TIBCO ActiveMatrix BusinessWorks™ Plug-in for Snowflake

User Guide

Version 6.3.0 | April 2024

Contents

Overview

TIBCO ActiveMatrix BusinessWorks™ is an easy-to-use integration product suite for enterprise, web, and mobile applications. TIBCO ActiveMatrix BusinessWorks™ uses the Eclipse graphical user interface (GUI) and TIBCO Business Studio™ for BusinessWorks™ for defining business processes and the process engine to execute the business processes.

TIBCO ActiveMatrix BusinessWorks™ Plug-in for Snowflake provides the interoperability between TIBCO ActiveMatrix BusinessWorks™ and Snowflake. With this plug-in, you can perform operations on the Snowflake data warehouse entities using JDBC. The Snowflake data warehouse uses a new SQL database engine with an architecture designed for the cloud.

ActiveMatrix BusinessWorks™ Plug-in for Snowflake extends TIBCO ActiveMatrix BusinessWorks™ and adds a Snowflake JDBC connection shared resource and a Snowflake Palette to TIBCO Business Studio™ for BusinessWorks™ (hereinafter referred to as "TIBCO Business Studio").

With this plug-in, you can perform operations using services supported by Snowflake. The plug-in supports the following features:

- Creating and Configuring the Snowflake JDBC Connection Resource: You can use this feature to connect to the Snowflake data warehouse.
- Snowflake Palette: The palette supports the following activities:
    - Snowflake Insert Activity: Use this activity to run the insert SQL statements on the Snowflake data warehouse.
    - Snowflake Query Activity: Use this activity to run the query SQL statements on the Snowflake data warehouse.
    - Snowflake Update Activity: Use this activity to run the update SQL statements on the Snowflake data warehouse.
    - Snowflake Delete Activity: Use this activity to run the delete SQL statements on the Snowflake data warehouse.
    - Snowflake Bulk Load Activity: Use this activity to load bulk data to Snowflake using STAGE.

Snowflake Ready Validation Program

The TIBCO ActiveMatrix BusinessWorks Plug-in for Snowflake is certified per the Snowflake Ready Validation Program for Data Integration.

Getting Started

A typical workflow for the plug-in (to achieve different goals) includes creating a process, testing it in the debugger, and deploying the application.

Most procedures in a typical workflow are performed in TIBCO Business Studio for BusinessWorks. See Overview if you are not familiar with it.

The following list has the sequence of topics that can help you get started with ActiveMatrix BusinessWorks™ Plug-in for Snowflake:

1. Creating a Project
2. Creating and Configuring the Snowflake JDBC Connection Resource
3. Configuring a Process
4. Testing a Process
5. Pushing an Application to Cloud

Note: When configuring the shared resources or activities, some fields might inherit the module properties. To modify the value of such fields, you must specify the values on the **Module Properties** tab of the Module Properties editor.

Overview of TIBCO Business Studio for BusinessWorks

TIBCO Business Studio™ for BusinessWorks™ is an Eclipse-based integration development environment that is used to design, develop, and test ActiveMatrix BusinessWorks applications. The studio provides a workbench in which you can create, manage, and navigate resources in your workspace. A *workspace* is the central location on your computer where all data files are stored.

The following table introduces the workbench UI elements:

UI Element	Description
Menu	Contains menu items such as File, Edit, Navigate, Search, Project, Run, Window, and Help.
Toolbar	Contains the following buttons for frequently used commands: • **New**  • **Save**  • **Enable/Disable Business Studio capabilities**  • **Create a new BusinessWorks Application Module**  • **Debug As**  • **Run As** 
Perspectives	Contains an initial set and layout of views that are required to perform a certain task. TIBCO Business Studio for BusinessWorks launches the Design perspective by default. Use the Design perspective when designing a process and the Debug perspective when testing and debugging a process. To change the perspective, select **Window > Open Perspective >** ***perspective_name*** from the main menu. Alternatively, click the **Open Perspective**  button from the top upper right of the workbench and select the perspective.
Views	Lists the resources and helps you navigate within the workbench. For example, the Project Explorer view displays the ActiveMatrix BusinessWorks applications, modules, and other resources in your workspace, and the Properties view displays the properties for the selected resource. To open a view, select **Window > Show View >** ***view_nameview_name*** from the main menu.
Editors	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 on a process (MortgageAppConsumer.bwp) in the Project Explorer view to open the process in the editor.

UI Element	Description
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.

Creating a Project

Projects are TIBCO ActiveMatrix BusinessWorks application modules that are created in TIBCO Business Studio for BusinessWorks. Begin by creating a project, and then 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.

Note: When importing an existing project to the current workspace of TIBCO Business Studio for BusinessWorks, if you click **Select root directory** in the Import Projects dialog, you must select the **Copy projects into workspace** checkbox.

Procedure

1. To start TIBCO Business Studio for BusinessWorks perform the steps for the platforms that are applicable to your plug-in.

   All the platforms might not be applicable to your plug-in. For the list of platforms supported by your plug-in, see the Readme file.

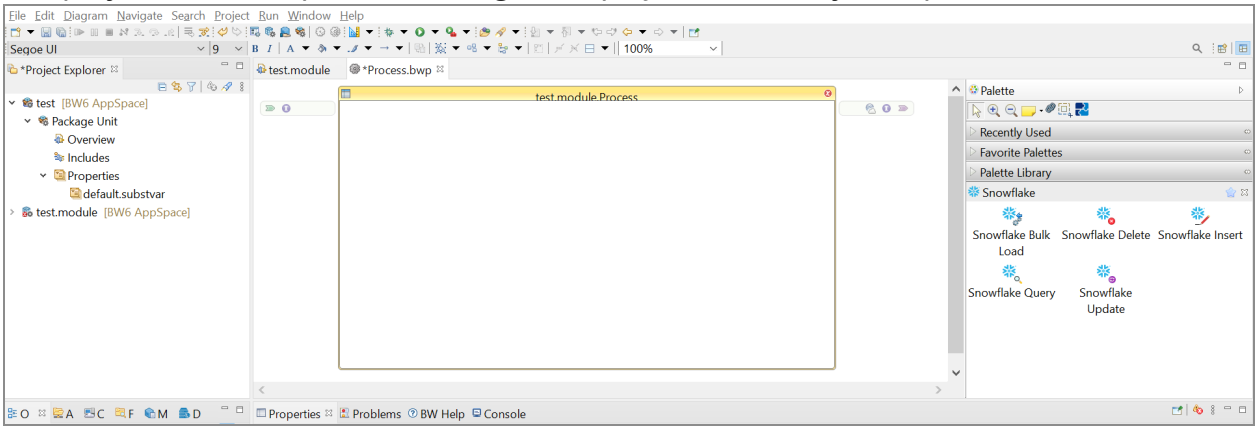
Platform	Step
Microsoft Windows	Click **Start > All Programs > TIBCO > *TIBCO_HOME* > TIBCO Business Studio *<version>* > Studio for Designers**
Linux or macOS	Run the TIBCO Business Studio for BusinessWorks executable file located in the *TIBCO_HOME*/studio/*<version>*/eclipse directory.

> **Note:** On macOS, to load the environment variables in the launchd.conf file correctly, ensure that the shell is bash when you start TIBCO Business Studio on a command line.

2. From the menu, click **File > New > BusinessWorks Resources** to open the BusinessWorks Resource wizard.
3. On the Select a wizard page, click **BusinessWorks Application Module** and click **Next** to open the Project page.
4. On the Project page, configure the project that you want to create:
    a. In the **Project name** field, enter a project name.
    b. If you do not want to use the default location (current workspace) for the project, clear the **Use default location** checkbox and click **Browse** to select a new location.
    c. In the Version field, retain the default version of the application module or enter a new version.
    d. To automatically create an empty process and an application when creating the project, ensure that the **Create empty process** and **Create Application** checkboxes are selected.
    e. **Optional:** To create a Java module, select the **Use Java configuration** checkbox.
    f. Click **Finish**.

Result

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



Creating and Configuring the Snowflake JDBC Connection Resource

After creating a project, you can add a Snowflake JDBC connection shared resource to establish a connection between the plug-in and the Snowflake data warehouse.

Procedure

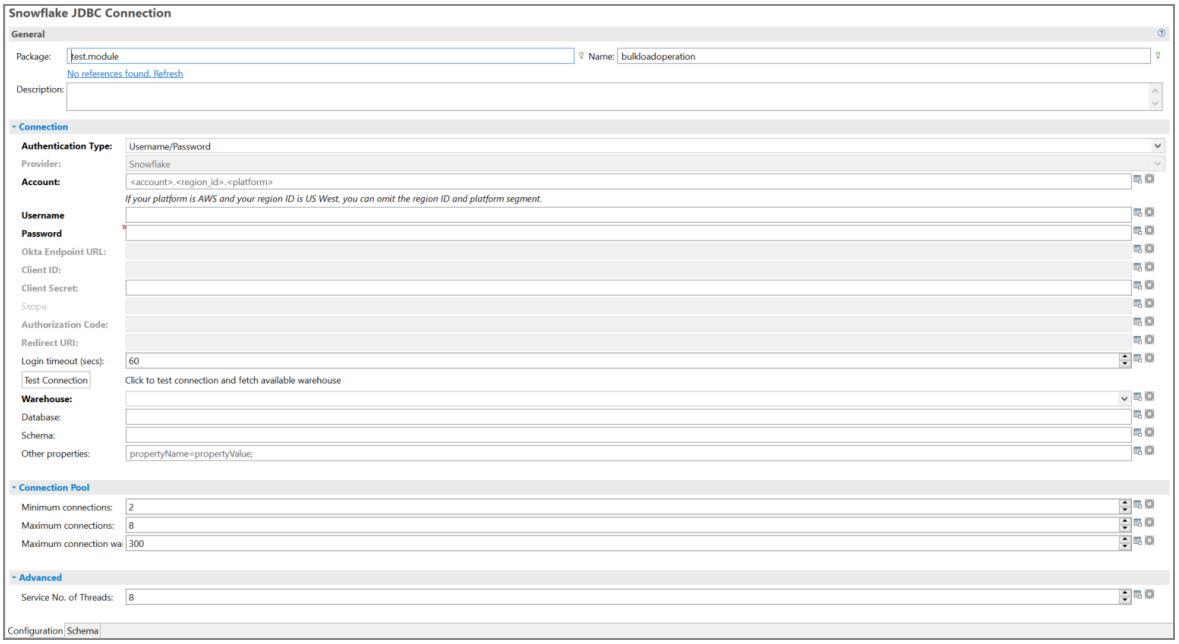
1. In the Project Explorer view, right-click **Resources** folder and select **New > Snowflake JDBC Connection**.

   Hint: The project must be expanded to display all the folders.

