Configuring iWay Service Manager for Bi-directional Communication With Salesforce.com

This topic describes how to configure communication between Salesforce.com and iWay Service Manager (iSM). An example is provided that shows data being propagated from Salesforce.com to Oracle and from Oracle to Salesforce.com.

This how-to includes the following topics:

Prerequisites

Before continuing, ensure that your environment supports the following prerequisites:

- iWay Integration Tools (iIT) version 8.0.1 or higher (for design time).
- iWay Service Manager (iSM) version 8.0.1 or higher (for run time and deployment)
- Oracle JDBC driver (for example, *ojdbc7.jar*)
- Oracle database access
- Salesforce.com login account
- IWAYHOME set in the iIT workspace

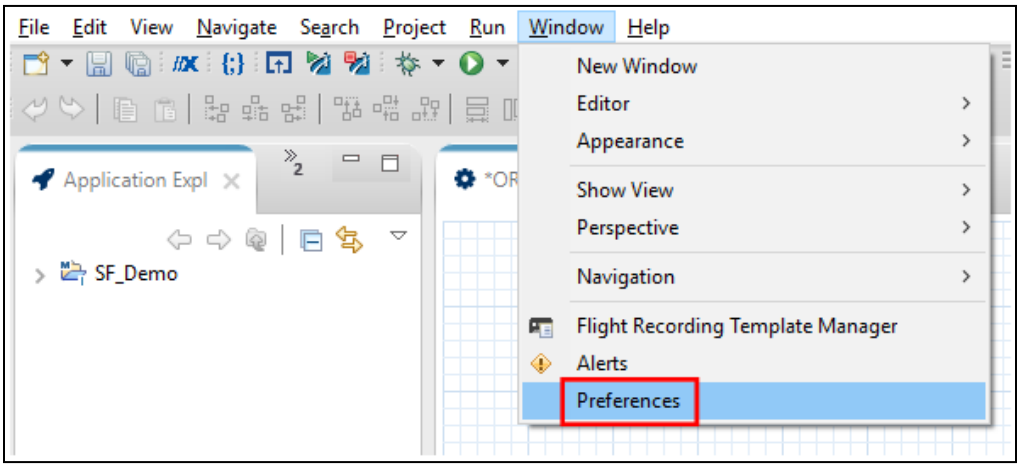
Setting the iWay Home Directory in the iWay Integration Tools Workspace

When creating a new workspace in iWay Integration Tools (iIT), you must define the *iWay Home Directory*. The iWay Home Directory is the local installation directory of your iWay Service Manager (iSM) instance or a copy of this directory if a local installation is not available.

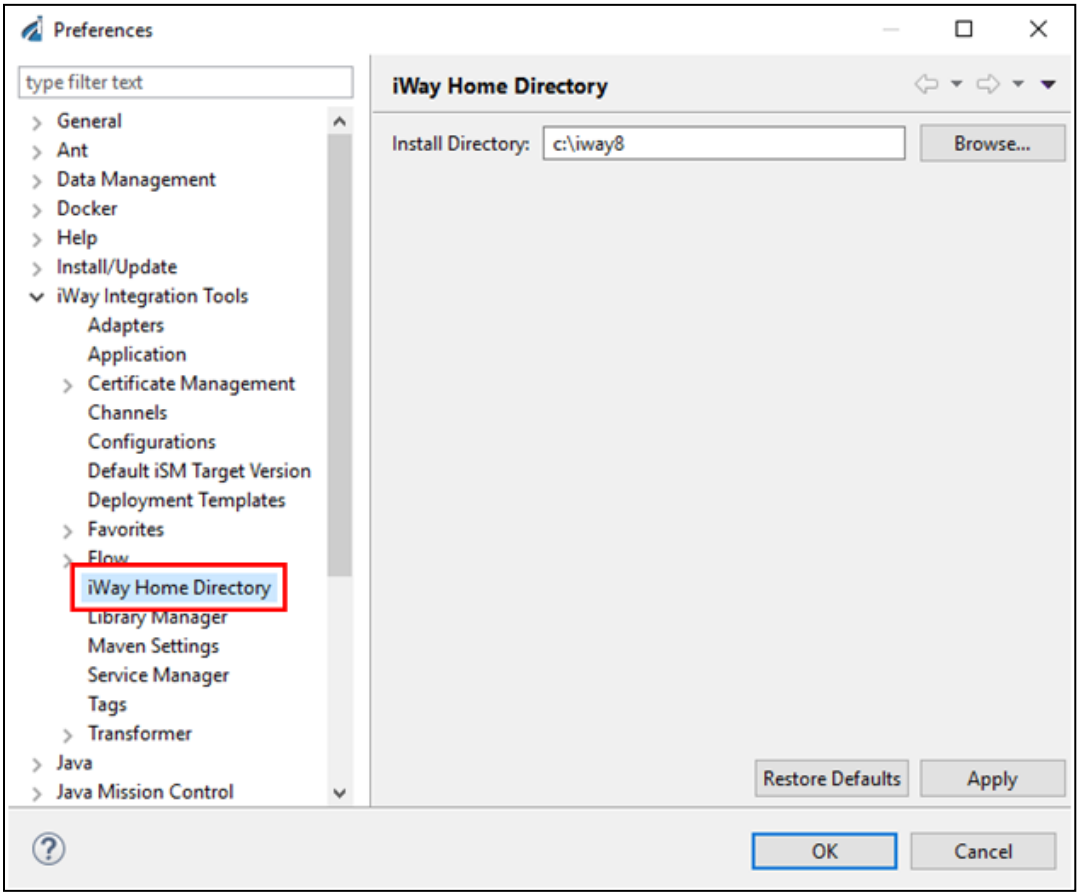
To set the iWay Home Directory in the iIT workspace:

1. Open iWay Integration Tools (iIT).

2. Click *Window* in the menu bar and select *Preferences* from the context menu, as shown in the following image.



The Preferences dialog opens, as shown in the following image.



3. Expand *iWay Integration Tools* and *iWay Home Directory* in the left pane.

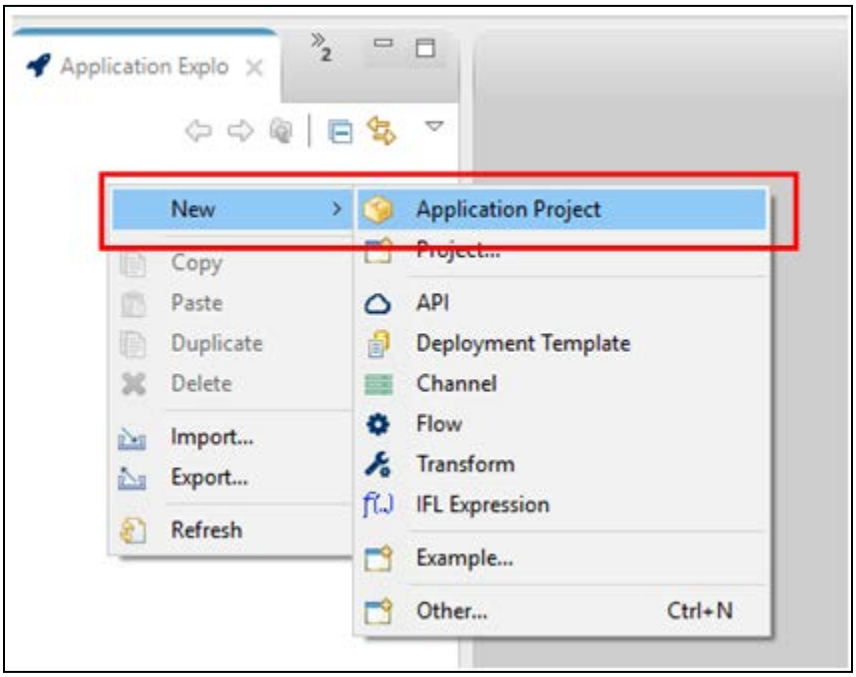
4. In the Install Directory field, specify or browse to the path of your iSM installation directory (for example, *c:\iway8*).

5. Click *Apply* and then *OK*.

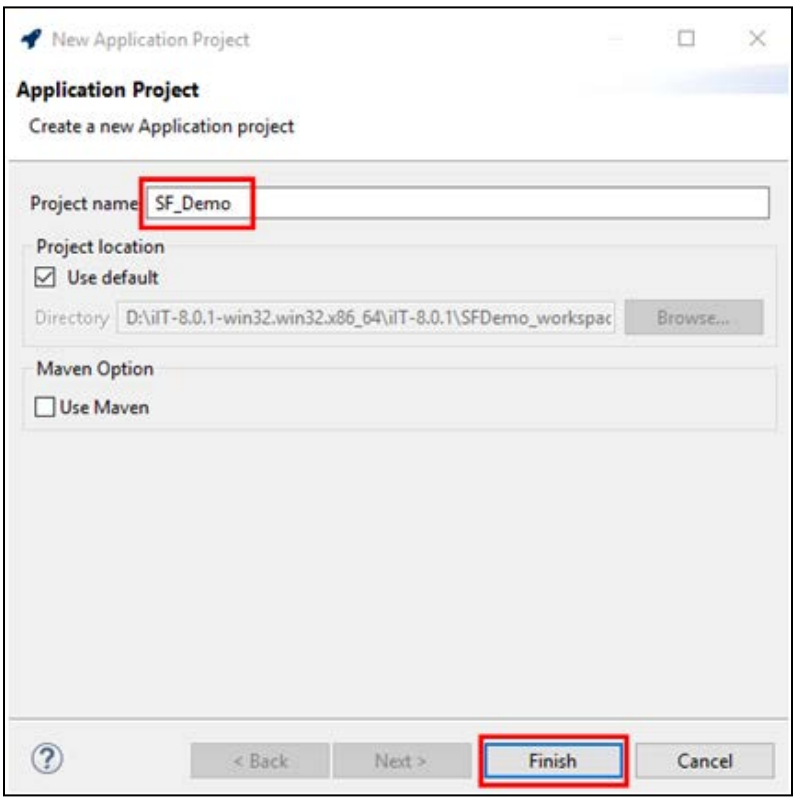
Configuring a Connection From Oracle to Salesforce.com

To configure a connection from Oracle to Salesforce.com:

1. Right-click anywhere in the Application Explorer tab, select *New* from the context menu, and then click *Application Project*, as shown in the following image.



