaming** check box.
Files	Yes	The comma-separated .jar files to be copied to the MapReduce cluster. This field is displayed only when you clear the **Streaming** check box.
Status Directory	Yes	The directory where the status of MapReduce jobs is stored.
Arguments	No	The program arguments. • If you select the **Streaming** check box, specify a list of program arguments that contain space-separated strings to pass to the Hadoop streaming utility. For example: `- files /user/hdfs/file - D mapred.reduce.task=0 - input format org.apache.hadoop.mapred.lib.NLineInputFormat - cmdenv info=wc-reducer` • If you clear the **Streaming** check box, specify the Java main class arguments.
Define	No	In this field, you can define the Hadoop configuration variables. A variable is associated with a name and a value. This field is displayed only when you clear the **Streaming** check box.

Description

In the **Description** tab, you can enter a short description for the MapReduce activity.

Input

The values that you specify in the **Input** tab override the ones that you specify in the corresponding fields in the **General** tab.

The following table lists the input elements in the **Input** tab of the MapReduce activity:

Input Item	Data Type	Description
`Input`	String	The path of the input data in Hadoop. This element is displayed only when you select the **Streaming** check box in the **General** tab.

Input Item	Data Type	Description
Output	String	The path of the output data. This element is displayed only when you select the **Streaming** check box in the **General** tab.
Mapper	String	The path of the mapper program in Hadoop. This element is displayed only when you select the **Streaming** check box in the **General** tab.
Reducer	String	The path of the reducer program in Hadoop. This element is displayed only when you select the **Streaming** check box in the **General** tab.
JarName	String	The name of the .jar file for the MapReduce activity to use. This element is displayed only when you clear the **Streaming** check box in the **General** tab.
ClassName	String	The name of the class for the MapReduce activity to use. This element is displayed only when you clear the **Streaming** check box in the **General** tab.
Libjars	String	The comma-separated .jar file to be included in the classpath. This element is displayed only when you clear the **Streaming** check box in the **General** tab.
Files	String	The comma-separated .jar files to be copied to the MapReduce cluster. This element is displayed only when you clear the **Streaming** check box in the **General** tab.
Status Directory	String	The directory where the status of MapReduce jobs is stored.
Arguments	String	The program arguments.
Defines	String	You can define the Hadoop configuration variables. A variable is associated with a name and a value. This field is displayed only when you clear the **Streaming** check box in the **General** tab.
timeout	Long	The amount of time, in milliseconds, to wait for this activity to complete. By default, this activity does not time out if you do not specify a value.

Output

In the **Output** tab, you can view the job ID of the MapReduce operation.

The following table lists the output element in the **Output** tab of the MapReduce activity:

Output Item	Data Type	Description
`jobId`	String	The job ID of the MapReduce operation.  You can use the WaitForJobCompletion activity to wait for the job to complete. The `exitValue` output element in the **Output** tab of the WaitForJobCompletion activity displays the exit value of MapReduce execution.

Fault

In the **Fault** tab, you can view the error code and error message of the MapReduce activity. See Error Codes for more detailed explanation of errors.

The following table lists the error schema elements in the **Fault** tab of the MapReduce activity:

Error Schema Element	Data Type	Description
`msg`	String	The error message description that is returned by the plug-in.
`msgCode`	String	The error code that is returned by the plug-in.

Pig

You can use the Pig activity to create and queue a Pig job.

If you run this activity on the Red Hat platform (except version 7.0), you have to upgrade XML User Interface Language (XUL) Runner to version 1.8 or later. After the upgrading, reinstall Mozilla Firefox. To run this activity on Red Hat Enterprise Linux 7.0, see the "Know Issues" part in *TIBCO ActiveMatrix BusinessWorks Plug-in for Big Data Release Notes* for more details.

General

In the **General** tab, you can specify the activity name in the process, establish a connection to HCatalog, and create and queue a Pig job.

The following table lists the configurations in the **General** tab of the Pig activity:

Field	Module Property?	Description
Name	No	The name to be displayed as the label for the activity in the process.
HCatalog Connection	Yes	The HCatalog Connection shared resource that is used to create a connection between the plug-in and HCatalog. Click  to select an HCatalog Connection shared resource. If no matching HCatalog Connection shared resources are found, click **Create Shared Resource** to create one. For more details, see Creating an HCatalog Connection.