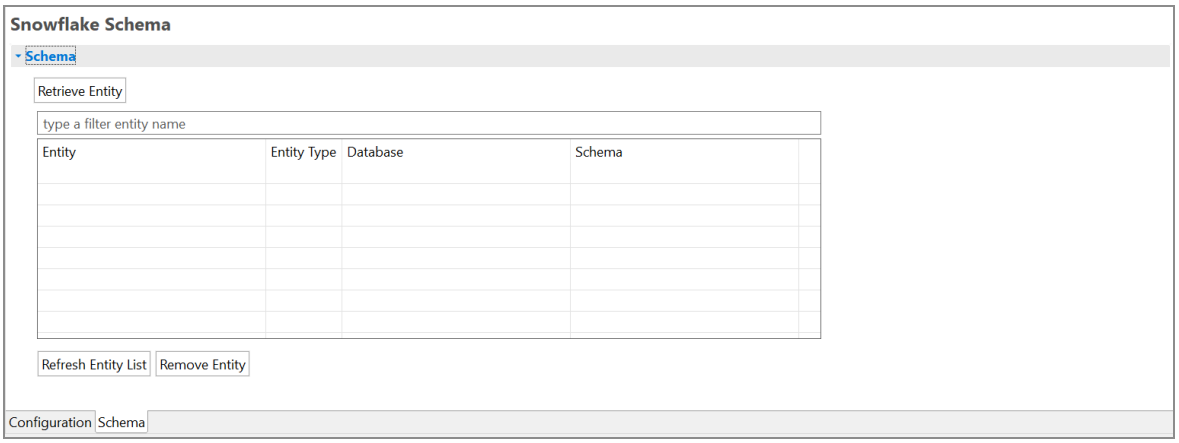
2. In the Snowflake JDBC Connection Resource dialog, fill the **Resource Name** field and click **Finish**. To create the shared resource in a resources folder of a different project or in a different package, select **Resource Folder** or **Package** accordingly .

   The Snowflake Connection shared resource is created. The shared resource consists of two tabs: **Configuration** and **Schema**.

   Snowflake JDBC Connection: **Configuration** Tab



   Snowflake JDBC Connection: **Schema** Tab



3. Configure the Snowflake JDBC Connection resource in the displayed editor, as described in Snowflake JDBC Connection Resource and download the required table metadata in the **Schema** tab .

Configuring a Process

A newly created project contains an empty process. Configure the process by adding activities, conditions, and services to complete a task.

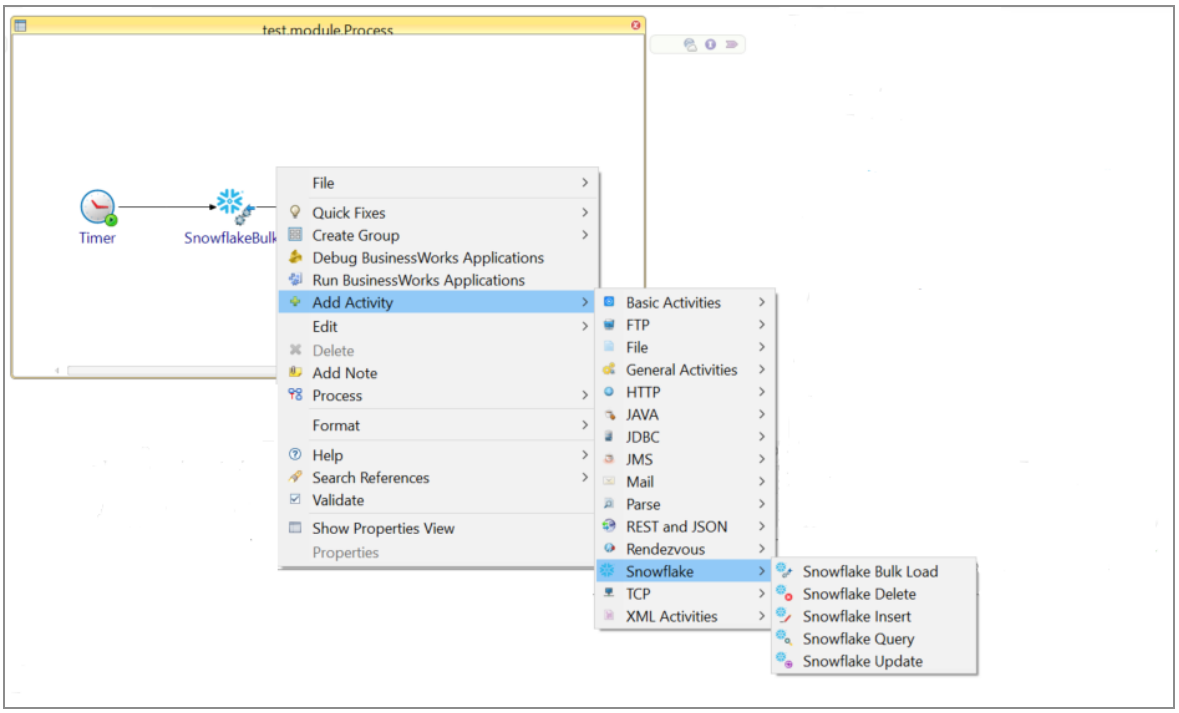
Before you begin

1. Ensure that an empty process is created when creating a project. See Creating a Project for details.
2. Ensure that you have established a Snowflake connection. See Snowflake JDBC Connection Resource for details.

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 drag it to the Process editor.

   For example, select and drop the Timer activity from the General Activities palette.
3. Click  and drag to create the activity. In this manner, create links between the activities and configure the condition types.



4. Configure the added Snowflake activity. See as described in Snowflake Palette.

> **Note:** A Snowflake JDBC Connection shared resource is required when configuring the activities. For details on creating the Snowflake JDBC Connection shared resource, see Creating and Configuring the Snowflake JDBC Connection Resource.

5. Save the project.

Testing a Process

After configuring a process, you can test the process to check whether the process completes the defined task.

Before you begin

Ensure that you have configured a process. For details, see Configuring a Process.

Procedure

1. On the toolbar, click

 **Debug > Debug Configurations**.

2. Click **BusinessWorks Application > BWApplication** in the left pane.

   By default, all the applications in the current workspace are selected on the **Applications** tab. Ensure that only the application you want to debug is selected on the **Applications** tab in the right pane.

3. Click **Debug** to test the process in the selected application.

   TIBCO Business Studio for BusinessWorks changes to the Debug perspective. The debug information is displayed in the Console view.

4. On 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 Applications

After testing, if the configured process works as expected, you can deploy the application that contains the configured process to a runtime environment. After deploying applications, you can manage TIBCO ActiveMatrix BusinessWorks applications by using TIBCO® Enterprise Administrator.

Before you begin

The following tasks are required before deploying applications:

- Creating a Project
- Generating an EAR File

You can deploy an application EAR file in the command-line mode with the **bwadmin** utility. See *TIBCO ActiveMatrix BusinessWorks™ Administration* for more details about how to deploy an application.

Deploying an application involves the following tasks:

Procedure

1. Upload an EAR file.
2. Deploy an application.
3. Configure an application.
4. Start an application.

Generating an EAR File

Application archives are enterprise archive (EAR) files that are created in TIBCO Business Studio for BusinessWorks. An EAR file is required when deploying an application.

There are many ways to generate an EAR file, the following is one method. For more information, see *TIBCO ActiveMatrix BusinessWorks™ Administration*.

Before you begin

An application project must have been created, as described in Creating a Project.

Procedure

1. In the File Explorer view, click the **Open Directory to Browse** icon.
2. Select the folder where you want to generate the EAR file and click **OK**.

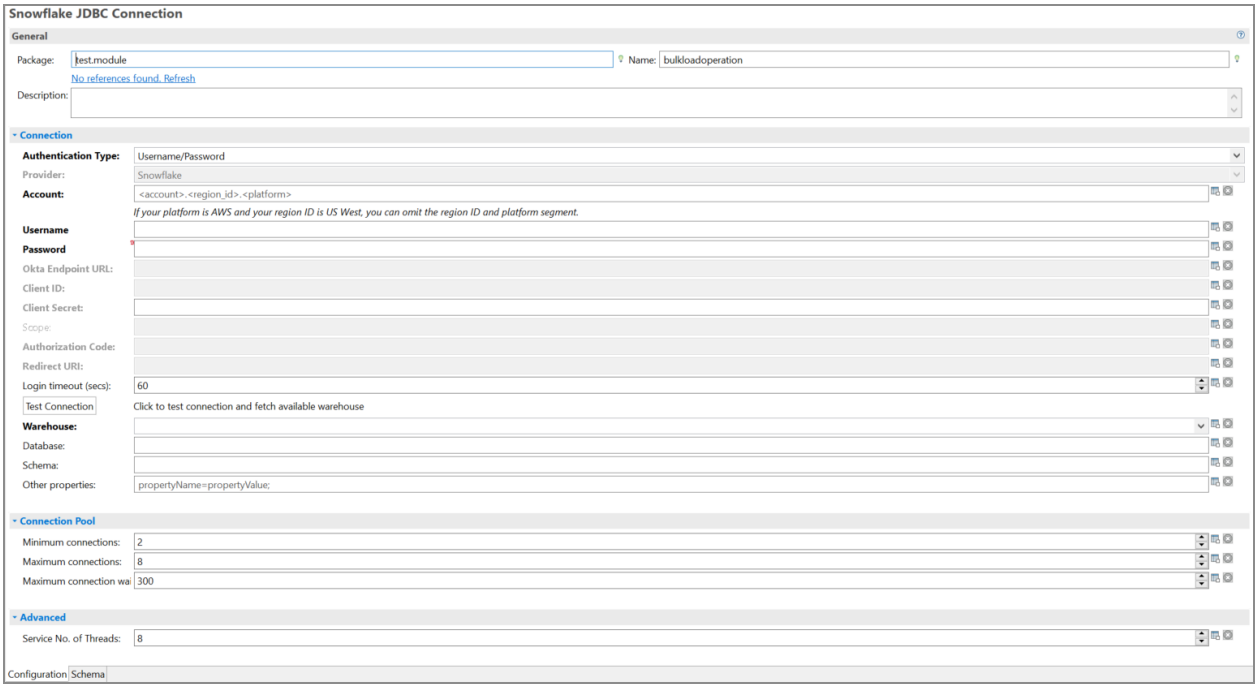
   The new folder is displayed in the File Explorer view.
3. Drag the application from the Project Explorer view to the new folder in the File Explorer view.

   The EAR file is generated with the name <name>.<application>_<version>.ear.

Snowflake JDBC Connection Resource

You can use the Snowflake JDBC Connection shared resource to connect to the Snowflake data warehouse system using the Snowflake JDBC driver and download the entity metadata at design time.

The Snowflake JDBC Connection shared resource has two tabs - **Configuration** and **Schema**.