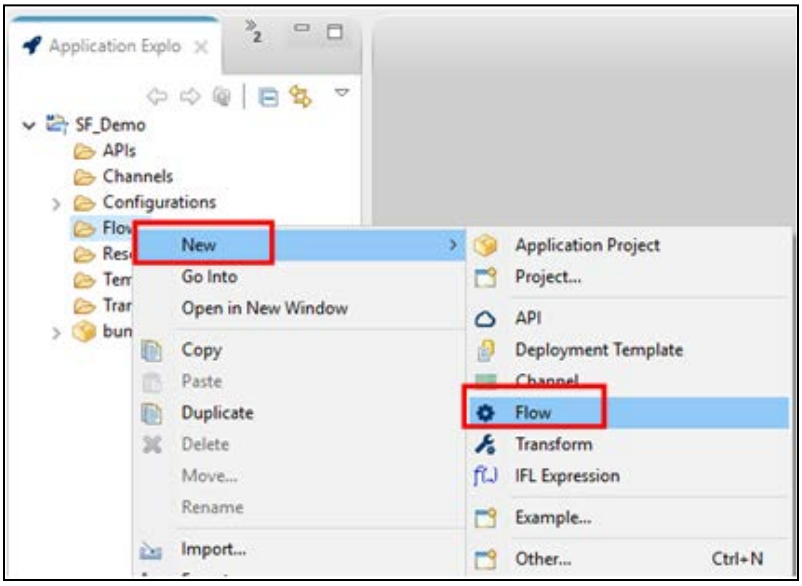
The New Application Project dialog opens, as shown in the following image.



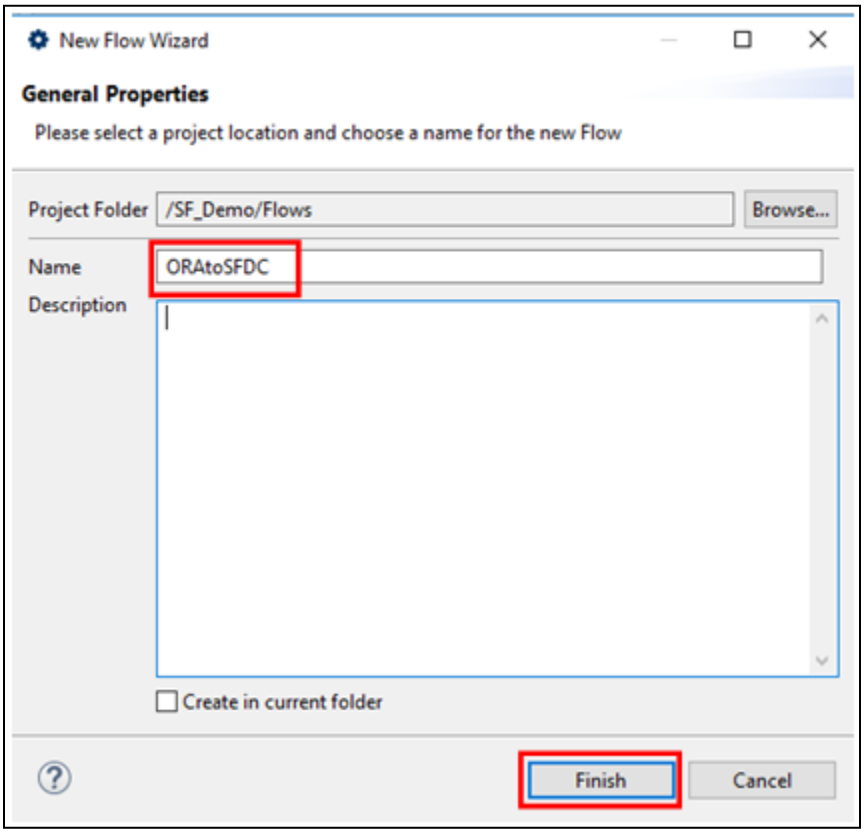
2. Type a name for your application project in the Project name field (for example, *SF_Demo*) and click *Finish*.

   The new application project is created and listed in the Application Explorer tab.

3. Right-click the *Flows* subfolder, select *New* from the context menu, and then click *Flow*, as shown in the following image.

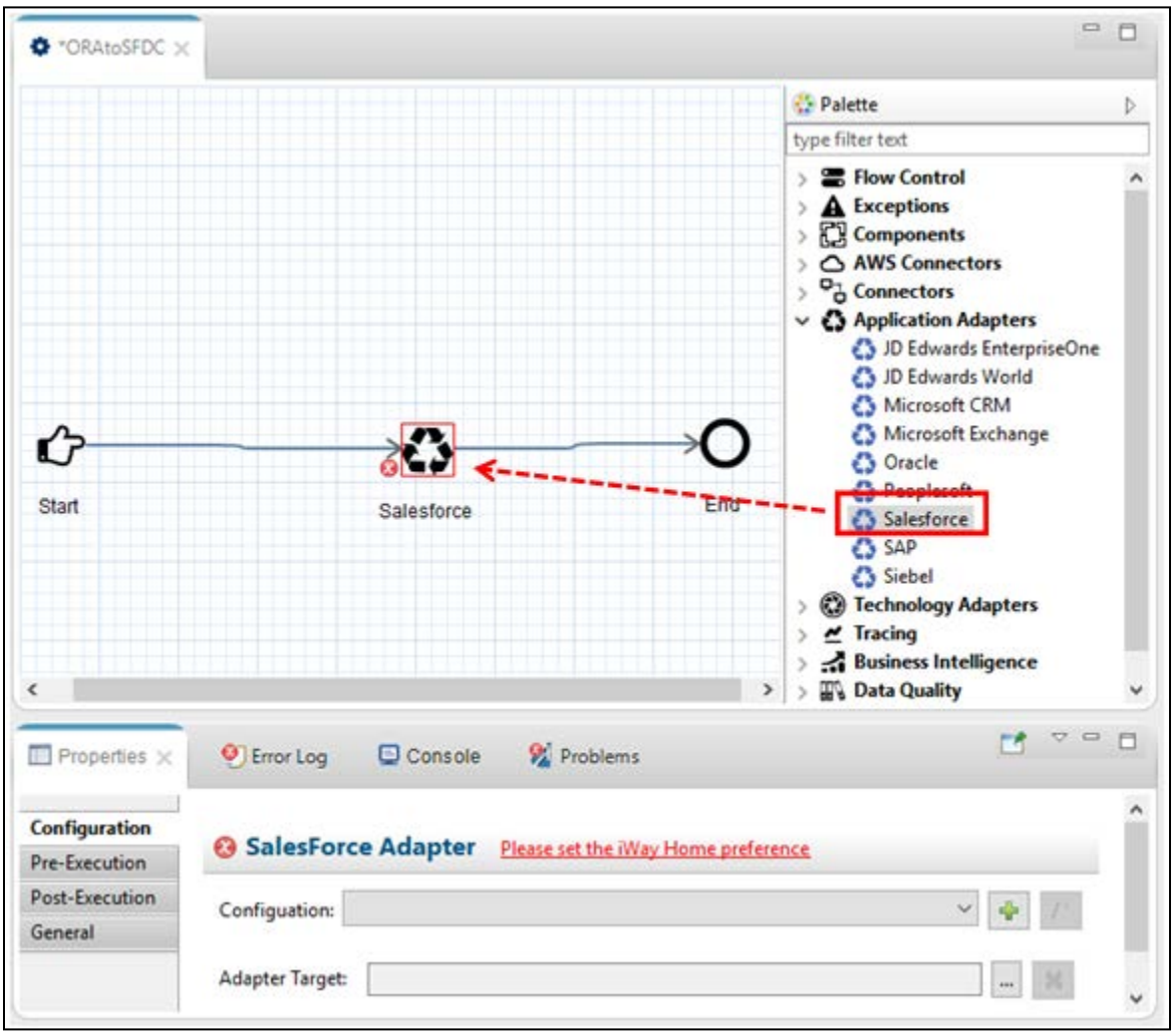


The New Flow Wizard dialog opens, as shown in the following image.



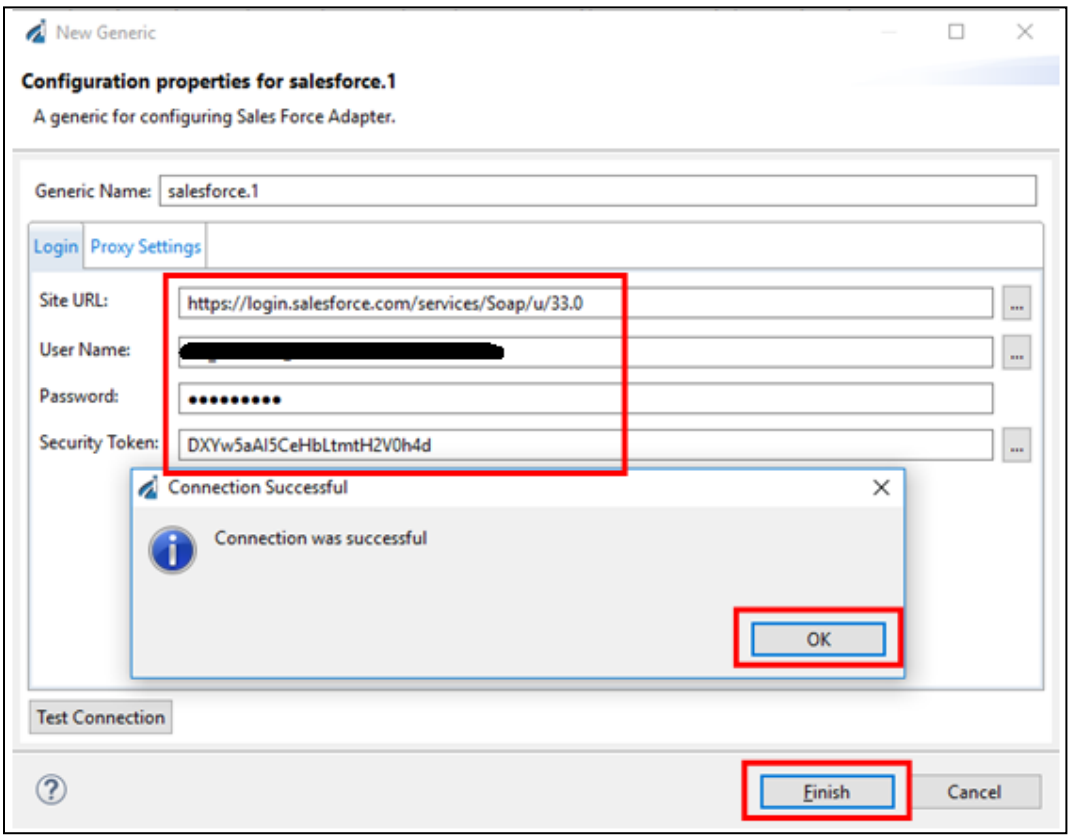
4. Type a name for your process flow in the Name field (for example, *ORAtoSFDC*) and click *Finish.*

5. From the Palette, which is located in the right pane, expand *Application Adapters*. Click and drag the *Salesforce* object to the workspace area on the line between the *Start* and *End* objects, as shown in the following image.