Field	Module Property?	Description
IsFileBase	No	If the Pig scripts are from a file, you can select this check box.
Pig File	Yes	The path of the file that contains the Pig scripts. This field is displayed only when you select the **IsFileBase** check box.
Pig Editor	No	If the Pig scripts are not from a file, you can enter the Pig scripts directly in the **Pig Editor** field. The keywords of the scripts are highlighted automatically. This field is displayed only when you clear the **IsFileBase** check box.
Arguments	No	The Pig arguments that contain space-separated string.
Status Directory	Yes	The directory where the status of the Pig job is located.
Files	Yes	The comma-separated files to be copied to the MapReduce cluster.

Description

In the **Description** tab, you can enter a short description for the Pig activity.

UDF

In the **UDF** tab, you can use user-defined functions (UDFs) to specify custom processing.

The following table lists the configurations in the **UDF** tab of the Pig activity:

Field	Module Property?	Description
UDF Directory	Yes	The directory where the UDFs are located.
Available UDF Files	No	The available UDF files under the specified UDF directory. You have to specify the UDF file that you want to use. Click **Sync** to retrieve the available UDF files, select the UDF file that you want to use, and then click **Register**. The UDF file is displayed in the **Pig Editor** field in the **General** tab.
Upload UDF File	Yes	You can click  to select the UDF file that you want to upload , and then click **Upload** to upload the file to the specified directory.

Input

The values that you specify in the **Input** tab override the ones that you specify in the corresponding fields in the **General** tab.

The following table lists the input elements in the **Input** tab of the Pig activity:

Input Item	Data Type	Description
`PigScript`	String	The Pig scripts. This element is displayed only when you clear the **IsFileBase** check box in the **General** tab.
`PigFile`	String	The comma-separated files to be copied to the MapReduce cluster. This element is displayed only when you select the **IsFileBase** check box in the **General** tab.
`Arguments`	String	The Pig arguments.
`Status Directory`	String	The directory where the status of the Pig job is located.
`Files`	String	The comma-separated files to be copied to the MapReduce cluster.
`timeout`	Long	The amount of time, in milliseconds, to wait for this activity to complete. By default, this activity does not time out if you do not specify a value.

Output

In the **Output** tab, you can view the job ID of the Pig operation.

The following table lists the output element in the **Output** tab of the Pig activity:

Output Item	Data Type	Description
`jobId`	String	The job ID of the Pig operation.  You can use the WaitForJobCompletion activity to wait for the job to complete. The `exitValue` output element in the **Output** tab of the WaitForJobCompletion activity displays the exit value of Pig scripts execution.

Fault

In the **Fault** tab, you can view the error code and error message of the Pig activity. See Error Codes for more detailed explanation of errors.

The following table lists the error schema elements in the **Fault** tab of the Pig activity:

Error Schema Element	Data Type	Description
`msg`	String	The error message description that is returned by the plug-in.
`msgCode`	String	The error code that is returned by the plug-in.

WaitForJobCompletion

You can use the WaitForJobCompletion activity to wait for a specified job to complete until it reaches a specified time.

General

In the **General** tab, you can specify the activity name in the process, establish a connection to HCatalog, specify the maximum time for the activity to wait for a job to complete, and specify the interval to check the job execution.

The following table lists the configurations in the **General** tab of the WaitForJobCompletion activity:

Field	Module Property?	Description
Name	No	The name to be displayed as the label for the activity in the process.
HCatalog Connection	Yes	The HCatalog Connection shared resource that is used to create a connection between the plug-in and HCatalog. Click  to select an HCatalog Connection shared resource. If no matching HCatalog Connection shared resources are found, click **Create Shared Resource** to create one. For more details, see Creating an HCatalog Connection.
Timeout	Yes	The amount of time, in milliseconds, to wait for a job to complete. The default value is `1200000`.
Interval	Yes	The time interval, in milliseconds, to check the job execution. The default value is `5000`.

Description

In the **Description** tab, you can enter a short description for the WaitForJobCompletion activity.

Input

The values that you specify in the **Input** tab override the ones that you specify in the corresponding fields in the **General** tab.

The following table lists the input elements in the **Input** tab of the WaitForJobCompletion activity:

Input Item	Data Type	Description
`jobId`	String	The ID of the job to be waited for.
`timeout`	Long	The amount of time, in milliseconds, to wait for a job to complete.

Input Item	Data Type	Description
interval	Long	The time interval, in milliseconds, to check the job execution.

Output

In the **Output** tab, you can view a job details including its status, profile, ID, and so on.