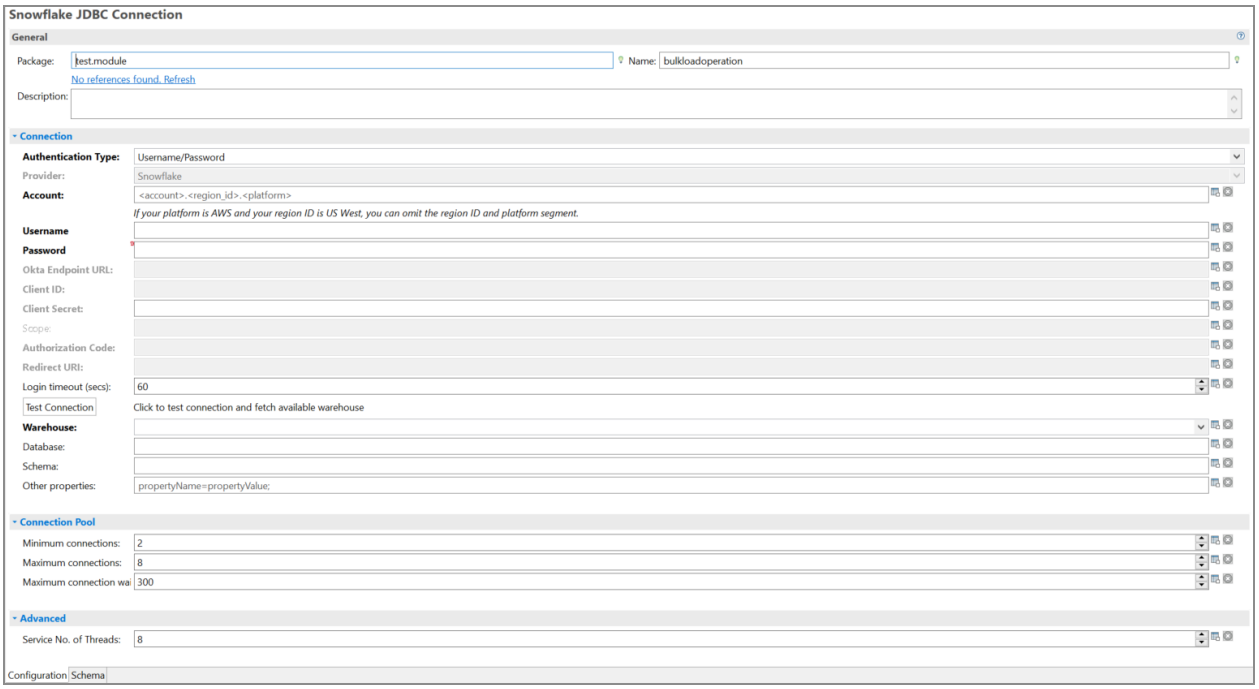


Note: Snowflake JDBC Connection Resource created in the shared module is not set to default when the Snowflake activities are created in the application module. You must manually select the Snowflake JDBC Connection Resource from the shared module.

Configuration

On the **Configuration** tab, you can specify information to connect to the Snowflake data warehouse. The **Configuration** tab has the following sections: General, Connection,

Connection Pool, and Advanced.



General

The **General** panel fields are described below.

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

Connection

The **Connection** panel has the following fields:

Field	Module Property	Description
Authentication type	No	Select the authentication mechanism, in this version the following authentication types are supported: • Username/Password • Federated Authentication And SSO • OAuth
Provider	No	Select the provider for a particular **Authentication Type**. In this version, the following authentication providers are available: • Snowflake • Okta
Account	Yes	Enter the Snowflake account name to be used for the connection. It must be specified in the format: [account.regionid.platform] **Note:** Do not use underscores (_) in the account name. Use hyphens (-).
Username	Yes	Enter the username of the account to be used for authentication
Password	Yes	Enter the password of the account to be used for authentication.
Okta Token Endpoint	Yes	Enter the token endpoint provided by Okta for OAuth authentication. **Note:** This field is enabled only when **Authentication Type** is OAuth and **Provider** is Okta.
Okta Username	Yes	Enter the username of the Okta account to be used for authentication.

Field	Module Property	Description
Okta Password	Yes	Enter the password of the Okta account to be used for authentication.
Okta Endpoint URL	Yes	Enter the Okta account URL: `https://<okta_account_name>.okta.com` This field is enabled only when **Authentication Type** is 'Federated Authentication and SSO' and **Provider** is Okta.
Client ID	Yes	Enter the client ID of the OAuth user.
Client Secret	Yes	Enter the client secret of the OAuth user.
Scope	Yes	Limits the operations and roles permitted by the access token and what you can access after instantiating a Snowflake session. Enter a scope for Okta OAuth token in the format "session:role:<role_name>". This field is enabled only when **Authentication Type** is OAuth and **Provider** is Okta. **Note:** This field is not mandatory.
Authorization Code	Yes	Authorizes generation of access token and refresh token. This field is enabled only when **Authentication Type** is OAuth and **Provider** is Snowflake. **Note:** For Snowflake OAuth, the authorization code has to be generated using a browser. It can be used only once. A new authorization code needs to be generated at the time of creating a connection. Using **Authorization Code**, access token and refresh token are generated. Validity of a refresh token can be extended up to 3 months to 1 year as per Snowflake policies.

Field	Module Property	Description
Redirect URI	Yes	The redirect URI of the integration object. This field is enabled only when **Authentication Type** is OAuth and **Provider** is Snowflake.
Login timeout (secs)	Yes	The time (in seconds) to wait for a successful database connection. The default value is 60 seconds.
Warehouse	Yes	The warehouse to run queries, which is selected from a list of warehouses.
Database	Yes	Default Database name to be used. Optional.
Schema	Yes	Default Schema name to be used. Optional.
Other Properties	Yes	Additional connection properties in the format [PropertyName=PropertyValue;]. Optional.

Note: For configuring federated authentication and SSO with Okta and external OAuth with Okta. For more information on **Other Properties**, see the Snowflake documentation.

Note: You can make Basic Authentication through the username and password more secure by restricting traffic from a particular IP through network and certificates.

Note: You can assign multiple roles to a user in Snowflake by mentioning these roles in the **Other Properties** field. The last role that you mention takes precedence when establishing a connection.

Connection Pool

Field	Module Property	Description
Minimum connections	Yes	The initial number of connections that are created when the pool is started. The default value is 2.
Maximum connections	Yes	The maximum number of connections that can be allocated from the connection pool at the same time. The default value is 8.
Maximum connection wait (secs)	Yes	The maximum number of seconds that the pool must wait for a connection to be returned before generating an exception. The default value is 300 seconds.

Advanced

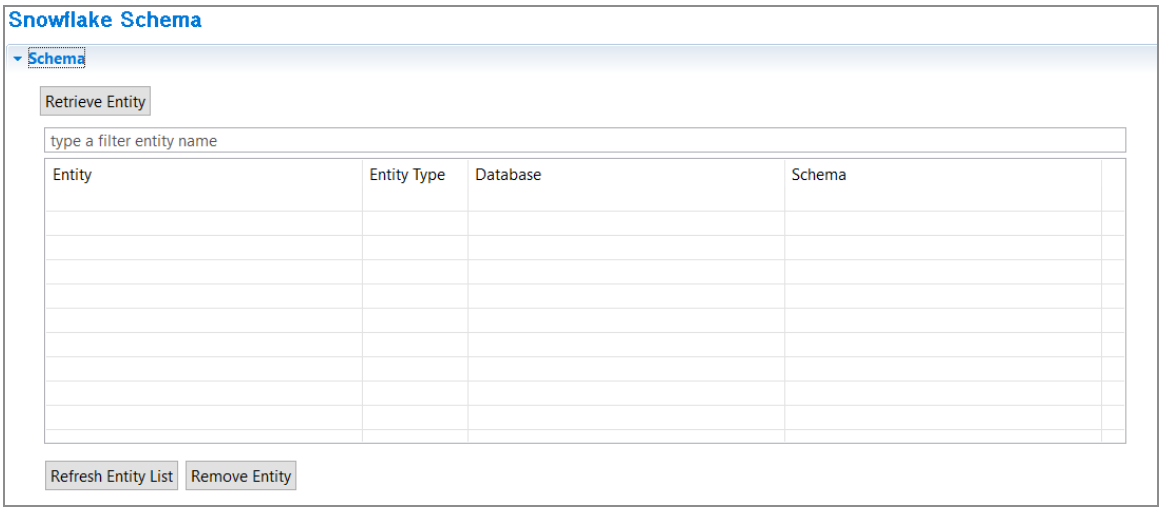
In the Advanced panel of the **Configuration** tab, you can specify additional information about the number of threads needed for processing requests.

Field	Module Property	Description
Service No. of Threads	Yes	The number of concurrent threads for processing requests to the activity. Default value: 8. (A value less than 1 is automatically changed to the default value).

Schema

The **Schema** tab lists the entities that have a downloaded metadata. To filter for specific Entities, enter the search string for the **Entity** column.

Note: TABLE and Views entities are supported.



You can perform the following tasks on the **Schema** tab:

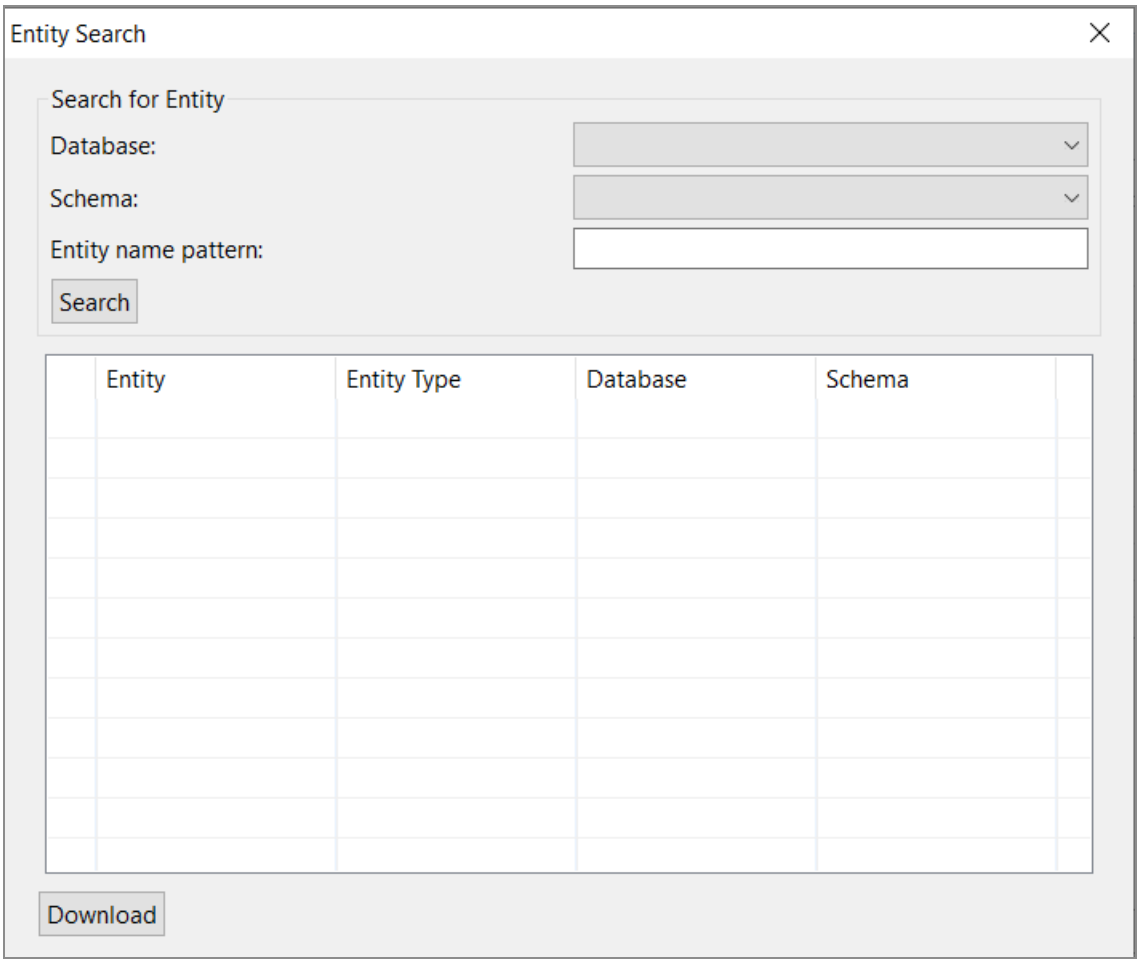
- **Retrieve Entity:** Search for entities and download the metadata.
- **Refresh Entity List:** Update the entity list and metadata of the entities previously downloaded.