6. In the Properties tab on the bottom of the screen, click the green plus sign (+) to the right of the Configuration field.

   The New Generic dialog opens, as shown in the following image.



7. Perform the following steps:

   a) Specify a name for the new generic in the Generic Name field, or accept the default, which is *salesforce.1*.

   b) Specify values for the following parameters that are specific to the Salesforce.com instance to which you are connecting:

      i. Site URL

      ii. User Name

      iii. Password

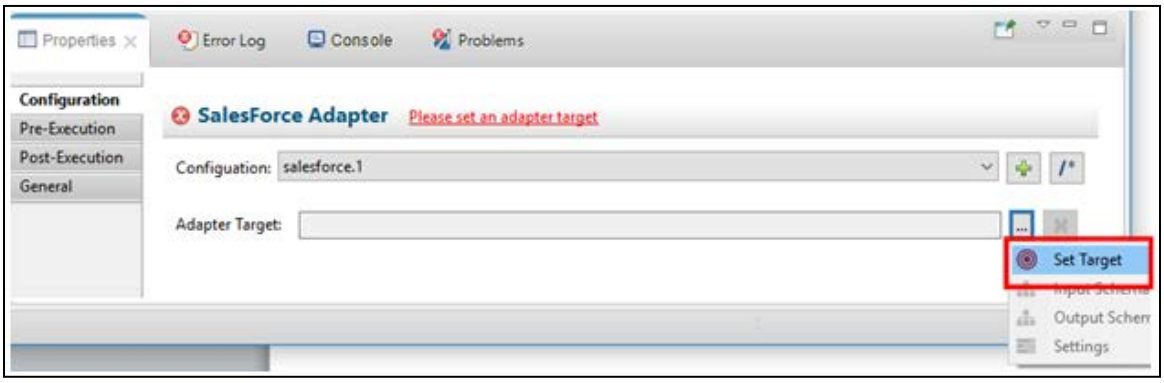
      iv. Security Token

   c) Click *Test Connection* to confirm and validate your connection to Salesforce.com.

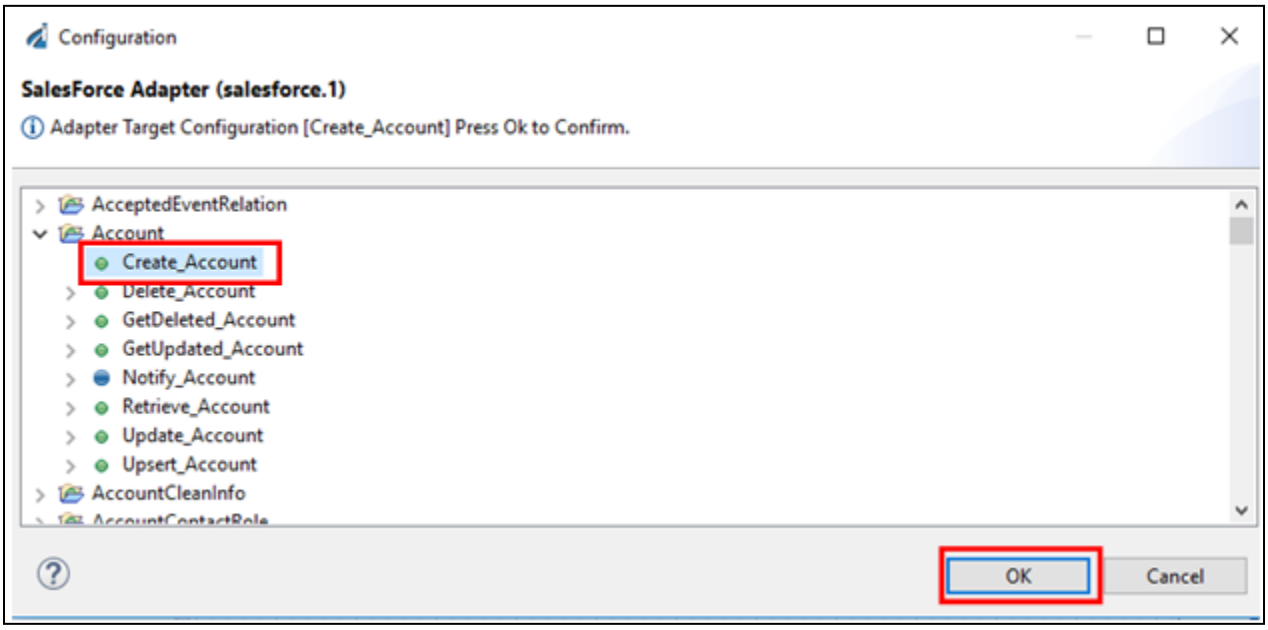
   d) If the connection test is successful, click *OK*, and then click *Finish*.

      You are returned to the Properties tab on the bottom of the screen.

8. Click the ellipsis icon (…) to the right of the Adapter Target field and select *Set Target* from the context menu, as shown in the following image.



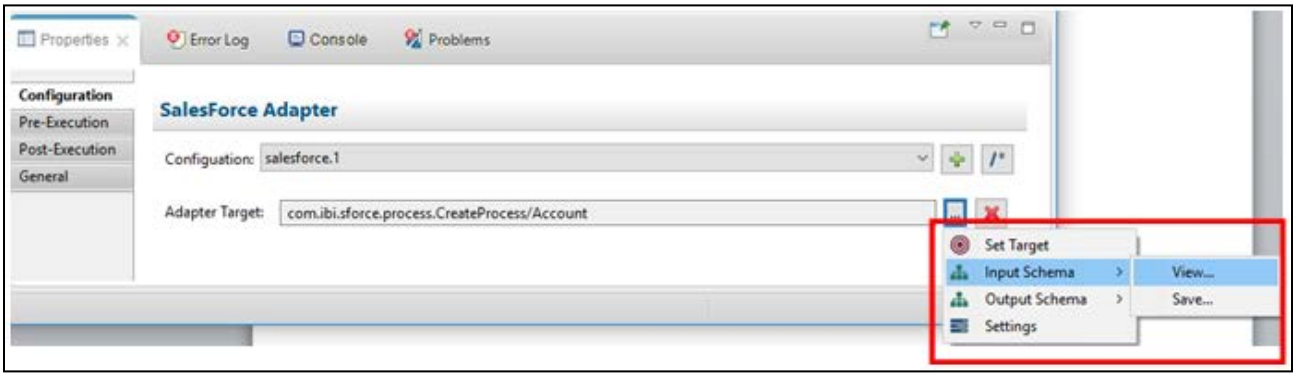
The Configuration dialog opens, as shown in the following image.



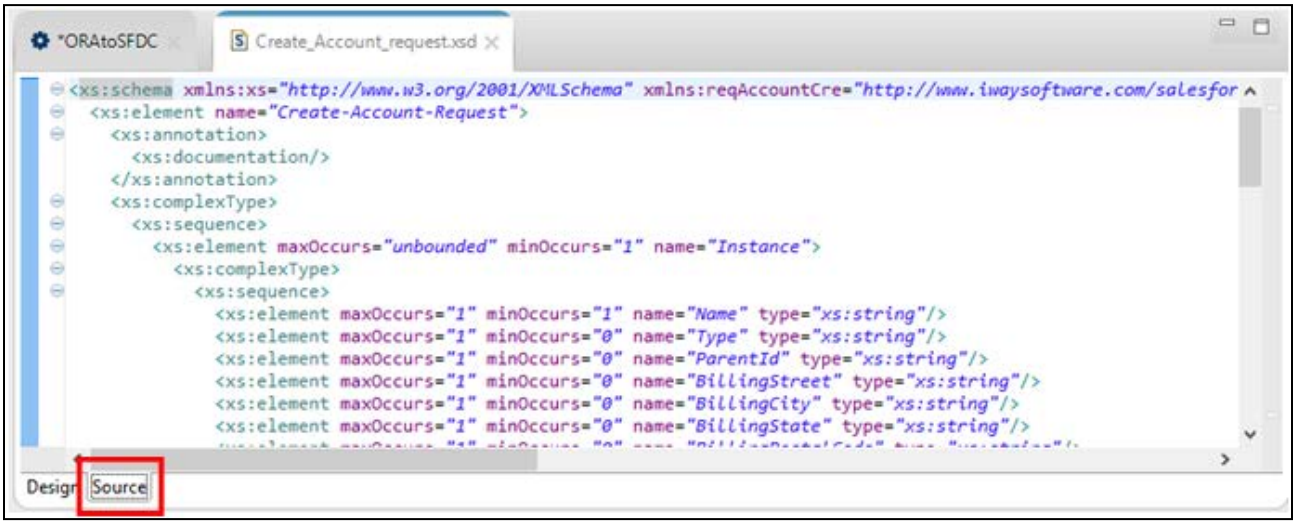
9. Expand the *Account* folder node, select *Create_Account*, and then click *OK*.

10. To view or save an XML input schema, click the ellipsis icon (…) to the right of the Adapter Target field, select *Input Schema*, and then either *View* or *Save* from the context menu.

    In the following image and example, *View* is selected from the context menu.