The following table lists the output elements in the **Output** tab of the WaitForJobCompletion activity:

Output Item	Data Type	Description
Job	Complex	The job. This element contains the elements from **status** to **userargs** that are listed in this table.
status	Complex	The job status. This element contains child elements, see Child Elements of Status for more information.
Profile	Complex	The job profile. This element contains child elements, see Child Elements of Profile for more information.
id	String	The job ID.
parentId	String	The parent ID of the job.
percent Complete	String	The completion percentage of the job.
exitValue	String	The exit value of the job. If a value of 0 is returned, it indicates that the job is executed successfully. Otherwise, the job is completed with errors.  You can get error messages of the job from the `StatusDirectory/stderr` directory. You can specify the `StatusDirectory` value in the **Status Directory** field in the **General** tab of the activity that executes the job. The ErrorHandler process in the sample project that is shipped with the installer shows how to get error messages of a job. See Working with the Sample Project for more information.
user	String	The user name of the job owner.
callback	String	The callback URL.
completed	String	The completion status of the job. For example, **done**.

Output Item	Data Type	Description
userargs	Complex	The user arguments. This element contains child elements, see Child Elements of Userargs for more information.

Fault

In the **Fault** tab, you can view the error code and error message of the WaitForJobCompletion activity. See Error Codes for more detailed explanation of errors.

The following table lists the error schema elements in the **Fault** tab of the WaitForJobCompletion activity:

Error Schema Element	Data Type	Description
Hadoop Exception	Complex	The exception thrown by the plug-in. This element contains the **msg** and **msgCode** elements.
msg	String	The error message description that is returned by the plug-in.
msgCode	String	The error code that is returned by the plug-in.
ActivityTimeout Exception	Complex	The exception thrown by TIBCO ActiveMatrix BusinessWorks. This element contains the **msg** and **msgCode** elements.
msg	String	The error message description that is returned by TIBCO ActiveMatrix BusinessWorks.
msgCode	String	The error code that is returned by TIBCO ActiveMatrix BusinessWorks.

Child Elements of Status

The table lists the child elements of the **status** element in the **Output** tab of the WaitForJobCompletion activity.

Output Item	Data Type	Description
startTime	String	The start time of the job.
username	String	The user name of the job owner.
jobID	String	The job ID.
jobACLs	String	The ACLs for the job.
scheduling Info	String	The scheduling information associated with the job.

Output Item	Data Type	Description
failureInfo	String	The reason for the failure of the job.
jobId	String	The job ID.
jobPriority	String	The job priority.
runState	String	The current running state of the job.
State	String	The job state.
jobComplete	String	The completion status of the job.
priority	String	The job priority.
jobName	String	The job name.
map Progress	String	The map progress.
reduce Progress	String	The reduce progress.
cleanup Progress	String	The cleanup progress.
setup Progress	String	The setup progress.
queue	String	The queue name of the job.
jobFile	String	The configuration file of the job.
finishTime	String	The finish time of the job.
historyFile	String	The job history file URL for a completed job.
trackingUrl	String	The tracking URL for details of the job.
numUsed Slots	String	The number of used slots.
numReserved Slots	String	The number of reserved slots.
usedMem	String	The used memory.
neededMem	String	The needed memory.
retired	String	You can view whether the job is retired.

Output Item	Data Type	Description
uber	String	You can view whether the job is running in uber mode.

Child Elements of Profile

The table lists the child elements of the **profile** element in the **Output** tab of the WaitForJobCompletion activity.

Output Item	Data Type	Description
url	String	The job URL.
jobID	String	The job ID.
user	String	The user name of the job owner.
queueName	String	The queue name of the job.
jobFile	String	The name of the job file.
jobName	String	The job name.
jobId	String	The job ID.

Child Elements of Userargs

The table lists the child elements of the **userargs** element in the **Output** tab of the WaitForJobCompletion activity.

Output Item	Data Type	Description
statusdir	String	The directory where the status of the job is located.
files	String	The comma-separated files to be copied to the cluster.
define	String	The define configuration of the job.
enablelog	String	You can view whether the **enablelog** parameter is set to `true`.
execute	String	The entire short command to run.
username	String	The user name of the job owner.
file	String	The HDFS file name that is used in a job.
callback	String	The callback URL.

Working with the Sample Project

The plug-in packages a sample project with the installer. The sample project shows how TIBCO ActiveMatrix BusinessWorks Plug-in for Big Data works.

After installing the plug-in, you can find the `samples.zip` file in the *`TIBCO_HOME`*`/bw/palettes/bigdata/`*`version_number`*`/Samples` directory.

This sample project contains one process and a subprocess.