  Note: For the changes to reflect in the activity, click on the activity once.

- **Remove Entity:** Remove the downloaded entity from the list.

  Note: Only one entity can be removed at a time.

Searching for an Entity



After clicking **Retrieve Entity**, the Entity Search page is displayed. You can search an entity here by selecting the database, schema, and entity name pattern.

Note: If entity name pattern has an underscore, it is escaped internally by default. Hence, additional escaping is not required.

XSD Datatype Mapping

Snowflake supports many alias and synonyms for datatypes. The following table shows how those datatypes are mapped to XSD data types in the plug-in.

Snowflake Column Definition	XSD Type
Number(N,10)	integer
Number(N,2)	decimal
Decimal(N,2)	decimal
Numeric(N,2)	decimal
Int	integer
Integer	integer
Bigint	integer
Smallint	integer
Double	double
Float	double
Double Precision	double
Real	double
Varchar	string
Varchar(N)	string
Char	string
Char(N)	string
Character	string
Character(N)	string
String	string

Snowflake Column Definition	XSD Type
String(N)	string
Text	string
Text(N)	string
Binary	base64Binary
Binary(N)	base64Binary
varbinary	base64Binary
Boolean	boolean
Date	date
Time	time
Timestamp	dateTime
Timestamp_LTZ	dateTime
Timestamp_NTZ	dateTime
Timestamp_TZ	dateTime
Variant	anyType
Object	anyType
ARRAY	anyType

Snowflake Palette

A palette groups the activities that connect to the same external applications together. The Snowflake palette is added to TIBCO Business Studio for BusinessWorks after installing TIBCO ActiveMatrix BusinessWorks™ Plug-in for Snowflake.

The Snowflake palette contains the following activities.

- Snowflake Insert Activity
- Snowflake Query Activity
- Snowflake Update Activity
- Snowflake Delete Activity
- Snowflake Bulk Load Activity

Snowflake Insert Activity

You can use this activity to run the insert SQL statements on the Snowflake data warehouse. You can insert multiple rows in the database in batches. Error in one batch does not stop the execution of the subsequent batch. To stop the execution of batch for error, select the **Fault on Batch Failure** checkbox. The failure record of a batch depends on the behavior of the underlying Snowflake JDBC driver and Snowflake data warehouse engine. If a batch fails to insert one or more records, the detailed message is displayed in logs including the batch number, reason of failure, and rows that failed to insert. The insertion operation results in the formation of total rows attempted and total rows affected.

General

The **General** tab contains the following fields.

Field	Module Property?	Visual Diff?	Description
Name	No	Yes	Specify the name to be displayed as the label for the activity in the process.
Snowflake Connection Resource	Yes	Yes	Click the **Create New Process Property** icon to select a Snowflake JDBC Connection shared resource. If no matching Snowflake JDBC Connection shared resource is found, click **Create Shared Resource** to create one. For more details, see Creating and Configuring the Snowflake JDBC Connection Resource.
Create Table from XSD	No	Yes	Use this option to create a new table in Snowflake from the XSD provided by you and select table format from external schema. When this checkbox is selected, **Entity** field on **General** tab is disabled and **Table Name** field, **Load Schema** button from **SQL Builder** tab are enabled. Click **SQL Builder** > **Load Schema** to provide the XSD schema for the table.
Entity	No	Yes	Click the **Browse resource** icon to select an entity. The entities downloaded in Snowflake JDBC Connection shared resource's **Schema** tab is available for selection. **Note:** This field is disabled when **Create Table from XSD** checkbox is selected.
Batch Size	Yes	Yes	All incoming messages are inserted performing SQL, number specified here defines how many SQL statements can be batched together that can be represented as a single SQL. The default value is 100.
Time Out	Yes	Yes	Defines activity timeout in seconds. Default value

Field	Module Property?	Visual Diff?	Description
			is 0, means activity timeout disabled. **Note:** Insert activity works in multiple batches, so per batch timeout is evaluated using the timeout value specified. Refrain from using a smaller timeout value.

Description

On the **Description** tab, provide a short description for the activity.

SQL Builder

The **SQL Builder** tab displays the metadata of the entity selected in the **General** tab or details of the entity schema loaded using the **Load Schema** button.

Field	Editable	Description
Table Name	Yes	Displays the name of the table to be created. When **General** tab > **Create Table from XSD** checkbox is selected, this field is enabled. If you provide XSD using the **Load Schema** button, the value of this field is set to name of root complex element of the XSD. However, you can edit this value and set to desired value.
Field Name	No	Displays name of the column.
Data Type	No	Displays the data type of the column.
Primary Key	Yes, if 'Metadata' table is loaded using XSD provided with 'Load Schema'	Displays if the column is a primary key. The column is editable if table is selected from external schema.

Field	Editable	Description
	button from **SQL Builder** tab. No, if 'Metadata' table is loaded using Entity field from **General** tab.	
Not Null	No	Displays if the column accepts null value. The checkbox is selected if the constraints to the column in snowflake DB is marked as 'Not Null'. The checkbox is cleared if the constraints to the column in snowflake DB is not marked as 'Not Null'. This checkbox is non-editable.
Dimension	No	Displays the dimension of the column.
Values	Yes	Displays the fields that must be part of the **Values** clause for an insert SQL statement. By default, **Values** column is selected for all the fields.
Load Schema	No	When **General** tab > **Create Table from XSD** checkbox is selected, this field is enabled. If you click this button, a dialog is opened where you can create or browse XSD present in the workspace. Once you select XSD in the dialog, its details are displayed into 'Metadata' table and the 'Table Name' field is set to name of root complex element of the XSD.

Advanced

This tab has the following fields:

Field	Module Property?	Description
Override Database Name	Yes	Overrides Database name specified in snowflakejdbcResource.
Override Schema Name	Yes	Overrides Schema name specified in snowflakejdbcResource.
Interpret Empty String as NULL	Yes	If this checkbox is selected, empty string is interpreted as NULL value.
Fault On Batch Failure	Yes	If this checkbox is selected and if an error is encountered in one of the batches, the batch insert process stops and does not continue to the next batch.
Merge	No	If this checkbox is selected and if the record already exists, the insert process updates the record by executing the merge query. **Note:** Merge is supported only when batch size is 1.
Merge On Columns	Yes	If the **Merge** checkbox is selected, this field is enabled. Enter column names on which the merge operation should be performed. For more than one column names, enter column names separated by comma. Column names are not case-sensitive.

Input

All the fields in the **SQL Builder** tab that have **Values** column selected are a part of the **Input** tab. The **Input** tab displays the input schema of the activity as a tree structure. The information in the schema depends on the fields selected on the **SQL Builder** tab.

Note: It is necessary to have input data for all columns specified in the **Merge On Columns** field.

Output

The **Output** tab displays the **rowsAttempted** and **rowsAffected** fields. The **rowsAttempted** field holds the count of the number of rows that were attempted by Snowflake Insert activity and the **rowsAffected** field holds the count of the number of rows inserted successfully after the Snowflake Insert activity is invoked. The difference between these two is the number of rows that failed to insert. The **Output** tab displays the output schema of the activity as a tree structure. The output is read-only.

Note: If new columns are added in the source table, the target table gets altered during runtime and the new columns are added in the target table.

Fault

The **Fault** tab lists exceptions that are thrown by this activity.

Snowflake Query Activity

Use this activity to run the query SQL statements on the Snowflake data warehouse.

Note: You must use the Date and Time data type conversion functions provided by Snowflake to create a query with the Timestamp data type.

General

The **General** tab contains the following fields.

Field	Module Property?	Visual Diff?	Description
Name	No	Yes	Specify the name to be displayed as the label for the activity in the process.
Snowflake Connection Resource	Yes	Yes	Click the **Create New Process Property** icon to select a Snowflake JDBC Connection shared resource. If no matching Snowflake JDBC Connection shared resource is found, click **Create Shared Resource** to create one. For more details, see Creating and Configuring the Snowflake JDBC Connection Resource.
Entity	No	Yes	Click the **Browse resource** to select an entity. The entity includes database tables and database views. The entities downloaded in the Snowflake JDBC Connection shared resource **Schema** tab are available for selection.
Maximum Rows	Yes	Yes	The maximum number of rows to retrieve. To retrieve all rows, specify 0. The default value is 100.
Time Out	Yes	Yes	Defines activity timeout in seconds. The default value is 100 seconds.

Description

On the **Description** tab, provide a short description for the activity.

SQL Builder

You can use the **SQL Builder** tab to configure SQL queries.

Field	Editable	Description
SQL	No	Opens the Eclipse SQL Builder. For more

Field	Editable	Description
		information about using the SQL Query Builder to visually build queries, see the Eclipse documentation.
Fetch	No	Creates the output schema based on the SQL query. If the SQL query does not conform to the Snowflake syntax, a compilation error occurs on clicking the **Fetch** button.
Statement	Yes	The SQL SELECT statement to be performed on the database. You can use an SQL Builder wizard to build the query for the Snowflake database.
Prepared Statement	Yes	The **Prepared Statement** contains the **Field Name** and the **Data Type** fields. You can provide the value for all the fields defined in the Prepared Statement table that are displayed on the **Input** tab of the activity. Each prepared statement corresponds to the question mark in the same position in the SQL statement. That is, the first prepared statement in the list corresponds to the first question mark, the second prepared statement in the list corresponds to the second question mark, and so on. Ensure that the parameters in this field correctly correspond to the statement. **Note:** The field names added in the **Prepared Statement** field must be compliant with the XML element naming standard. For invalid names, an error occurs when you run the application.
Parameter	Yes	Displays the fields that must be part of the WHERE clause of a query SQL statement.

Advanced

This tab has the following fields:

Field	Module Property?	Visual Diff?	Description
Override Database Name	Yes	Yes	Overrides the Database name specified in snowflakejdbcResource.
Override Schema Name	Yes	Yes	Overrides the Schema name specified in snowflakejdbcResource.
Interpret Empty String as NULL	Yes	Yes	If this checkbox is selected, the empty string is interpreted as a NULL value.

Input

All the fields in the **SQL Builder** tab that have **Parameter** column selected are a part of the **Input** tab. The **Input** tab displays the input schema of the activity as a tree structure. The information in the schema depends on the fields selected on the **SQL Builder** tab.

Output

All the fields in the **SQL Builder** tab that have the **Selected** column selected are a part of the **Output** tab. Output of Snowflake Query activity holds multiple records. The **Output** tab displays the output schema of the activity as a tree structure. The output is read-only. The information in the schema depends on the fields selected on the **SQL Builder** tab.

Fault

The **Fault** tab lists exceptions that are generated by this activity.

Snowflake Update Activity

Use this activity to run the update SQL statements on the Snowflake database. You can update multiple rows in the database. This activity returns the information in the form of

total rows affected.

General

The **General** tab contains the following fields.