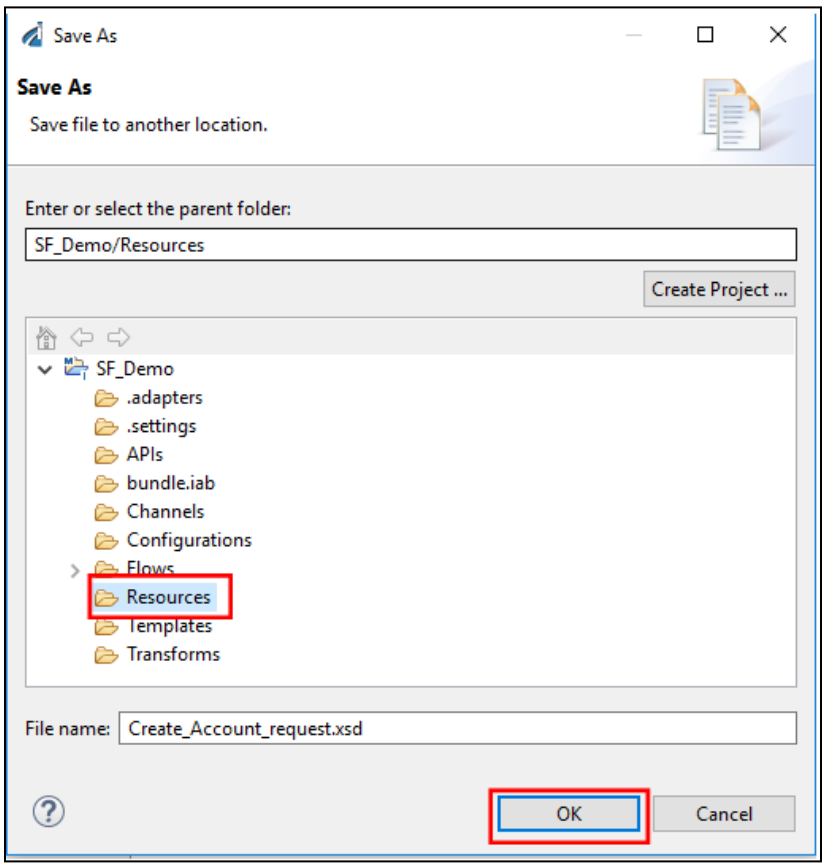
The XML input schema document opens as a new tab in your workspace area, as shown in the following image.



11. Click the *Source* sub-tab to view the structure of the XML input schema.

12. Repeat step 10, but select *Input Schema* and click *Save* from the context menu to save the XML input schema.

    The Save As dialog opens, as shown in the following image.



13. Expand your application project, select the *Resources* folder, and then click *OK*.

    Note: An XML schema shows the fields that are required to update data properly in Salesforce.com. Data from any source can now be used by using a Transform to build a canonical output that matches this XML schema.

Building a Canonical Transform

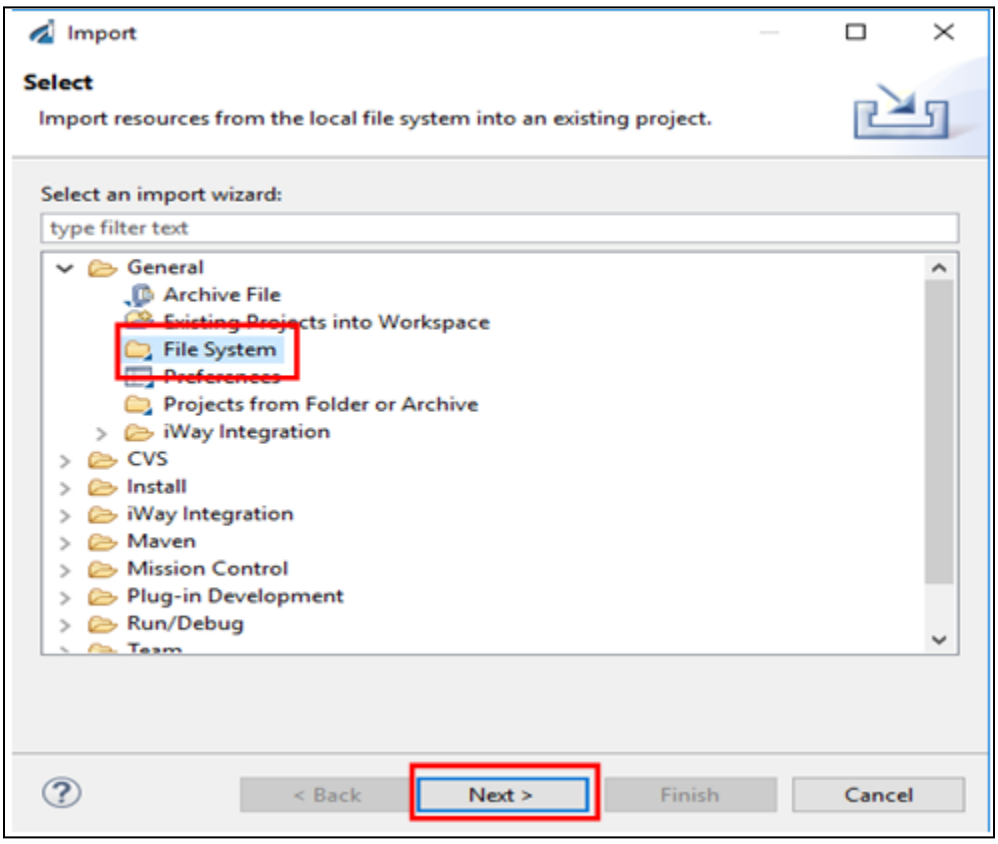
To build a canonical transform:

1. Import an input and output structure (XML schema).

   Note: In the previous procedure, you already saved the output structure (*Create_Account_request.xsd*) in the Resources folder of your application project.

2. Import an input structure by right-clicking the *Resources* folder and selecting *Import* from the context menu.

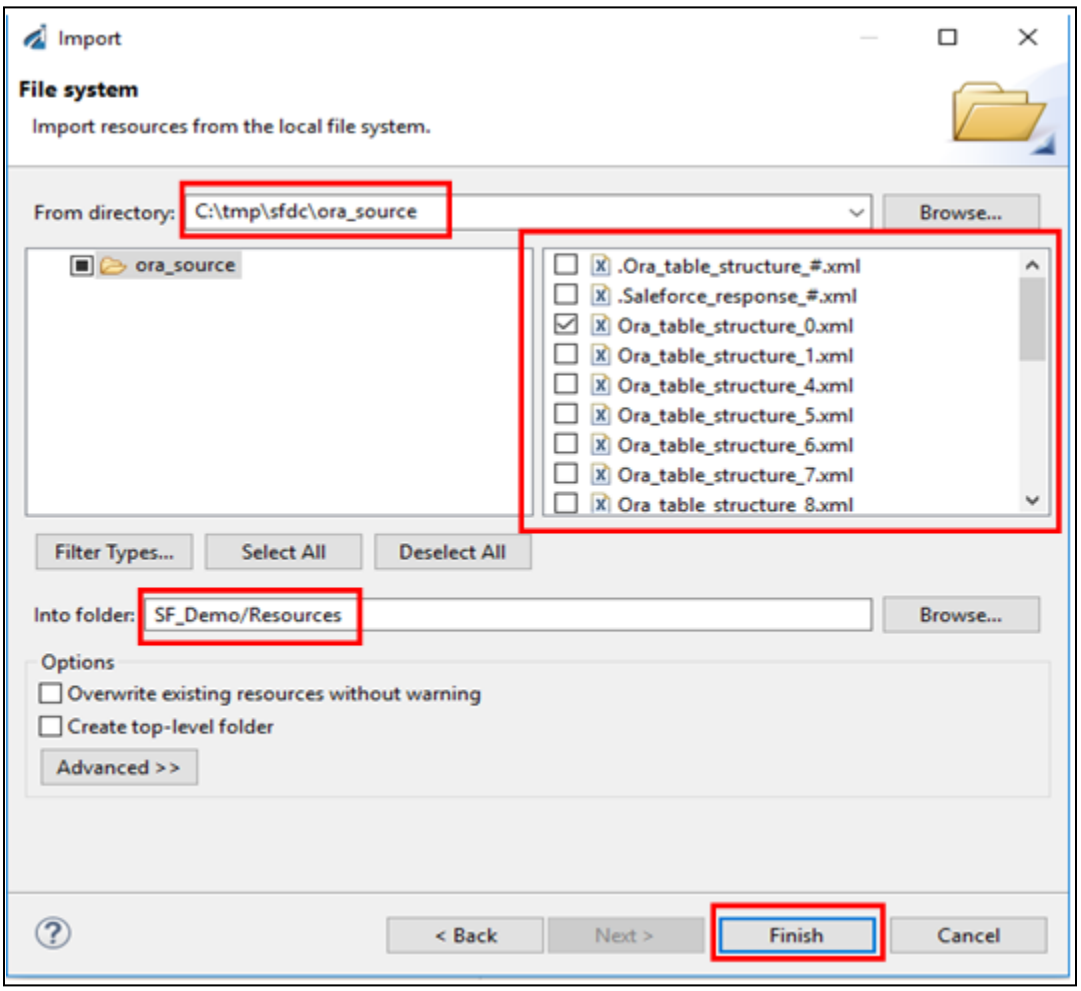
   The Import dialog opens, as shown in the following image.



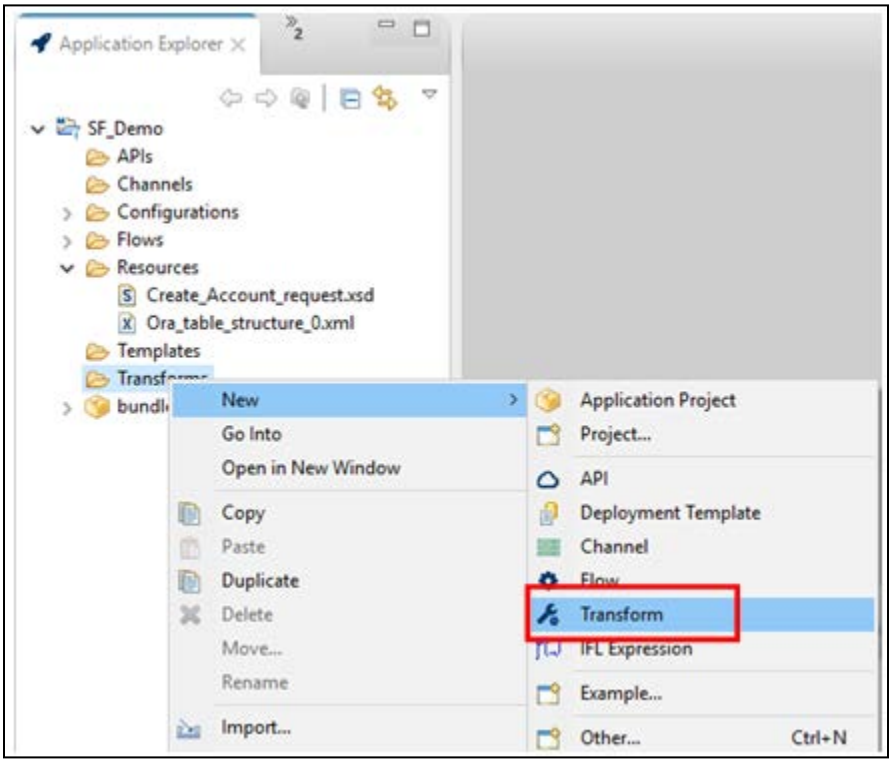
3. Expand *General*, select *File System*, and then click *Next*.