- DemoWorkflow.bwp

  The DemoWorkflow.bwp process is a main process. It shows how to operate HDFS files, transfer files between HDFS and a local directory, create a Hive table, and query data.

  The DemoWorkflow.bwp process contains the following activities:

Activity	Description
Timer	Starts the process when the specified time interval expires.
RemoveExistingFiles	Deletes the `customers.csv` file from the `user/hdfs/bwdemo` directory in HDFS.
CopyDataToHDFS	Copies the `customers.csv` file from the *`TIBCO_HOME`*`/bw/palettes/bigdata/`*`version_number`*`/Samples/samples.zip/SampleData` directory to the `user/hdfs/bwdemo` directory in HDFS.
CreateHiveTableReferToData	Creates an external Hive table named `customers` in HDFS.
CheckError	Performs the subprocess. If no error occurs when the ErrorHandler.bwp process is running, the process goes to the HiveQuery activity. If any error occurs, the process goes to the Generate-Error activity.
HiveQuery	Queries 100 records from the `customers.csv` Hive table created by the CreateHiveTableReferToData activity.
Generate-Error	Generates error messages from the **Output** tab of the ErrorHandler.bwp subprocess.
Catch	Catches error messages and displays in the **Output** tab when any error occurs in the DemoWorkflow.bwp process.
End	Ends the process.

- ErrorHandler.bwp

  The ErrorHandler.bwp is a subprocess of the DemoWorkflow.bwp process.

  The ErrorHandler.bwp subprocess contains the following activities:

Activity	Description
Start	Starts the process.
WaitForJobCompletion-Activity	Waits for the CreateHiveTableRefertToData activity to complete the job.
ReadResult	Reads the output of the job executed by the CreateHiveTableRefertToData activity, which is in the `user/hdfs/hive/createstatus/stdout` directory.
ReadError	Reads error messages of the job executed by the CreateHiveTableRefertToData activity, which is in the `user/hdfs/hive/createstatus/stderr` directory.
Reply	Sends a message in response to the ErrorHander.bwp process and the ErrorHander operation.
Reply1	Sends a message in response to the ErrorHander.bwp process and the ErrorHander operation.
Generate Error	Generates error messages from the **Output** tab of the ReadError activity.
Catch	Catches the error messages and displays in the **Output** tab when any error occurs in the ErrorHandler.bwp subprocess.
RenderXML	Renders the error messages in XML string.
End	Ends the process.

Importing the Sample Project

Before running the sample project, you must import the project to TIBCO Business Studio.

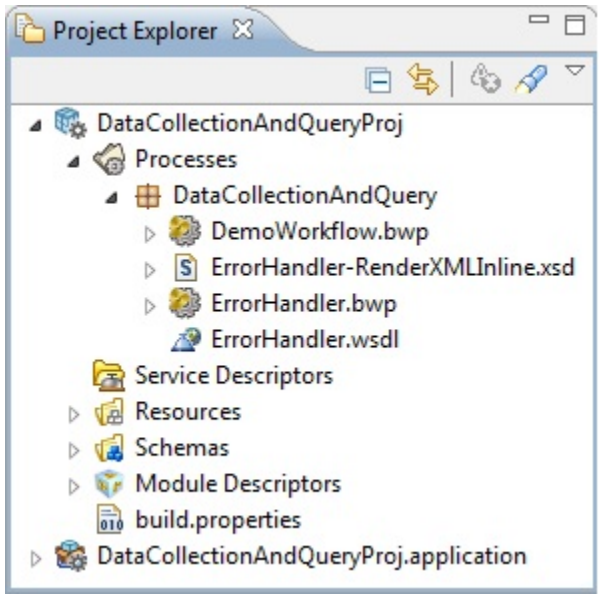
Procedure

1. Start TIBCO Business Studio using one of the following ways:
   - Microsoft Windows: click **Start** > **All Programs** > **TIBCO** > *TIBCO_HOME* > **TIBCO Business Studio** *version_number* > **Studio for Designers**.
   - Mac OS and Linux: run the TIBCO Business Studio executable file located in the *TIBCO_HOME*`/studio/`*version_number*`/eclipse` directory.
2. From the menu, click **File** > **Import**.
3. In the Import dialog, expand the **General** folder and select the **Existing Studio Projects into Workspace** item. Click **Next.**
4. Click **Browse** next to the **Select archive file** field to locate the sample. Click **Finish.**

   The sample project is located in the *TIBCO_HOME*`/bw/palettes/bigdata/`*version_number*`/Samples` directory.

Result

The sample project is imported to TIBCO Business Studio.