Field	Module Property?	Visual Diff?	Description
Name	No	Yes	Specify the name to be displayed as the label for the activity in the process.
Snowflake Connection Resource	Yes	Yes	Click the **Create New Process Property** icon to select a Snowflake JDBC Connection shared resource. If no matching Snowflake JDBC Connection shared resource is found, click **Create Shared Resource** to create one. For more details, see Creating and Configuring the Snowflake JDBC Connection Resource.
Entity	No	Yes	Click the **Browse resource** icon to select an entity. The entities downloaded in Snowflake JDBC Connection shared resource's **Schema** tab is available for selection.
Time Out	Yes	Yes	Defines activity timeout in seconds. Default value is 0, means activity timeout disabled.

Description

On the **Description** tab, provide a short description for the activity.

SQL Builder

The **SQL Builder** tab displays the metadata of the entity selected in the **General** tab.

Field	Editable	Description
Field Name	No	Displays name of the column.
Data Type	No	Displays the data type of the column.
Primary Key	No	Displays if the column is a primary key.
Not Null	No	Displays if the column accepts null value. The checkbox is selected if the constraints to the column in snowflake DB is marked as 'Not Null'. The checkbox is cleared if the constraints to the column in snowflake DB is not marked as 'Not Null'. This checkbox is non-editable.
Dimension	No	Displays the dimension of the column.
Values	Yes	Displays the fields that must be a part of the SET clause for an update SQL statement. By default **Values** column is selected for all the fields.
Parameter	Yes	Displays the fields that must be part of the WHERE clause of an update SQL statement.

Advanced

This tab has the following fields:

Field	Module Property?	Description
Override Database Name	Yes	Overrides Database name specified in snowflakejdbcResource.
Override	Yes	Overrides Database name specified in snowflakejdbcResource.

Field	Module Property?	Description
Schema Name		
Interpret Empty String as NULL	Yes	If this checkbox is selected, empty string is interpreted as NULL value.
Merge	No	If this checkbox is selected and if the record does not exist, the update process inserts the record by executing the merge query.
Merge On Columns	Yes	This field is enabled only when the **Merge** checkbox is selected. Enter the column names on which the merge operation should be performed. For more than one column names, enter column names separated by comma. Column names are not case-sensitive.
Create Table If None Exists	Yes	This field is enabled only when the **Merge** checkbox is selected. If this checkbox is selected and if the table does not exist in Snowflake, the update process creates a new table based on the entity metadata downloaded at design time.

Input

All the fields in the **SQL Builder** tab that have **Values** and **Parameter** columns selected are a part of the **Input** tab. The **Input** tab displays the input schema of the activity as a tree structure. The information in the schema depends on the fields selected on the **SQL Builder** tab.

Note: The columns which are not mapped are ignored automatically from the SQL query.

Note: It is necessary to have input data for all columns specified in the **Merge On Columns** field.

Output

The **Output** tab displays the **rowsAffected** field which holds the count of the number of rows affected after the Snowflake Update activity is invoked. The **Output** tab displays the output schema of the activity as a tree structure. The output is read-only.

Fault

The **Fault** tab lists exceptions that are thrown by this activity.

Snowflake Delete Activity

You can use this activity to remove a single or all rows from a table in Snowflake Data warehouse.

General

The **General** tab contains the following fields.

Field	Module Property?	Visual Diff?	Description
Name	No	Yes	Specify the name to be displayed as the label for the activity in the process.
Snowflake Connection Resource	Yes	Yes	Click the **Create New Process Property** icon to select a Snowflake JDBC Connection shared resource. If no matching Snowflake JDBC Connection shared resource is found, click **Create Shared Resource** to create one. For more details, see Creating and Configuring the Snowflake JDBC Connection Resource.

Field	Module Property?	Visual Diff?	Description
Entity	No	Yes	Click the **Browse resource** icon to select an entity. The entities downloaded in Snowflake JDBC Connection shared resource's **Schema** tab is available for selection.
Time Out	Yes	Yes	Defines activity timeout in seconds. The default value is 100 seconds.

Description

On the **Description** tab, provide a short description for the activity.

SQL Builder

The **SQL Builder** tab displays the metadata of the entity selected on the **General** tab.

Field	Editable	Description
Field Name	No	Displays name of the column.
Data Type	No	Displays the data type of the column.
Primary Key	No	Displays if the column is a primary key.
Not Null	No	Displays if the column accepts null value. The checkbox is selected if the constraints to the column in snowflake DB is marked as 'Not Null'. The checkbox is cleared if the constraints to the column in snowflake DB is not marked as 'Not Null'. This checkbox is non-editable.
Dimension	No	Displays the dimension of the column.
Parameter	Yes	Displays the fields that must be part of the WHERE clause of a query SQL statement.

Advanced

This tab has the following fields:

Field	Module Property?	Description
Override Database Name	Yes	Overrides Database name specified in snowflakejdbcResource.
Override Schema Name	Yes	Overrides Schema name specified in snowflakejdbcResource.
Interpret Empty String as NULL	Yes	If this checkbox is selected, empty string is interpreted as NULL value.
Merge	No	If this checkbox is selected and if the record does not exist, the delete process inserts the record by executing the merge query.
Merge On Columns	Yes	If the **Merge** checkbox is selected, this field is enabled. Enter column names on which the merge operation should be performed. For more than one column names, enter column names separated by comma. Column names are not case-sensitive.
Create Table If None Exists	Yes	This field is enabled only when the **Merge** checkbox is selected. If this checkbox is selected and if the table does not exist in Snowflake, the delete process creates a new table based on the entity metadata downloaded at design time.

Input

All the fields in the **SQL Builder** tab that have **Parameter** column selected are a part of the **Input** tab. The **Input** tab displays the input schema of the activity as a tree structure. The information in the schema depends on the fields selected on the **SQL Builder** tab.

Note: It is not mandatory to provide values for fields on the **Input** tab for Snowflake Delete activity. A default mapping is displayed at the **ActivityInput** element when the activity is created for the first time. In case the required symbol is visible, then right-click the **ActivityInput** element and click **Show Check** > **Repair** > **Select ActivityInput** field and click **OK**.

Note: It is necessary to have input data for all columns specified in the **Merge On Columns** field.

Output

The **Output** tab displays the **rowsAffected** field. The **rowsAffected** field holds the count of the number of rows inserted successfully after the Snowflake Delete activity is invoked. The **Output** tab displays the output schema of the activity as a tree structure. The output is read-only.

Fault

The **Fault** tab lists exceptions that are thrown by this activity.

Snowflake Bulk Load Activity

You can use this activity to load a large amount of data into the database. This activity uses the staging concept for loading data into the database table. The stage is an internal location or an external location, like the Amazon S3 that is used to store compressed or uncompressed data files. The plug-in can later load the data stored in the stage into the database table.

General

The **General** tab contains the following fields.

Field	Module Property?	Visual Diff?	Description
Name	No	Yes	Specify the name to be displayed as the label for

Field	Module Property?	Visual Diff?	Description
			the activity in the process.
Snowflake Connection Resource	Yes	Yes	Click the **Create New Process Property**  icon to select a Snowflake JDBC Connection shared resource. If no matching Snowflake JDBC Connection shared resource is found, click **Create Shared Resource** to create one. For more details, see Creating and Configuring the Snowflake JDBC Connection Resource.
Create Table from XSD	No	Yes	Use this option to create a table in Snowflake from the XSD provided by the user. When this checkbox is selected, the **Entity** field on the **General** tab is disabled. The **Table Name** field and the **Load Schema** button from the **SQL Builder** tab are enabled. Click the **Load Schema** button to provide the XSD schema for the table. **Note:** When **Stage Type** is set to Amazon S3, **Create Table from XSD** works only when **File Format** is set as Delimited Files.
Entity	No	Yes	Click the **Browse resource**  icon to select an entity. The entities downloaded in the Snowflake JDBC Connection shared resource **Schema** tab are available for selection. This field is disabled when the **Create Table from XSD** checkbox is selected.
Time Out	Yes		Defines activity timeout in seconds. The default value is 0 seconds.
Stage Type	No	Yes	There are four stage types available: • User Stage

Field	Module Property?	Visual Diff?	Description
			• Table Stage • Named Stage • Amazon S3 Select the stage type. The Snowflake plug-in supports both the internal stage and the external stage. User, Table, and Named stages are internal stages whereas Amazon S3 is an external stage.
Named Stage	Yes	Yes	Specify the name of the **Named Stage** defined in the Snowflake. The field is only enabled if **Stage Type** is selected as Named Stage or Amazon S3.
File Format	No	No	This combo box is enabled only when **Stage Type** is selected as Amazon S3. This field provides information on the format of the file that is present in the Amazon S3 bucket. The following file formats are supported: • Delimited Files • JSON • AVRO • ORC • PARQUET • XML **Note:** XML data can only be loaded into a variant type column.

For **Stage Type**=Amazon S3, you need to configure a storage integration object to delegate authentication responsibility for external cloud storage to a Snowflake identity and access management (IAM) entity. For more information, see the Snowflake documentation.

Description

On the **Description** tab, provide a short description for the activity.

SQL Builder

The **SQL Builder** tab displays the metadata of the entity selected in the General tab.

Field	Editable	Description
Table Name	Yes	Displays the name of the table to be created. When the **General** tab > **Create Table from XSD** checkbox is selected, this field is enabled. If you provide XSD using the **Load Schema** button, the value of this field is set to the name of the root complex element of the XSD. However, you can edit this value and set to the desired value. **Note:** Table Name can be added as a module property.
Field Name	No	Displays the name of the column.
Data Type	No	Displays the data type of the column.
Primary Key	Yes, if the 'Metadata' table is loaded using the XSD provided with the 'Load Schema' button from the SQL Builder tab. No, if the 'Metadata' table is loaded using the Entity field from the General tab.	Displays if the column is a primary key.

Field	Editable	Description
Not Null	No	Displays if the column accepts a null value. The checkbox is selected if the constraints to the column in snowflake DB is marked as 'Not Null'. The checkbox is cleared if the constraints to the column in snowflake DB are not marked as 'Not Null'. This checkbox is non-editable.
Dimension	No	Displays the dimension of the column.
Values	Yes	Displays the fields that must be a part of the **Values** clause for an INSERT SQL statement. By default, the **Values** column is selected for all the fields.
Load Schema	No	When the **General** tab > **Create Table from XSD** checkbox is selected, this field is enabled. If you click this button, a dialog is open where you can create or browse XSD present in the workspace. Once you select the XSD in the dialog, its details are displayed into the 'Metadata' table and the 'Table Name' field is set to the name of the root complex element of the XSD.

Advanced

This tab has the following fields:

Field	Module Property?	Visual Diff?	Description
Override Database Name	Yes	Yes	Overrides Database name specified in snowflakejdbcResource.
Override Schema	Yes	Yes	Overrides Schema name specified in snowflakejdbcResource.