4. Browse to an XML file on your file system that represents the expected input structure.

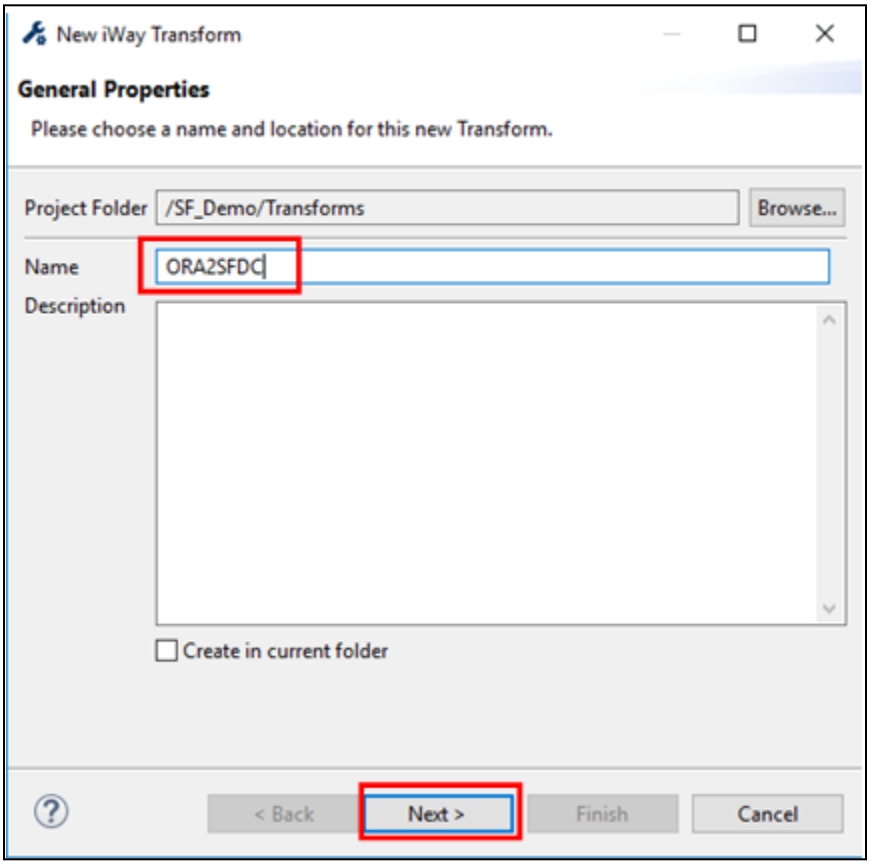
5. Select the file and click *Finish*, as shown in the following image.



6. Right-click the *Transforms* folder under your application project, select *New* from the context menu, and then click *Transform*, as shown in the following image.

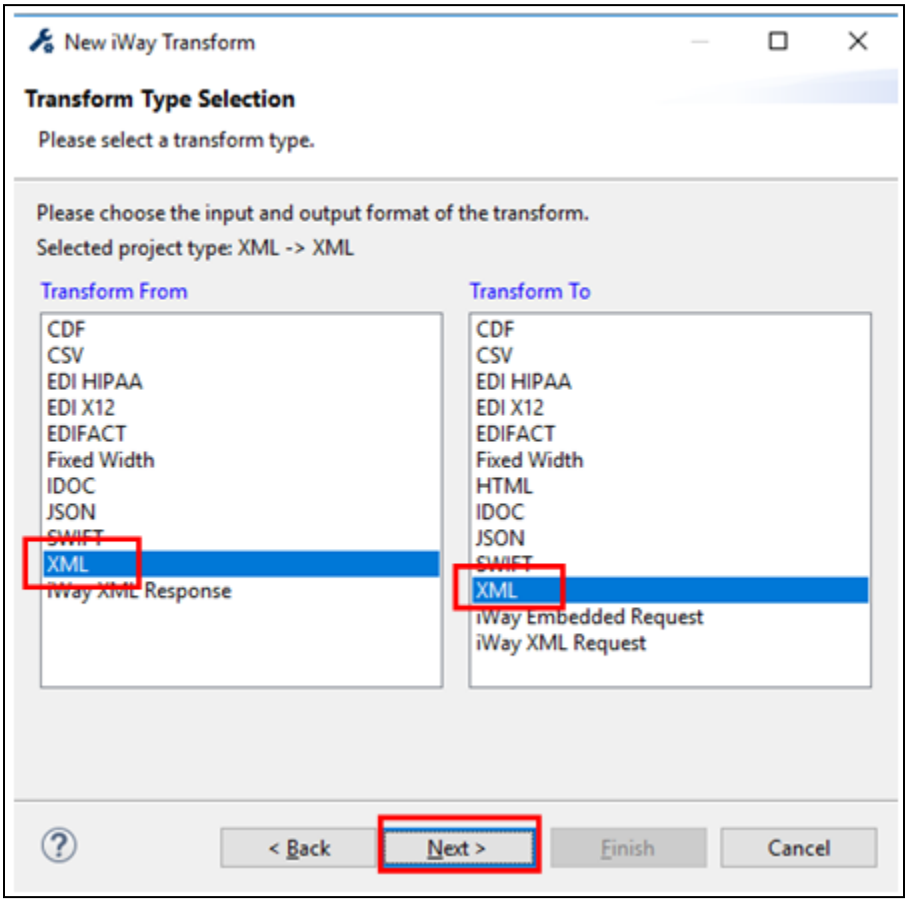


The New iWay Transform dialog opens, as shown in the following image.



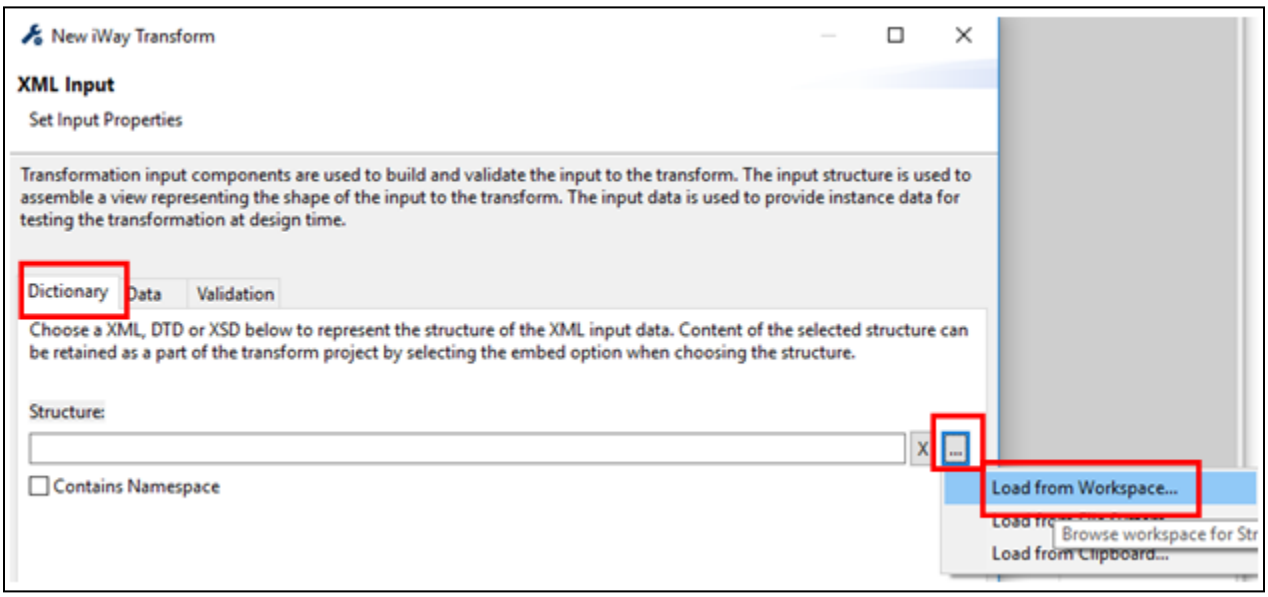
7. Specify a name for your Transform (for example, ORA2SFDC), and click *Next*.

   The Transform Type Selection pane opens, as shown in the following image.

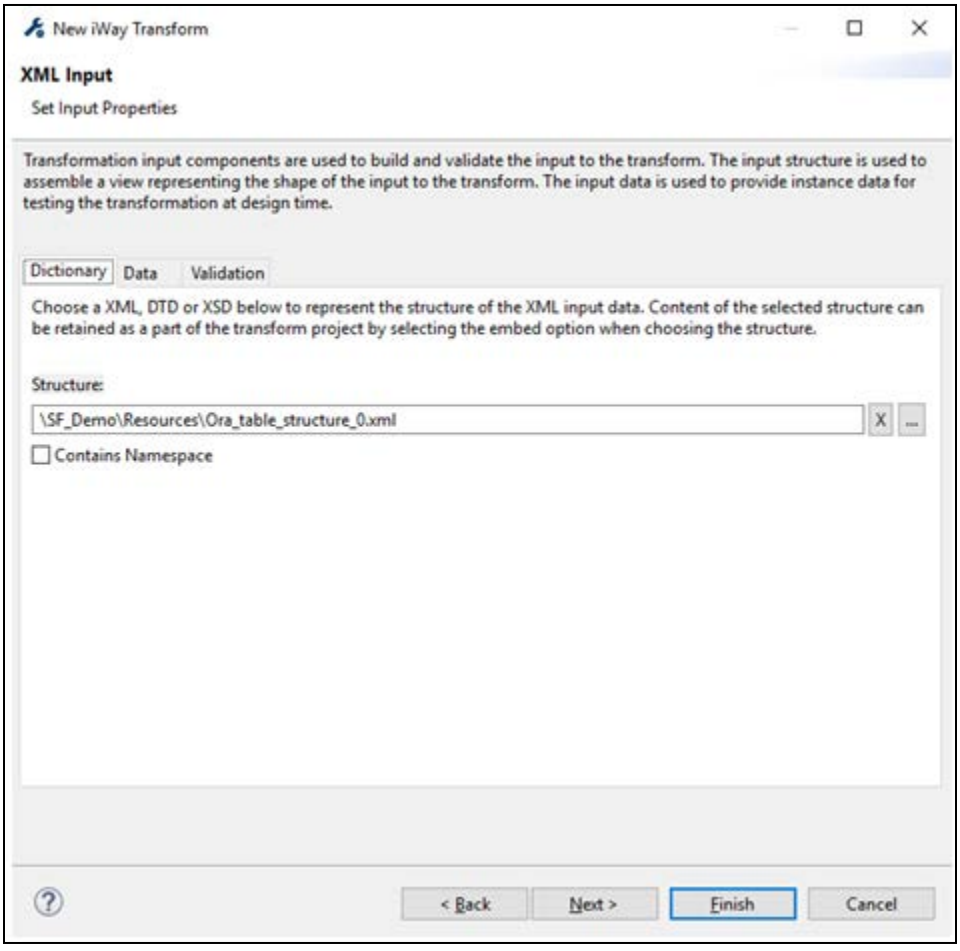


8. Select *XML* in the Transform From and Transform To columns, and then click *Next*.

   The XML Input pane opens, as shown in the following image.