Running the Sample Project

This sample project contains one process that shows how to manage and query data.

Prerequisites

Ensure that you have imported the sample project to TIBCO Business Studio, as described in Importing the Sample Project.

Procedure

1. Configure the module properties that are used in the sample project:
   a) In the Project Explorer view, expand **DataCollectionAndQueryProj** > **Module Descriptors** and double-click **Module Properties**.
   b) Expand **Example** and configure the following properties that are used in the shared resources:
      - `HCATALOG_URL=`*`server_URL`*
      - `HDFS_URL=`*`server_URL`*
      - `Project_Path=`*`TIBCO_HOME`*`/bw/palettes/bigdata/`*`version_number`*`/Samples`
2. Expand **Resource** > **SharedResources**, edit the connections that are used in the sample project, and then click **Test Connection** to validate your connections.
   a) Double-click **HcatalogConn.sharedhadoopresource** to edit the HCatalog connection.
   b) Double-click **HDFSConn.sharedhdfsresource** to edit the HDFS connection.
3. Expand the **Module Descriptors** resource, and then double-click **Components**.
4. In the **Component Configurations** area, ensure that the DataCollectionAndQuery.DemoWorkflow component is selected.
5. On the toolbar, click the icon to save your changes.
6. From the menu, click **Run** > **Run Configurations** to run the selected process.
7. In the left panel of the Run Configurations dialog, expand **BusinessWorks Application** > **BWApplication**.
8. In the right panel, click the **Applications** tab, select the check box next to **DataCollectionAndQueryProj.application**.

9. Click **Run** to run the process.
10. Click the  icon to stop the process.

Result

If the process is run successfully, 100 records in the `customers.csv` file created by the CreateHiveTableRefertToData activity are returned in the **Output** tab of the HiveQuery activity.

Managing Logs

When an error occurs, you can check logs to trace and troubleshoot the plug-in exceptions.

By default, error logs are displayed in the Console view when you run a process. You can change the log level of the plug-in to trace different messages and export logs to a file. Different log levels correspond to different messages, as described in Log Levels.

Log Levels

Different log levels include different information.

The plug-in supports the following log levels:

Log Level	Description
Info	Indicates normal plug-in operations. No action is required. A tracing message tagged with Info indicates that a significant processing step is reached, and logged for tracking or auditing purposes. Only info messages preceding a tracking identifier are considered as significant steps.
Error	Indicates that an unrecoverable error occurred. Depending on the severity of the error, the plug-in might continue with the next operation or might stop.
Debug	Indicates a developer-defined tracing message.

Setting Up Log Levels

You can configure a different log level for the plug-in and plug-in activities to trace different messages.

By default, the plug-in uses the log level configured for TIBCO ActiveMatrix BusinessWorks. The default log level value of TIBCO ActiveMatrix BusinessWorks is `ERROR`.

Procedure

1. Navigate to the *`TIBCO_HOME`*`/bw/`*`version_number`*`/config/design/logback` directory and open the `logback.xml` file.
2. Add the following node in the **BusinessWorks Palette and Activity loggers** area to specify a log level for the plug-in:

```
<logger name="com.tibco.bw.palette.hadoop.runtime">
    <level value="DEBUG"/>
  </logger>
```

```
<logger name="com.tibco.bw.palette.webhdfs.runtime">
    <level value="DEBUG"/>
  </logger>
```

```
<logger name="com.tibco.bw.sharedresource.webhdfs.runtime">
    <level value="DEBUG"/>
  </logger>
```

```
<logger name="com.tibco.bw.sharedresource.hadoop.runtime">
    <level value="DEBUG"/>
  </logger>
```

The value of the **`level`** element can be `ERROR`, `INFO`, or `DEBUG`.

If you set the log level to `Debug`, the input and output for the plug-in activities are also displayed in the Console view. See Log Levels for more details regarding each log level.

3. Optional: Add the following node in the **BusinessWorks Palette and Activity loggers** area to specify a log level for an activity. The node varies according to the activity and the palette that it belongs to.

   - For activities in the HDFS palette:

     ```
     <logger name="com.tibco.bw.palette.webhdfs.runtime.ActivityNameActivity">
        <level value="DEBUG"/>
     </logger>
     ```

   - For activities in the Hadoop palette:

     ```
     <logger name="com.tibco.bw.palette.hadoop.runtime.ActivityNameActivity">
        <level value="DEBUG"/>
     </logger>
     ```

   For example, you can add the following node to set the log level of the HDFSOperation activity to `DEBUG`:

   ```
   <logger name="com.tibco.bw.palette.webhdfs.runtime.HDFSOperationActivity">
      <level value="DEBUG"/>
   </logger>
   ```

   Likewise, you can add the following node to set the log level of the Hive activity to `DEBUG`:

   ```
   <logger name="com.tibco.bw.palette.hadoop.runtime.HiveActivity">
      <level value="DEBUG"/>
   </logger>
   ```

   The activities that are not configured with specific log levels use the log level configured for the plug-in.