Field	Module Property?	Visual Diff?	Description
Name			
Interpret Empty String as NULL	Yes	Yes	If this checkbox is selected, the empty string is interpreted as a NULL value.
Validation Mode	Yes	Yes	When this checkbox is selected, the plug-in returns errors in the data if present. The data is loaded into the table only when there are no errors present in the data. **Note:** Validation mode is not supported for semi-structured datatypes
Purge Stage Files	Yes	Yes	Select this checkbox to delete the stage files automatically once the data is loaded to the database table.
Compress Data	Yes	Yes	Select this checkbox to compress data in .gz format and load it to Snowflake Internal Stage. This is not applicable when loading data from External Stage like Amazon S3.
On Error	No	Yes	This field decides how the system behaves after encountering an error. The following actions are available: • CONTINUE • SKIP_FILE • SKIP_FILE_<num> • SKIP_FILE_<num>% • ABORT_STATEMENT
Skip a File if Error Count	Yes	Yes	This text box is enabled if the SKIP_FILE_<num>

Field	Module Property?	Visual Diff?	Description
			option is selected from the On Error dropdown list. It skips the processing of a file if the number of errors present in the data file is equal to the error count specified.
Skip a File if Error Percentage	Yes	Yes	This text box is enabled if the SKIP_FILE_<num>% option is selected from the On Error dropdown list. It skips the processing of a file if the percentage of errors in the data file is equal to the error percentage specified.
Merge	No	No	When you select the **Merge** checkbox, if the record exists, it is updated by running the MERGE query. If the record does not exist, it is inserted.
Merge On Columns	Yes	No	**Merge On Columns** is available only when the **Merge** checkbox is selected. Enter the names of the columns on which the merge operation must be performed. For more than one column, enter the column names separated by comma. The column names are not case-sensitive. **Note:** Provide all the column names in uppercase in Parquet, Avro, XML, or JSON format.

Input

For **User**, **Table**, and **Named** stage types, all the fields in the **SQL Builder** tab that have the **Values** column selected are a part of the Input tab. The information in the schema depends on the fields selected on the SQL Builder tab.

For **Stage Type**=Amazon S3, the schema on the **Input** tab is populated based on the file format selected on the **General** tab.

The **Input** tab displays the input schema of the activity as a tree structure.

The **loadOptions** field is not available when the **Stage Type** is Amazon S3 on the **General** tab and when the **Merge** checkbox is selected on the **Advanced** tab.

Output

The **Output** tab shows the output schema of the activity as a tree structure. The output is read-only. The **Output** tab displays the following fields:

Column Name	Description
FILE	Name of source file and relative path to the file
STATUS	Contains the following options: loaded, load failed, load_skipped, or partially_loaded.
ROWS_PARSED	Number of rows parsed from the source file
ROWS_LOADED	Number of rows loaded from the source file
ERROR_LIMIT	If the number of errors reaches this limit, then cancel.
ERRORS_SEEN	Number of error rows in the source file
FIRST_ERROR	First error of the source file
FIRST_ERROR_LINE	Line number of the first error
FIRST_ERROR_ CHARACTER	Position of the first error character
FIRST_ERROR_COLUMN_ NAME	Column name of the first error

Note: If new columns are added in the source table, the target table gets altered during runtime and the new columns are added in the target table. This is applicable only when the table does not contain any columns with semi-structured data types. The alter table operation is applicable only for delimited files.

Fault

The **Fault** tab lists exceptions that are generated by this activity.

Working with Sample Projects

The plug-in packages sample projects with the installer which help to understand how TIBCO ActiveMatrix BusinessWorks™ Plug-in for Snowflake works.

The sample projects are located at TIBCO_HOME/bw/palettes/snowflake/*<version_number>*/samples directory. The following sample projects are available:

- The MultipleOperations Project
- The BatchOperations Project
- The BulkLoadOperations Project

Importing Sample Projects

Before you begin

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

Procedure

1. Depending on the platform applicable to your plug-in, start TIBCO Business Studio for BusinessWorks:

Platform	Step
Microsoft Windows	Click **Start > All Programs > TIBCO > *TIBCO_HOME* > TIBCO Business Studio *<studio_version>* > Studio for Designers**
Linux or macOS	Run the TIBCO Business Studio for BusinessWorks executable file in the *TIBCO_HOME*/studio/*<studio_version>*/eclipse directory.

   For the list of platforms supported by your plug-in, see the Readme file.

2. Click **File > Import**.

3. In the **Import** dialog, expand the **General** folder and select the **Existing Studio Projects into Workspace** item. Click **Next**.
4. Next to the **Select root directory** field, click **Browse** to locate the samples. Click **Open**.

   The sample project is located in the TIBCO_HOME\bw\palettes\snowflake\<version_number>\samples directory.
5. Click **Finish**.

Result

The sample project is imported to TIBCO Business Studio for BusinessWorks.

Setting Up a Project

Before running the project you must configure it.

Before you begin

Run the DDL statement provided in the DDL_worksheet.txt in PUBLIC schema under TEST_DB database of your snowflake account. DDL_worksheet.txt can be located in the TIBCO_HOME/bw/palettes/snowflake/*<version_number>*/samples directory.

Procedure

1. In TIBCO Business Studio for BusinessWorks, expand the imported project in the **Project Explorer** view.
2. Expand the **Module Descriptors** resource, and then double-click **Module Properties**.
3. In the Module Properties panel, set the values for **Account**, **Username**, **Password** and **Warehouse** fields.
4. For the **BatchOperations** sample, additionally set value for the **Filename** field as the absolute path of employees.csv file, which can be located in the TIBCO_HOME/bw/palettes/snowflake/*<version_number>*/samples directory.
5. From the menu bar, click **File** > **Save** to save the project.

Running the Project

After setting up the sample project, you can run the project to see how the plug-in works.

Before you begin

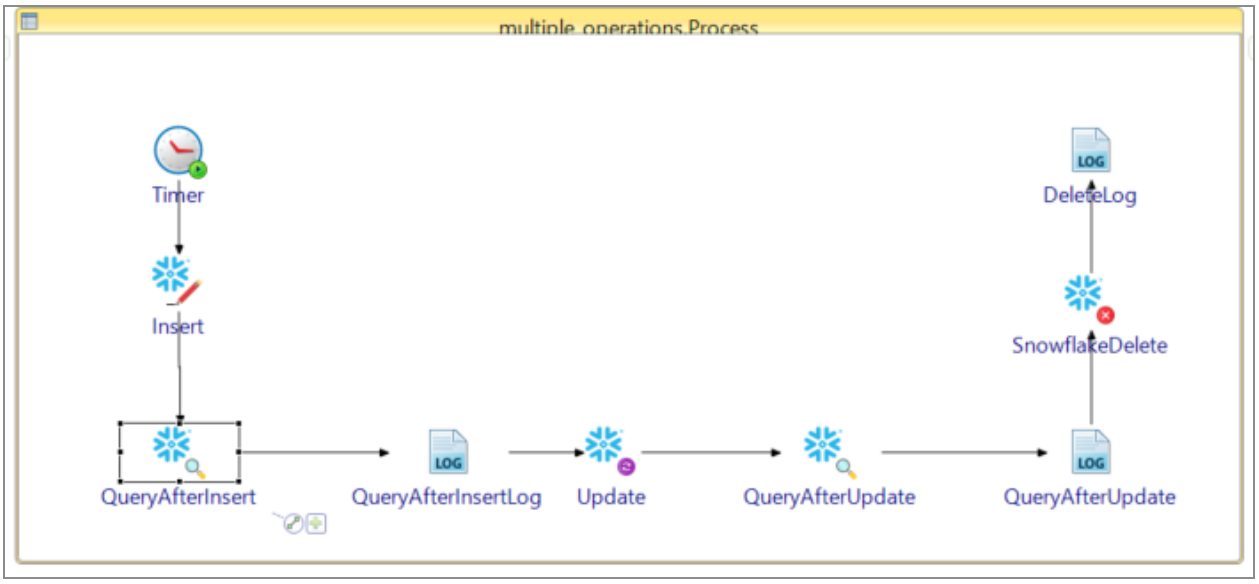
Ensure that you have set up the sample project to TIBCO Business Studio as described in Setting Up a Project.

Procedure

1. To run the selected process, from the menu, click **Run > Run Configurations**.
2. In the Run Configurations dialog, expand **BusinessWorks Application**, and then click **BWApplication**.
3. On the **Applications** tab, all the sample applications are selected. You can clear the sample applications that you do not want to run.
4. Click **Run** to run the process.
5. Click the **Terminate** icon to stop the process.

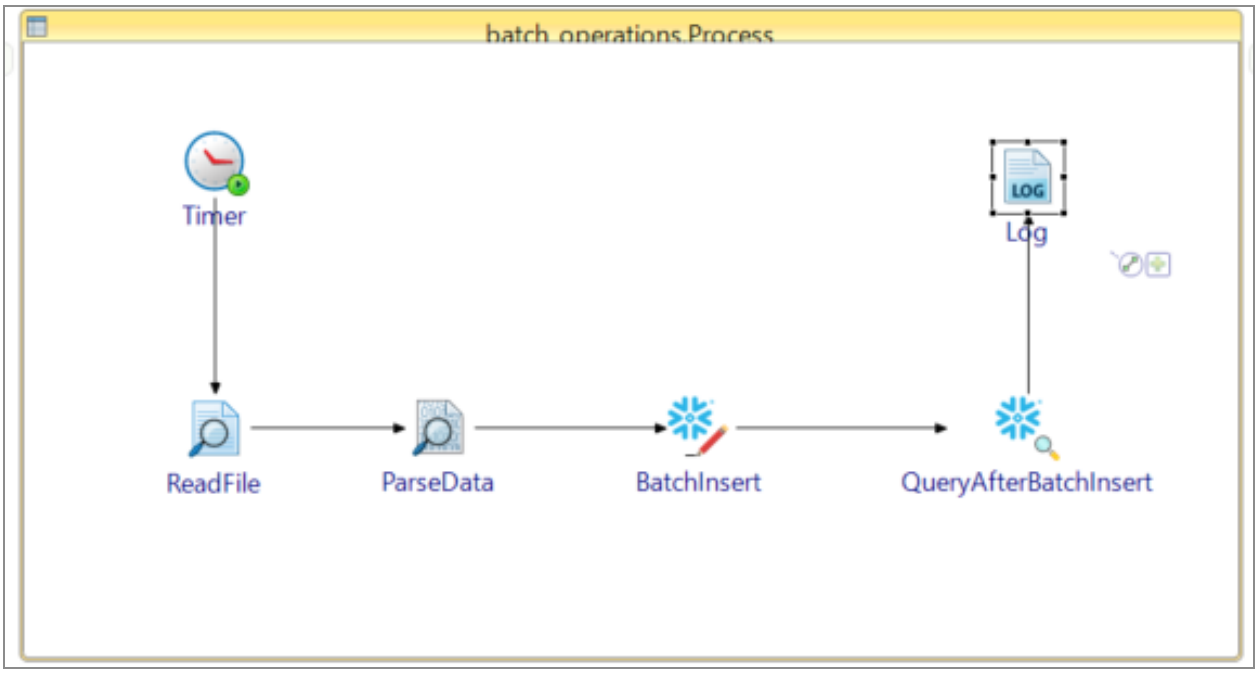
The MultipleOperations Project

The MultipleOperations project contains the **Multiple_Operations** process. This ActiveMatrix BusinessWorks process provides an example where multiple operations like Insert, Query, Update, and Delete are performed sequentially on the following Snowflake table TEST_DB.PUBLIC.EMPLOYEES.