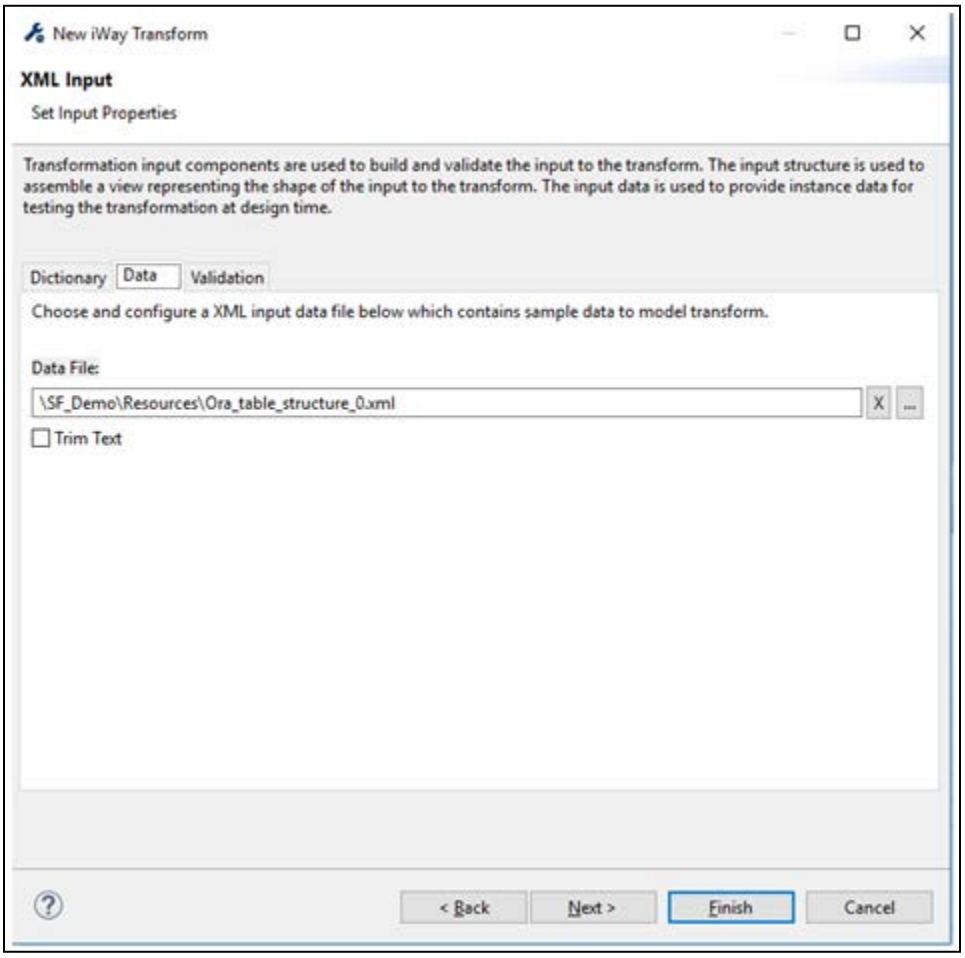


9. Click the ellipsis icon (…) to the right of the Structure field in the Dictionary tab and select *Load from Workspace* from the context menu.



10. Select your input structure.

11. Click the *Data* tab, as shown in the following image.

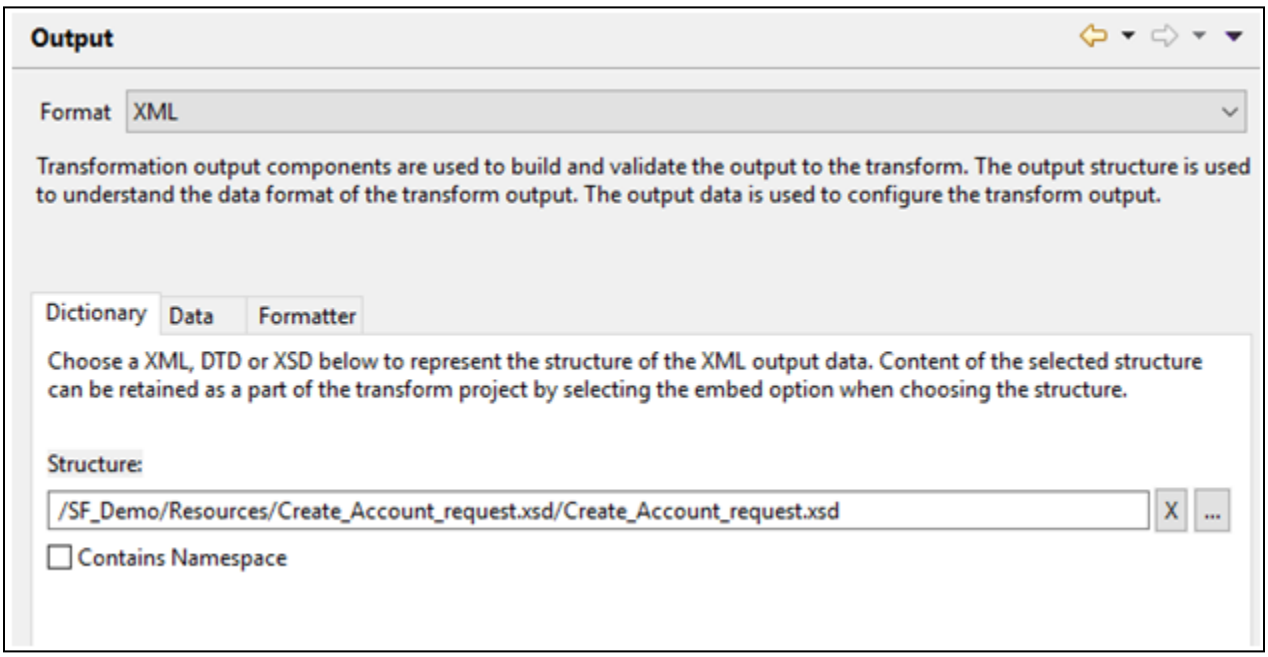


12. Click the ellipsis icon (…) to the right of the Data File field in the Data tab and select *Load from Workspace* from the context menu.

13. Select your input structure.

14. Click *Next*.

    The Output pane opens, as shown in the following image.



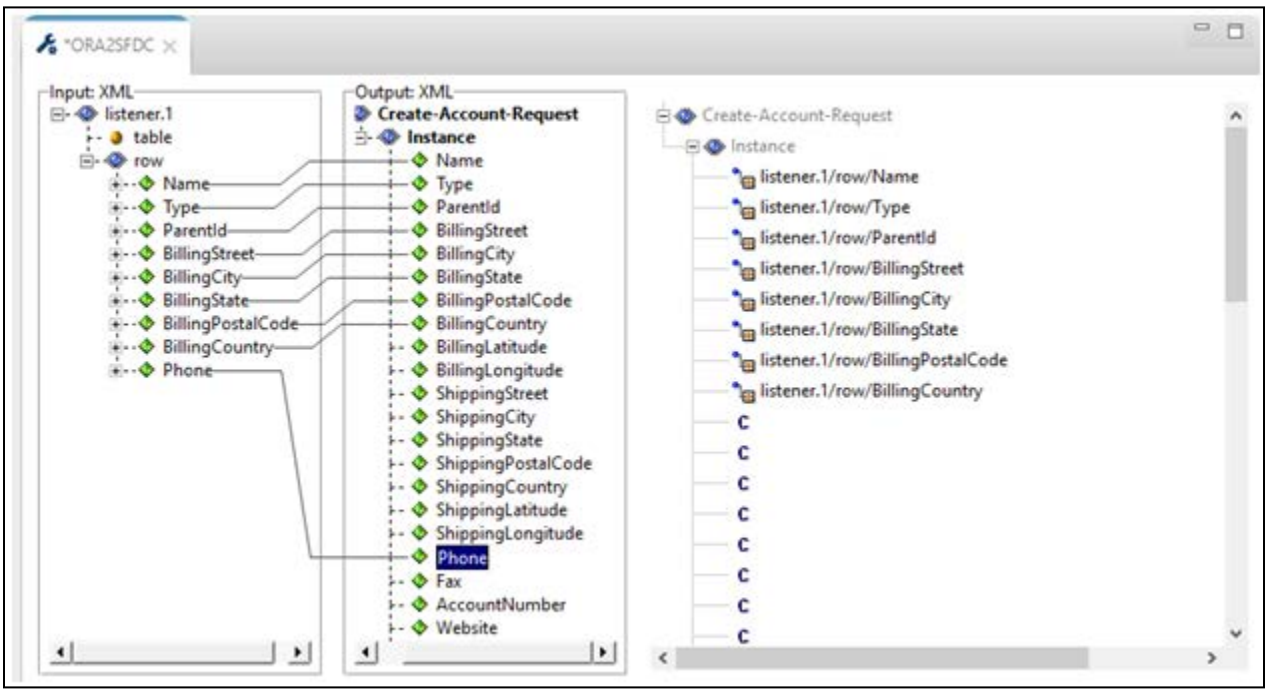
15. Click the ellipsis icon (…) to the right of the Structure field in the Dictionary tab and select *Load from Workspace* from the context menu.

16. Select the output structure (*Create_Account_request.xsd*) that you previously saved in the Resources folder.

17. Click *Finish*.

    The Transform opens as a new tab in your workspace area.

18. Expand the input XML and output XML structures, as shown in the following image.



19. In the left Input pane click and drag *Name* to the right Output pane and release it on top of Name. Repeat this for all of the remaining fields that you wish to propagate.

20. Click *Save* and close the Transform component.

    Note: Transforms can be called based on the input data. As a result, a single process flow could handle many inbound data structures.