4. Save the file.

Enabling Kerberos-Related Logging

To enable Kerberos-related logging follow these steps:

1. In TIBCO Business Studio™, click **Run** > **Debug Configurations**.
2. Expand **BusinessWorks Application** and click **BWApplication**.
3. Click the **Arguments** tab.
4. Add the following in the **VM arguments** text box:

   ```
   -Dsun.security.krb5.debug=true
   ```

   ```
   -Dsun.security.jgss.debug=true
   ```

5. Click **Apply**.

Exporting Logs to a File

You can update the `logback.xml` file to export plug-in logs to a file.

Procedure

1. Navigate to the *`TIBCO_HOME`*`/bw/`*`version_number`*`/config/design/logback` directory and open the `logback.xml` file.

   After deploying an application in TIBCO Enterprise Administrator, navigate to the *`TIBCO_HOME`*`/bw/`*`version_number`*`/domains/`*`domain_name`*`/appnodes/`*`space_name`*`/`*`node_name`* directory to find the `logback.xml` file.

2. Add the following node to specify the file where the log is exported:

   ```
   <appender name="FILE" class="ch.qos.logback.core.FileAppender">
      <file>c:/bw6-bigdata.log</file>
         <encoder>
   ```

```
        <pattern>%d{HH:mm:ss.SSS} [%thread] %-5level %logger{36}-%msg%n</pattern>
      </encoder>
</appender>
```

The value of the **file** element is the absolute path of the file that stores the exported log.

3. Add the following node to the `root` node at the bottom of the `logback.xml` file:

```
<root level="DEBUG">
   <appender-ref ref="STDOUT" />
   <appender-ref ref="FILE" />
</root>
```

4. Save the file.

Installing JCE Policy Files

When connecting to a Kerberized HDFS or WebHCat server, if your server uses the AES-256 encryption, you must install Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy Files on your machine.

Procedure

1. Download the policy files from http://www.oracle.com/technetwork/java/embedded/embedded-se/downloads/jce-7-download-432124.html.
2. Extract the downloaded `UnlimitedJCEPolicyJDK7.zip` file, and then copy the files to the *`TIBCO_HOME`*`/tibcojre64/1.7.0/lib/security` directory.
3. Restart TIBCO Business Studio.

Error Codes

The following tables list error codes, detailed explanation of each error, and where applicable, ways to solve different errors.

HDFS Error Codes

The following table lists the error codes of the HDFS Connection shared resource and the error codes of the activities in the HDFS palette:

Error Code and Error Message	Role	Category	Description	Solution
TIBCO-BW-PALETTE-HDFS-2000000 `{0}`	debugRole	BW-Plug-in	Any debug message.	No action.
TIBCO-BW-PALETTE-HDFS-2000001 `Start execute: {0}`	debugRole	BW-Plug-in	Debug message for start activity execution.	No action.
TIBCO-BW-PALETTE-HDFS-2000002 `End execute: {0}`	debugRole	BW-Plug-in	Debug message for end activity execution.	No action.
TIBCO-BW-PALETTE-HDFS-2000003 `Customer thread start, thread ID: {0}`	debugRole	BW-Plug-in	Customer thread starts. The thread ID is {0}.	No action.
TIBCO-BW-PALETTE-HDFS-2000004 `Input data: {1}`	debugRole	BW-Plug-in	The activity input data is {1}.	No action.
TIBCO-BW-PALETTE-HDFS-2000005 `Output data: {1}`	debugRole	BW-Plug-in	The activity output data is {1}.	No action.
TIBCO-BW-PALETTE-HDFS-2000006 `{0} : {1}`	debugRole	BW-Plug-in	Debug message for two parameters.	No action.
TIBCO-BW-PALETTE-HDFS-2000007 `Cached tasks:{0}`	debugRole	BW-Plug-in	The number of cached tasks.	No action.