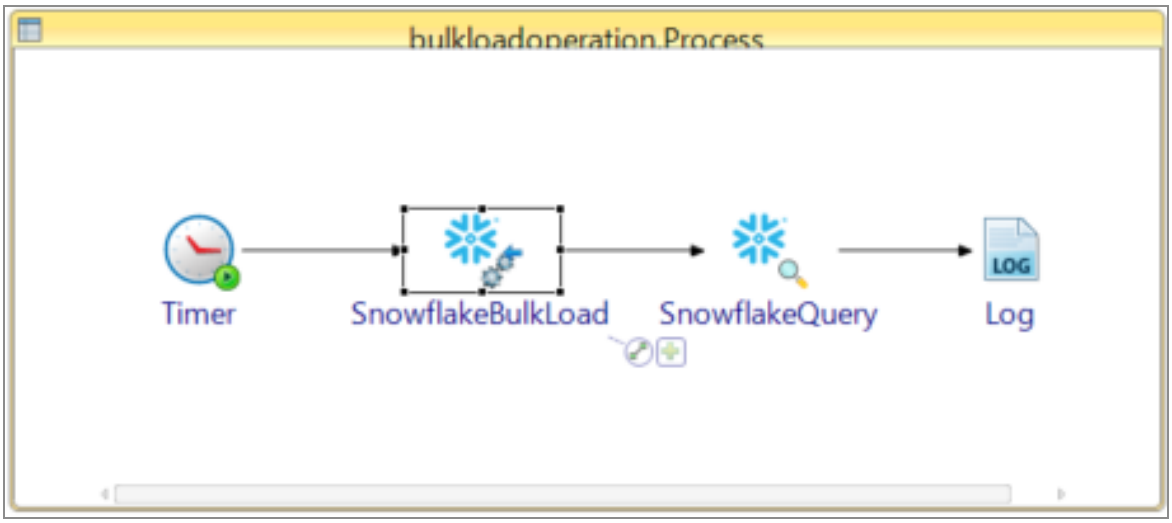
The BatchOperations Project

The BatchOperations project contains the **Batch_Operations** process. This ActiveMatrix BusinessWorks process provides an example where a batch of 456 records is inserted in the following Snowflake table TEST_DB.PUBLIC.EMPLOYEES.



The BulkLoadOperations Project

The BulkLoadOperations project contains the **BulkLoadOperations** process. This ActiveMatrix BusinessWorks process provides an example where you can upload data in a bulk to the TEST_DB.PUBLIC.EMPLOYEES Snowflake table.



Troubleshooting

This topic contains basic troubleshooting information for a project.
If errors occur when you run a process in TIBCO Business Studio for BusinessWorks, first clean up the project. Cleaning deletes all the old files and reorganizes the project.

Procedure

1. In the Project Explorer view, right-click the project and click **Refresh**.
2. Select **Project > Clean** to start the cleaning process.

Log Management

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

A logback.xml file is located in the TIBCO_HOME\bw\6.0\config\design\logback directory. Update this file to Setting Up Log Levels and Exporting Logs to a File.

By default, error logs are displayed in the Console view when you run a process in debug mode. 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.

Property	Description
Trace	Includes all information regarding the running process.
Debug	Indicates a developer-defined tracing message.
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.
Warn	Indicates that an abnormal condition occurred. Processing continues, but for best practice, you can contact the administrator to investigate it.
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.

Setting Up Log Levels

By default, the log level is Error. You can set the log level to change the log level to trace different messages.

Note: If neither the plug-in log nor the ActiveMatrix BusinessWorks log is configured in the logback.xml file, the error logs of the plug-in will be displayed in the Console view by default.

If the plug-in log is not configured but the ActiveMatrix BusinessWorks log is configured in the logback.xml file, the configuration for the ActiveMatrix BusinessWorks log is implemented by the plug-in.

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 Console Appender area to specify the log level for the plug-in.

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

   The level tag defines the log level and the value is one of the values mentioned in Log Levels.

   Note: When the level is set 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 one of the following nodes in the BusinessWorks Palette and Activity.

   Loggers area to specify a log level for the activity.

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

```
 <level value="DEBUG"/>
</logger>
<logger name="com.tibco.bw.palette.snowflake.runtime.update">
 <level value="DEBUG"/>
</logger>
<logger name="com.tibco.bw.palette.snowflake.runtime.delete">
<level value="DEBUG"/>
</logger>
"<logger name="com.tibco.bw.palette.snowflake.runtime.bulkload">
<level value="DEBUG"/></logger>
```

Note: The activities that are not configured with specific log levels also inherit log level configured for the plug-in or ActiveMatrix BusinessWorks.

4. To control the debug log level for the Snowflake JDBC shared resource, set the following parameters:

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

5. Save the file.

Exporting Logs to a File

To export plug-in logs to a file, modify the logback.xml.

Procedure

1. Navigate to the <BW_HOME>\bwcloud\<version_number>\config\design\logback directory and open the logback.xml file.

Note: When deploying an application in TIBCO Enterprise Administrator, you must navigate to the TIBCO_HOME\bw\domains\mydomain\appnodes\myspace\mynode directory to find the logback.xml file.

2. Add the following node to specify the file location:

```
<appender name="FILE" class="ch.qos.logback.core.FileAppender">
  <file>c:/snowflake.log</file>
    <encoder>
      <pattern>%d{HH:mm:ss.SSS} [%thread] %-5level %logger{36}-%msg%n</pattern>
    </encoder>
</appender>
```

The file tag defines the location to which the log is exported, and the value in the tag is the absolute path of the file.

> **Note:** The file path must include the file name.

3. To enable exporting the logs to a file, add the following node to the root node at the end of the logback.xml file:

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

4. Save the file.

Error Codes

The exceptions that are generated by the plug-in are listed with corresponding descriptions and resolutions.

Error Code	Role	Category	Messages	Description
TIBCO-BW-PALETTE-SNOWFLAKE_DATABASE_JDBC-500001	Error	BW-Plug-in	Unexpected error occurred.{0}	The message is shown when an unidentified error occurs while running the activity.
TIBCO-BW-PALETTE-SNOWFLAKE_DATABASE_JDBC-500002	Error	BW-Plug-in	Failed to initialize activity.{0}\nActivity Name={1}\nProcess={2}\nModule={3}\nDeploymentUnit={4}:{5}	The message is shown when initialization fails for an activity.
TIBCO-BW-PALETTE-SNOWFLAKE_DATABASE_JDBC-500003	Error	BW-Plug-in	Failed to decrypt obfuscated password for field : {0}	The message is shown when decryption of obfuscated password fails.
TIBCO-BW-PALETTE-SNOWFLAKE_DATABASE_JDBC-500005	Error	BW-Plug-in	Failed to create Prepared Statement. {0}	The message is shown when prepared statement creation fails.
TIBCO-BW-PALETTE-SNOWFLAKE_DATABASE_JDBC-500006	Error	BW-Plug-in	Failed to set max rows. {0}	The message is shown for Snowflake database Query activity if error occurs when setting

Error Code	Role	Category	Messages	Description
				maximum rows.
TIBCO-BW-PALETTE-SNOWFLAKE_DATABASE_JDBC-500007	Error	BW-Plug-in	Failed to set parameters in prepared statement. {0}	The message is shown when error occurs while setting parameters in prepared statement.
TIBCO-BW-PALETTE-SNOWFLAKE_DATABASE_JDBC-500008	Error	BW-Plug-in	Failed to retrieve values from ResultSet. {0}	The message is shown for Snowflake database Query activity if error occurs when retrieving values from ResultSet.
TIBCO-BW-PALETTE-SNOWFLAKE_DATABASE_JDBC-500009	Error	BW-Plug-in	Failed to execute query: {0}\n Error Message: {1}	The message is shown when error occurs in executing a query.
TIBCO-BW-PALETTE-SNOWFLAKE_DATABASE_JDBC-500010	Error	BW-Plug-in	Failed to close ResultSet. {0}	The message is shown for Snowflake database Query activity if error occurs in closing a ResultSet.
TIBCO-BW-PALETTE-SNOWFLAKE_DATABASE_JDBC-500011	Error	BW-Plug-in	Failed to close Prepared Statement. {0}	The message is shown when error occurs in closing prepared statement.
TIBCO-BW-PALETTE-SNOWFLAKE_DATABASE_	Error	BW-Plug-in	Failed to close Snowflake Database Connection. {0}	The message is shown when error occurs in closing

Error Code	Role	Category	Messages	Description
JDBC-500012				connection.
TIBCO-BW-PALETTE-SNOWFLAKE_DATABASE_JDBC-500013	Error	BW-Plug-in	Error occurred when generating XML Output for activity. {0}	The message is shown when error occurs while generating output for an activity.
TIBCO-BW-PALETTE-SNOWFLAKE_DATABASE_JDBC-500014	Error	BW-Plug-in	Invalid number of maximum rows entered in Activity: {0}. Value must be greater than or equal to 0, value was {1}	The message is shown for Snowflake database Query activity if you input negative value for maximum rows.
TIBCO-BW-PALETTE-SNOWFLAKE_DATABASE_JDBC-500016	Error	BW-Plug-in	{0}	The message is shown for Snowflake database Insert activity if insert of one or more records fails.
TIBCO-BW-PALETTE-SNOWFLAKE_DATABASE_JDBC-500017	Error	BW-Plug-in	Snowflake driver class not found. {0}	The message is shown when Snowflake JDBC driver class is not found
TIBCO-BW-PALETTE-SNOWFLAKE_DATABASE_JDBC-500018	Error	BW-Plug-in	Failed to upload data to Snowflake stage. {0}	The message is shown when Snowflake Bulk Load activity fails to upload data to Snowflake stage.
SNOWFLAKE_DATABASE_	Error	BW-Plug-in	Failed to close Input stream. {0}	The message is

Error Code	Role	Category	Messages	Description
JDBC-500019				shown when Snowflake activity fails to close input stream.
TIBCO-BW-PALETTE-SNOWFLAKE_DATABASE_JDBC-500020	Error	BW-Plug-in	{0}	The message is shown when there is a failure of data validation in the Snowflake Bulk Load activity.
TIBCO-BW-PALETTE-SNOWFLAKE_DATABASE_JDBC-500021	Error	BW-Plug-in	{0}	The message is shown when there is a failure of data validation in the Snowflake Bulk Load activity.
TIBCO-BW-PALETTE-SNOWFLAKE_DATABASE_JDBC-500022	Error	BW-Plug-in	Failed to create schema or table : {0}	The message is shown when there is a failure of schema or table creation.
TIBCO-BW-PALETTE-SNOWFLAKE_DATABASE_JDBC-500023	Error	BW-Plug-in	Failed to close statement {0}	The message is shown when Snowflake activity fails to close the statement.
TIBCO-BW-SR-SNOWFLAKE_DATABASE_CONNECTION-500001	Error	BW-Plug-in	Creating Shared Resource {0} failed due to [{1}]	The message is shown when error occurs during creation of shared resource.