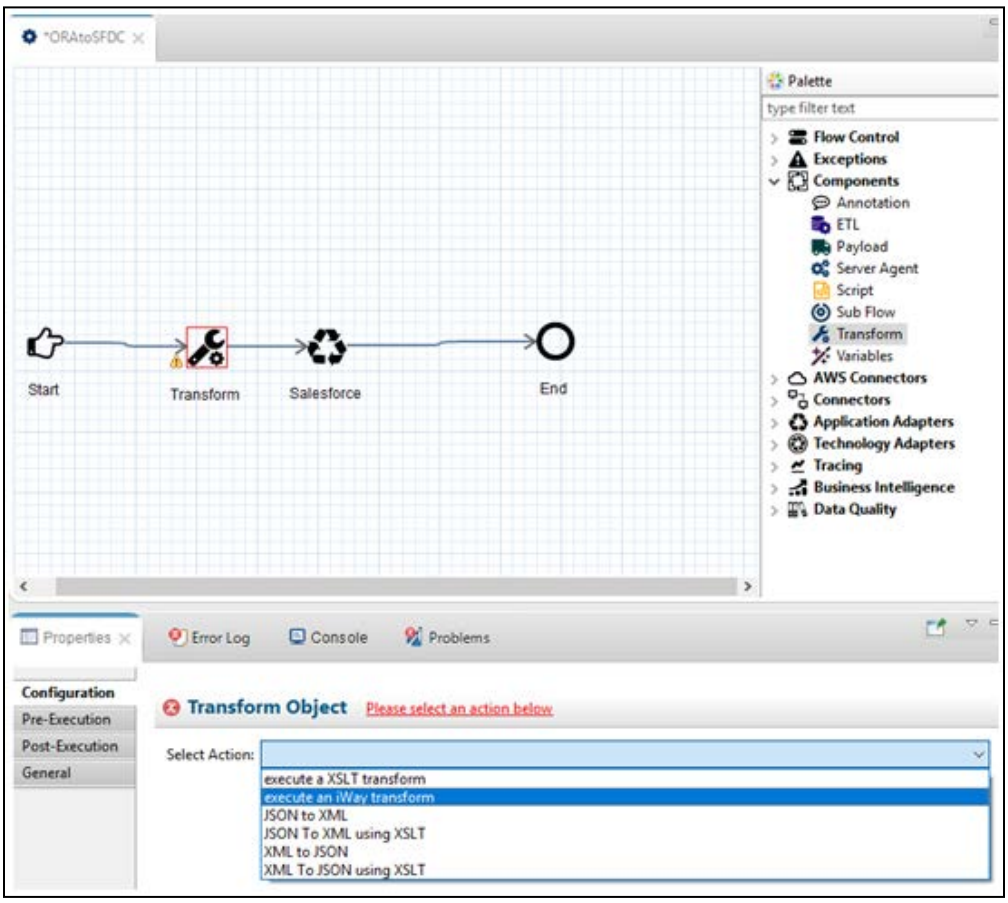
Adding a Transform to a Salesforce.com Process Flow

To add a transform to a Salesforce.com process flow:

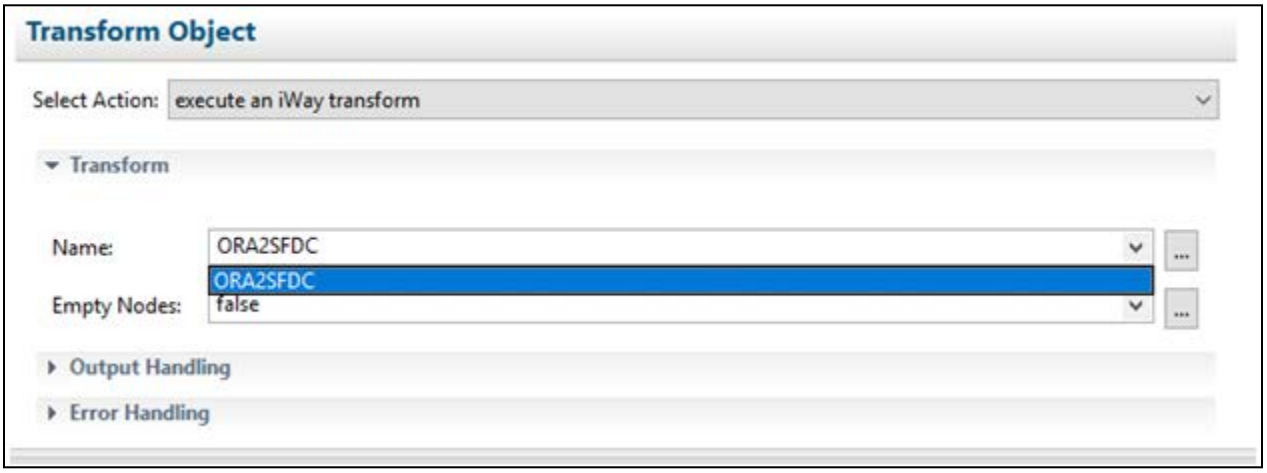
1. Open the process flow that was created earlier (ORAtoSFDC) to configure a connection from Oracle to Salesforce.com.

2. From the Palette, which is located in the right pane, expand *Components*. Click and drag the *Transform* object to the workspace area on the line between the *Start* and *Salesforce* objects.

3. In the Properties tab on the lower part of the screen, select *execute an iWay transform* from the Select Action drop-down list, as shown in the following image.



4. Select *ORA2SFDC* from the Name drop-down list, as shown in the following image.



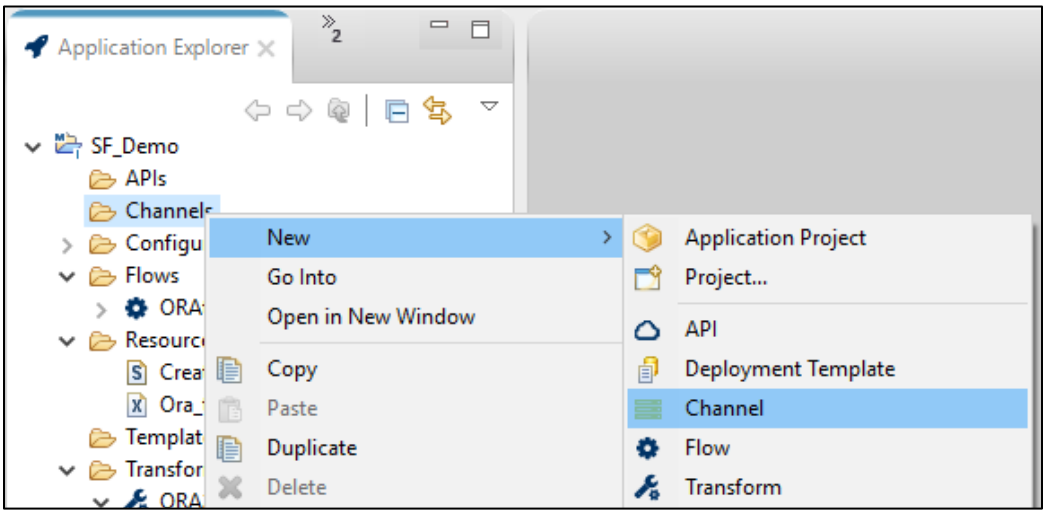
5. Save and close the process flow.

Note: The process flow lines are set to *On Complete*, meaning that the process flow will complete regardless of an error. If you want your process flow to fail on anything other than *On Success*, click the line that follows an object in a process flow to change its behavior.

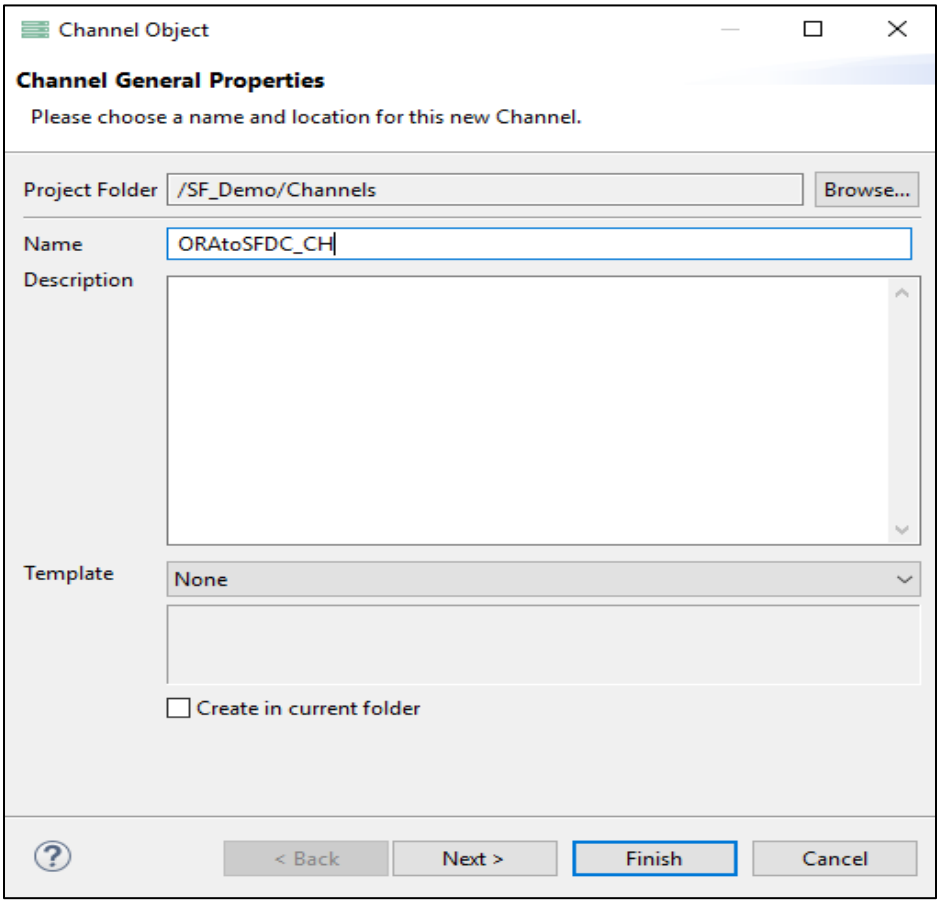
Configuring an RDBMS Channel Listener

To configure an RDBMS channel listener:

1. Expand your application project folder (for example, *SF_Demo*), right-click the *Channels* subfolder, select *New* from the context menu, and then click *Channel*, as shown in the following image.