Error Code and Error Message	Role	Category	Description	Solution
TIBCO-BW-PALETTE-HDFS-5000001 `Please specify HDFS connection shared resource.`	errorRole	BW-Plug-in	An HDFS Connection shared resource is not specified.	Specify an HDFS Connection shared resource.
TIBCO-BW-PALETTE-HDFS-5000002 `Remote Error while run {0}`	errorRole	BW-Plug-in	A remote error occurs while running {0}.	No action.
TIBCO-BW-PALETTE-HDFS-5000003 `Error Occurred: {0}`	errorRole	BW-Plug-in	An error occurs. Detailed information is {0}.	No action.
TIBCO-BW-PALETTE-HDFS-5000004 `Invalid hdfs operation: {0}.`	errorRole	BW-Plug-in	The HDFS operation is invalid. Detailed information is {0}.	No action.
TIBCO-BW-PALETTE-HDFS-5000005 `Invalid HDFS URL, Please specify a correct one.`	errorRole	BW-Plug-in	The HDFS URL entered is invalid.	Specify a correct HDFS URL.
TIBCO-BW-PALETTE-HDFS-5000006 `{0} is not a valid buffer size, valid buffer size should > 0`	errorRole	BW-Plug-in	{0} is an invalid buffer size. The valid buffer size must be greater than 0.	Specify a value that is greater than 0.
TIBCO-BW-PALETTE-HDFS-5000007 `{0} is not a valid replication, valid replication should > 0`	errorRole	BW-Plug-in	{0} is an invalid replication. The valid replication must be greater than 0.	Specify a value that is greater than 0.
TIBCO-BW-PALETTE-HDFS-5000008 `{0} is not a valid offset, valid offset should >= 0`	errorRole	BW-Plug-in	{0} is an invalid offset. The valid offset must be greater than or equal to 0.	Specify a value that is greater than or equal to 0.

Error Code and Error Message	Role	Category	Description	Solution
TIBCO-BW-PALETTE-HDFS-5000009 `{0} is not a length, valid length should >= 0`	errorRole	BW-Plug-in	{0} is an invalid length. The valid length must be greater than or equal to 0.	Specify a value that is greater than or equal to 0.
TIBCO-BW-PALETTE-HDFS-5000010 `{0} is not a valid block size, valid block size should > 0`	errorRole	BW-Plug-in	{0} is an invalid block size. The valid block size must be greater than 0.	Specify a value that is greater than 0.
TIBCO-BW-PALETTE-HDFS-5000011 `{0} is not a valid permission, valid values should 0 - 777`	errorRole	BW-Plug-in	{0} not an invalid permission. The valid values must be in the range 0 - 777.	Specify a value in the range 0 - 777.
TIBCO-BW-PALETTE-HDFS-5000012 `Unknow result returned: {0}`	errorRole	BW-Plug-in	An unknown result is returned. Detailed information is {0} .	No action.
TIBCO-BW-PALETTE-HDFS-5000013 `IOException occurred while retrieving XML Output for activity [{0}].`	errorRole	BW-Plug-in	An IOException exception occurs while retrieving XML output for activity [{0}].	No action.

Hadoop Error Codes

The following table lists the error codes of the HCatalog Connection shared resource and the error codes of the activities in the Hadoop palette:

Error Code and Error Message	Role	Category	Description	Solution
TIBCO-BW-PALETTE-HADOOP-2000000 `{0}`	debugRole	BW-Plug-in	Any debug message.	No action.
TIBCO-BW-PALETTE-HADOOP-2000001 `Start execute: {0}`	debugRole	BW-Plug-in	Debug message for start activity execution.	No action.

Error Code and Error Message	Role	Category	Description	Solution
TIBCO-BW-PALETTE-HADOOP-2000002 `End execute: {0}`	debugRole	BW-Plug-in	Debug message for end activity execution.	No action.
TIBCO-BW-PALETTE-HADOOP-2000003 `Customer thread start, thread ID: {0}`	debugRole	BW-Plug-in	Customer thread starts. The thread ID is {0}.	No action.
TIBCO-BW-PALETTE-HADOOP-2000004 `Input data: {1}`	debuRole	BW-Plug-in	The activity input data is {1}.	No action.
TIBCO-BW-PALETTE-HADOOP-2000005 `Output data: {1}`	debugRole	BW-Plug-in	The activity output data is {1}.	No action.
TIBCO-BW-PALETTE-HADOOP-2000006 `Define name field empty so ignore it, Value is: {0}`	debugRole	BW-Plug-in	The Define name field is empty. The value is {0}.	No action.
TIBCO-BW-PALETTE-HADOOP-2000007 `{0} : {1}`	debugRole	BW-Plug-in	Debug message for two parameters.	
TIBCO-BW-PALETTE-HADOOP-2000008 `Cached tasks:{0}`	debugRole	BW-Plug-in	The number of cached tasks.	No action.
TIBCO-BW-PALETTE-HADOOP-3000000 `{0}`	infoRole	BW-Plug-in	Any info message.	No action.
TIBCO-BW-PALETTE-HADOOP-3000001 `{0} is done({1}ms)`	infoRole	BW-Plug-in	Activity {0} is done in {1} milliseconds.	No action.
TIBCO-BW-PALETTE-HADOOP-5000000 `Please specify HCatalog connection shared resource.`	errorRole	BW-Plug-in	An HCatalog Connection shared resource is not specified.	Specify an HCatalog Connection shared resource.