Error Code	Role	Category	Messages	Description
TIBCO-BW-SR-SNOWFLAKE_DATABASE_CONNECTION-500002	Error	BW-Plug-in	Failed to decrypt obfuscated password for field : {0}	The message is shown when decryption of obfuscated password fails.
TIBCO-BW-SR-SNOWFLAKE_DATABASE_CONNECTION-500003	Error	BW-Plug-in	Failed to initialize connection pool for Shared Resource : {0} {1}	The message is shown when connection pool initialization fails.
TIBCO-BW-SR-SNOWFLAKE_DATABASE_CONNECTION-500004	Error	BW-Plug-in	Failed to close Snowflake Database Connection. {0}	The message is shown when error occurs in closing connection.
TIBCO-BW-SR-SNOWFLAKE_DATABASE_CONNECTION-500005	Error	BW-Plug-in	Failed to destroy connection pool for Shared Resource: {0} {1}	The message is shown when error occurs in destroying connection pool.
TIBCO-BW-PALETTE-SNOWFLAKE_DATABASE_JDBC-100001	Trace	BW-Plug-in	{0}	The message is shown while printing events for activity.
TIBCO-BW-PALETTE-SNOWFLAKE_DATABASE_JDBC-100002	Trace	BW-Plug-in	Connection successful with Snowflake database using DatabaseURL: {0}	The message is shown on successful connection with Snowflake Database.
TIBCO-BW-SR-SNOWFLAKE_DATABASE_CONNECTION-100001	Trace	BW-Plug-in	{0}	The message is shown while printing events for shared resource.

Error Code	Role	Category	Messages	Description
TIBCO-BW-PALETTE-SNOWFLAKE_DATABASE_JDBC-200001	Debug	BW-Plug-in	\nStart of the Activity {0}, \nInput received: \n {1} \n	Prints activity input.
TIBCO-BW-PALETTE-SNOWFLAKE_DATABASE_JDBC-200002	Debug	BW-Plug-in	\nActivity {0}, Output data: \n {1} \n Exit of Activity {2}	Prints activity output.
TIBCO-BW-PALETTE-SNOWFLAKE_DATABASE_JDBC-200003	Debug	BW-Plug-in	Query formed: {0}	Prints query formed.
TIBCO-BW-PALETTE-SNOWFLAKE_DATABASE_JDBC-200004	Debug	BW-Plug-in	Batch Size entered: {0}	The message is shown for Snowflake database Insert activity and prints batch size.
TIBCO-BW-PALETTE-SNOWFLAKE_DATABASE_JDBC-200005	Debug	BW-Plug-in	Maximum rows entered: {0}	The message is shown for Snowflake database Query activity and prints maximum rows entered.
TIBCO-BW-PALETTE-SNOWFLAKE_DATABASE_JDBC-200006	Debug	BW-Plug-in	Stage Type selected: {0}	The message is shown for stage types that is selected by user.
TIBCO-BW-PALETTE-SNOWFLAKE_	Debug	BW-Plug-in	Validation mode: {0}	The message is shown if user selected validation

Error Code	Role	Category	Messages	Description
DATABASE_ JDBC-200007				mode or not.
TIBCO-BW-PALETTE-SNOWFLAKE_ DATABASE_ JDBC-200008	Debug	BW-Plug-in	Purge stage files : {0}	The message is shown if user selected purge stage files checkbox or not.
TIBCO-BW-PALETTE-SNOWFLAKE_ DATABASE_ JDBC-200009	Debug	BW-Plug-in	On Error Input : {0}	The message is shown to display the **On Error** option selected by user.
TIBCO-BW-PALETTE-SNOWFLAKE_ DATABASE_ JDBC-200010	Debug	BW-Plug-in	Stage data to be uploaded : {0}	The message is shown to display stage data that would be uploaded.
TIBCO-BW-PALETTE-SNOWFLAKE_ DATABASE_ JDBC-200011	Debug	BW-Plug-in	Create Table If None Exists For activity : {0}	The message is shown if user selected **Create Table If None Exists** checkbox or not.
TIBCO-BW-PALETTE-SNOWFLAKE_ DATABASE_ JDBC-200012	Debug	BW-Plug-in	Create Schema Query : {0}	The message is shown to display the **Create Schema** query.
TIBCO-BW-PALETTE-SNOWFLAKE_ DATABASE_ JDBC-200013	Debug	BW-Plug-in	Create Table Query : {0}	The message is shown to display the **Create Table** query.
TIBCO-BW-SR-SNOWFLAKE_	Debug	BW-Plug-in	Creating Shared Resource {0}	The message is shown while creating

Error Code	Role	Category	Messages	Description
DATABASE_ CONNECTION-200001				the shared resource.
TIBCO-BW-SR-SNOWFLAKE_ DATABASE_ CONNECTION-200002	Debug	BW-Plug-in	Starting Shared Resource {0}	The message is shown while executing the shared resource.
TIBCO-BW-SR-SNOWFLAKE_ DATABASE_ CONNECTION-200003	Debug	BW-Plug-in	Stopping Shared Resource {0}	The message is shown while stopping the shared resource.
TIBCO-BW-SR-SNOWFLAKE_ DATABASE_ CONNECTION-200004	Debug	BW-Plug-in	Deleting Shared Resource {0}	The message is shown while deleting the shared resource.
TIBCO-BW-SR-SNOWFLAKE_ DATABASE_ CONNECTION-200005	Debug	BW-Plug-in	Number of thread is {0}	The message is shown to print number of thread.
TIBCO-BW-SR-SNOWFLAKE_ DATABASE_ CONNECTION-200006	Debug	BW-Plug-in	Minimum pool size is {0}	The message is shown to print minimum pool size.
TIBCO-BW-SR-SNOWFLAKE_ DATABASE_ CONNECTION-200007	Debug	BW-Plug-in	Maximum pool size is {0}	The message is shown to print maximum pool size.
TIBCO-BW-SR-SNOWFLAKE_	Debug	BW-Plug-in	Connection pool initialized for Shared	The message is

Error Code	Role	Category	Messages	Description
DATABASE_ CONNECTION-200008			Resource {0}	shown while initializing connection pool.
TIBCO-BW-SR-SNOWFLAKE_ DATABASE_ CONNECTION-200009	Debug	BW-Plug-in	Connection pool destroyed for Shared Resource {0}	The message is shown while destroying connection pool.
TIBCO-BW-SR-SNOWFLAKE_ DATABASE_ CONNECTION-200010	Debug	BW-Plug-in	Connection pool parameters : {0}	The message is shown to display connection pool parameters.
TIBCO-BW-PALETTE-SNOWFLAKE_ DATABASE_ JDBC-400001	Warn	BW-Plug-in	No input received for activity : {0}	The message is shown for Snowflake database Insert activity and Snowflake database Update activity if no input is received for values node.
TIBCO-BW-SR-SNOWFLAKE_ DATABASE_ CONNECTION-400001	Warn	BW-Plug-in	Number of thread is invalid, default to 8	The message is shown when number of thread entered is less than 1.
TIBCO-BW-SR-SNOWFLAKE_ DATABASE_ CONNECTION-400002	Warn	BW-Plug-in	Invalid minimum pool size entered. Value must be greater than or equal to 0, value was {0}, defaulted to 0	The message is shown when minimum pool size entered is less than 0.
TIBCO-BW-SR-SNOWFLAKE_	Warn	BW-Plug-in	Invalid maximum pool size entered. Value must	The message is shown when

Error Code	Role	Category	Messages	Description
DATABASE_ CONNECTION-400003			be greater than 0, value was {0}, defaulted to {1}	maximum pool size is less than 1.
TIBCO-BW-SR-SNOWFLAKE_ DATABASE_ CONNECTION-400004	Warn	BW-Plug-in	Minimum pool size: {0} should be less than or equal to Maximum pool size: {1}. Setting Maximum pool size to {2}	The message is shown when minimum pool size is greater than maximum pool size.

TIBCO Documentation and Support Services

For information about this product, you can read the documentation, contact TIBCO Support, and join TIBCO Community.

How to Access TIBCO Documentation

Documentation for TIBCO products is available on the Product Documentation website, mainly in HTML and PDF formats.

The Product Documentation website is updated frequently and is more current than any other documentation included with the product.

Product-Specific Documentation

The documentation for this product is available on the TIBCO ActiveMatrix BusinessWorks™ Plug-in for Snowflake Product Documentation page.

How to Contact Support for TIBCO Products

You can contact the Support team in the following ways:

- To access the Support Knowledge Base and getting personalized content about products you are interested in, visit our product Support website.
- To create a Support case, you must have a valid maintenance or support contract with a Cloud Software Group entity. You also need a username and password to log in to the product Support website. If you do not have a username, you can request one by clicking **Register** on the website.

How to Join TIBCO Community

TIBCO Community is the official channel for TIBCO customers, partners, and employee subject matter experts to share and access their collective experience. TIBCO Community offers access to Q&A forums, product wikis, and best practices. It also offers access to extensions, adapters, solution accelerators, and tools that extend and enable customers to gain full value from TIBCO products. In addition, users can submit and vote on feature

requests from within the TIBCO Ideas Portal. For a free registration, go to TIBCO Community.

Legal and Third-Party Notices

SOME CLOUD SOFTWARE GROUP, INC. (“CLOUD SG”) SOFTWARE AND CLOUD SERVICES EMBED, BUNDLE, OR OTHERWISE INCLUDE OTHER SOFTWARE, INCLUDING OTHER CLOUD SG SOFTWARE (COLLECTIVELY, “INCLUDED SOFTWARE”). USE OF INCLUDED SOFTWARE IS SOLELY TO ENABLE THE FUNCTIONALITY (OR PROVIDE LIMITED ADD-ON FUNCTIONALITY) OF THE LICENSED CLOUD SG SOFTWARE AND/OR CLOUD SERVICES. THE INCLUDED SOFTWARE IS NOT LICENSED TO BE USED OR ACCESSED BY ANY OTHER CLOUD SG SOFTWARE AND/OR CLOUD SERVICES OR FOR ANY OTHER PURPOSE.

USE OF CLOUD SG SOFTWARE AND CLOUD SERVICES IS SUBJECT TO THE TERMS AND CONDITIONS OF AN AGREEMENT FOUND IN EITHER A SEPARATELY EXECUTED AGREEMENT, OR, IF THERE IS NO SUCH SEPARATE AGREEMENT, THE CLICKWRAP END USER AGREEMENT WHICH IS DISPLAYED WHEN ACCESSING, DOWNLOADING, OR INSTALLING THE SOFTWARE OR CLOUD SERVICES (AND WHICH IS DUPLICATED IN THE LICENSE FILE) OR IF THERE IS NO SUCH LICENSE AGREEMENT OR CLICKWRAP END USER AGREEMENT, THE LICENSE(S) LOCATED IN THE “LICENSE” FILE(S) OF THE SOFTWARE. USE OF THIS DOCUMENT IS SUBJECT TO THOSE SAME TERMS AND CONDITIONS, AND YOUR USE HEREOF SHALL CONSTITUTE ACCEPTANCE OF AND AN AGREEMENT TO BE BOUND BY THE SAME.

TIBCO, the TIBCO logo, the TIBCO O logo, ActiveMatrix BusinessWorks, Business Studio, and TIBCO Business Studio are either registered trademarks or trademarks of Cloud Software Group, Inc. in the United States and/or other countries.

All other product and company names and marks mentioned in this document are the property of their respective owners and are mentioned for identification purposes only. You acknowledge that all rights to these third party marks are the exclusive property of their respective owners. Please refer to Cloud SG’s Third Party Trademark Notices (https://www.cloud.com/legal) for more information.

This document includes fonts that are licensed under the SIL Open Font License, Version 1.1, which is available at: https://scripts.sil.org/OFL

Copyright (c) Paul D. Hunt, with Reserved Font Name Source Sans Pro and Source Code Pro.

Cloud SG software may be available on multiple operating systems. However, not all operating system platforms for a specific software version are released at the same time. See the “readme” file for the availability of a specific version of Cloud SG software on a specific operating system platform.