Error Code and Error Message	Role	Category	Description	Solution
TIBCO-BW-PALETTE-HADOOP-5000001 `Error Occurred: {0}`	errorRole	BW-Plug-in	An error occurs. Detailed information is {0}.	No action.
TIBCO-BW-PALETTE-HADOOP-5000002 `Invalid {0} URL, Please specify a correct one.`	errorRole	BW-Plug-in	The {0} URL entered is invalid.	Specify a correct URL.
TIBCO-BW-PALETTE-HADOOP-5000003 `Unknow result returned: {0}`	errorRole	BW-Plug-in	An unknown result is returned. Detailed information is {0} .	No action.
TIBCO-BW-PALETTE-HADOOP-5000004 `IOException occurred while retrieving XML Output for activity [{0}].`	errorRole	BW-Plug-in	An IOException exception occurs while retrieving XML output for activity [{0}].	No action.
TIBCO-BW-PALETTE-HADOOP-5000005 `Remote Error while run {0}`	erroRole	BW-Plug-in	A remote error occurs while running {0}.	No action.
TIBCO-BW-PALETTE-HADOOP-5000006 `Hive error occurred: {0}`	errorRole	BW-Plug-in	A Hive error occurs. Detailed information is {0}.	No action.
TIBCO-BW-PALETTE-HADOOP-5000007 `HDFS URL is empty, please specify one.`	errorRole	BW-Plug-in	The HDFS URL is empty.	Specify an HDFS URL.
TIBCO-BW-PALETTE-HADOOP-5000008 `Status directory is empty, please specify one.`	errorRole	BW-Plug-in	The status directory is empty.	Specify a status directory.

Error Code and Error Message	Role	Category	Description	Solution
TIBCO-BW-PALETTE-HADOOP-5000009 `Hive script cannot be null.`	errorRole	BW-Plug-in	The Hive scripts cannot be null.	Specify valid Hive scripts.
TIBCO-BW-PALETTE-HADOOP-5000010 `Hive file script cannot be null, please select one script file.`	errorRole	BW-Plug-in	The Hive file scripts cannot be null.	Specify the file that contains the Hive scripts.
TIBCO-BW-PALETTE-HADOOP-5000011 `Pig script cannot be null.`	errorRole	BW-Plug-in	The Pig scripts cannot be null.	Specify valid Pig scripts.
TIBCO-BW-PALETTE-HADOOP-5000012 `Pig file script cannot be null, please select one script file.`	errorRole	BW-Plug-in	The Pig file scripts cannot be null.	Specify the file that contains the Pig scripts.
TIBCO-BW-PALETTE-HADOOP-5000013 `Mapreduce jar name is required, please provide one.`	errorRole	BW-Plug-in	The jar name is required.	Specify the jar name in the **Jar Name** field of the MapReduce activity.
TIBCO-BW-PALETTE-HADOOP-5000014 `Mapreduce main class is required, please provide one.`	errorRole	BW-Plug-in	The main class is required.	Specify the main class in the **Main Class** field of the MapReduce activity.
TIBCO-BW-PALETTE-HADOOP-5000015 `Mapreduce streaming input is required, please provide one.`	errorRole	BW-Plug-in	The streaming input is required.	Specify the streaming input in the **Input** field of the MapReduce activity.
TIBCO-BW-PALETTE-HADOOP-5000016 `Mapreduce streaming mapper is required, please provide one.`	errorRole	BW-Plug-in	The streaming mapper is required.	Specify the mapper in the **Mapper** field of the MapReduce activity.

Error Code and Error Message	Role	Category	Description	Solution
TIBCO-BW-PALETTE-HADOOP-5000017 `Mapreduce streaming reducer is required, please provide one.`	errorRole	BW-Plug-in	The streaming reducer is required.	Specify the reducer in the **Reducer** field of the MapReduce activity.