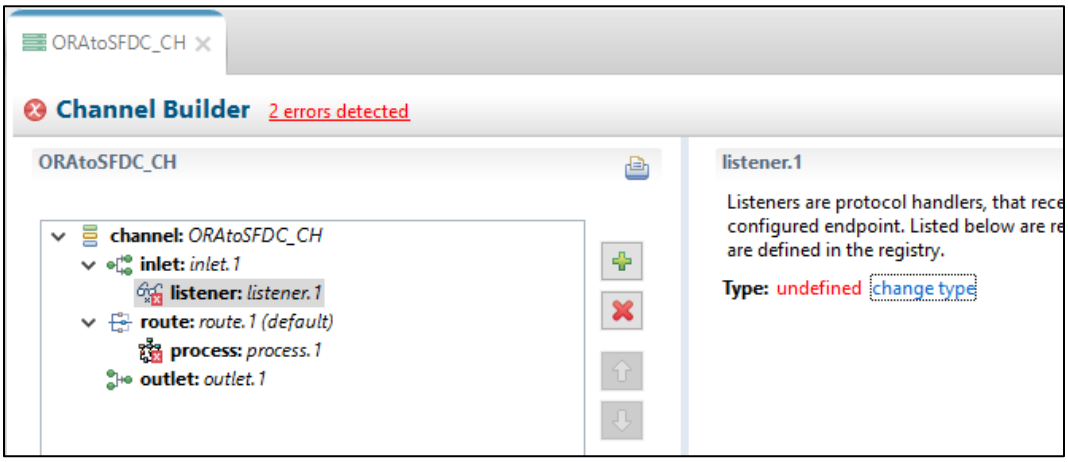


The Channel Object dialog opens, as shown in the following image.



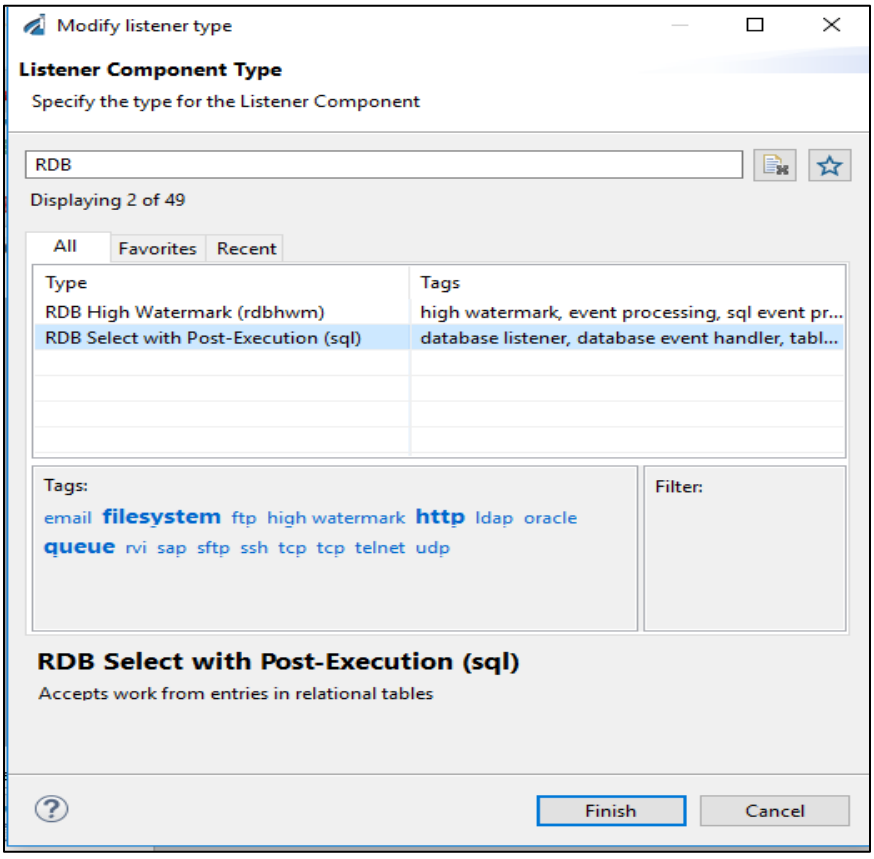
2. Type a name for your channel in the Name field (for example, *ORAtoSFDC_CH*) and click *Finish*.

   The Channel Builder opens as a new tab in your workspace area, as shown in the following image. The name of the tab corresponds to the name you specified for your channel.



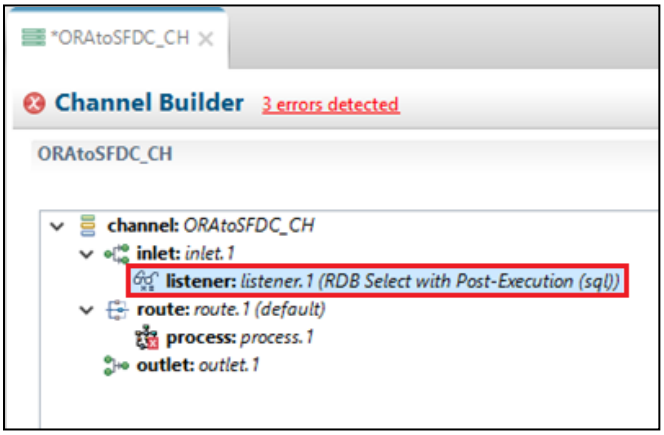
3. In the left pane, expand the *inlet: inlet.1* node, select *listener:listener.1*, and then click *change type* in the right pane.

   The Modify listener type dialog opens, as shown in the following image.



4. Type *RDB* in the listener search/filter field, select *RDB Select with Post-Execution (sql)* from the resulting list, and then click *Finish*.

   The listener type for your inlet is updated in the left pane of the Channel Builder, as shown in the following image.



5. In the right pane, expand the *Main* group and specify the following values for the *SQL Query* and *SQL Post-query* parameters:

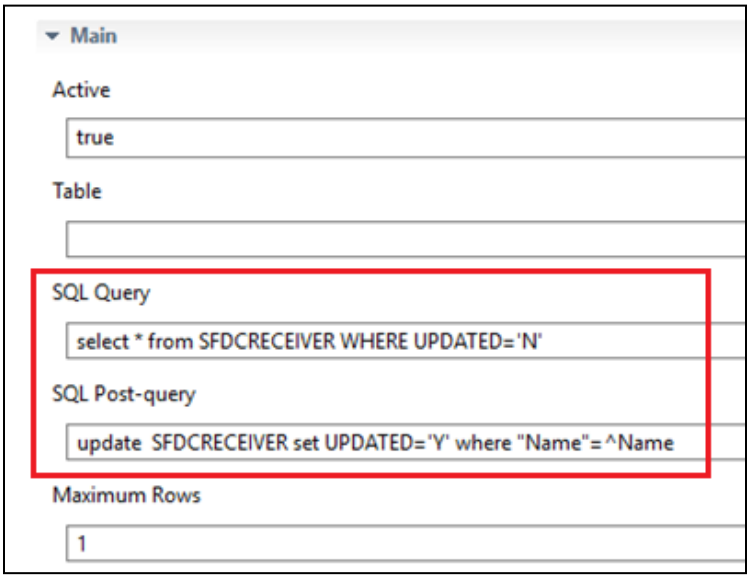
   - SQL Query:

     ```
     select * from SFDCRECEIVER WHERE UPDATED='N'
     ```

   - SQL Post-query:

     ```
     update SFDCRECEIVER set UPDATED='Y' where "Name"=^Name
     ```

   For example:



6. Retain the default values for the remaining parameters in the Main group.

7. In the right pane, expand the *Data Base Access* group and specify the following values for the *Driver* and *URL* parameters:

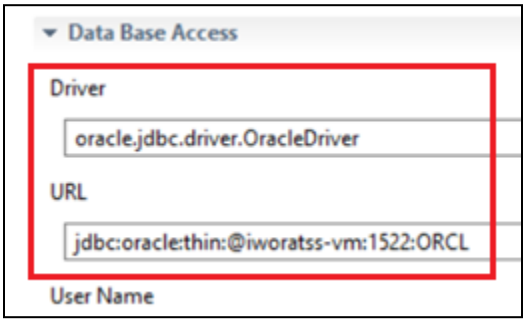
    - Driver:

      ```
      oracle.jdbc.driver.OracleDriver
      ```

    - URL:

      ```
      jdbc:oracle:thin:@iworatss-vm:1522:ORCL
      ```

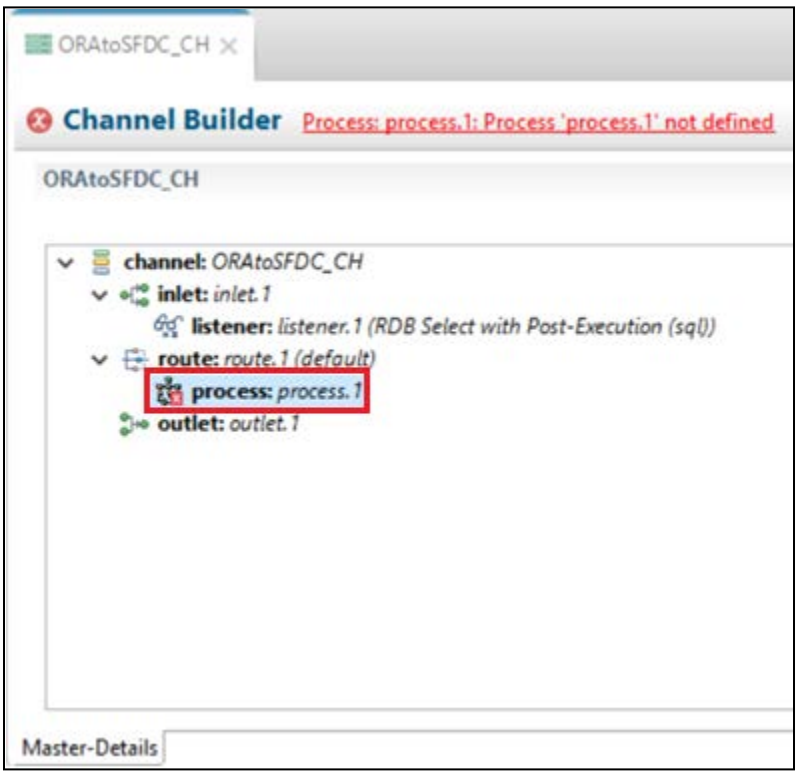
    For example:



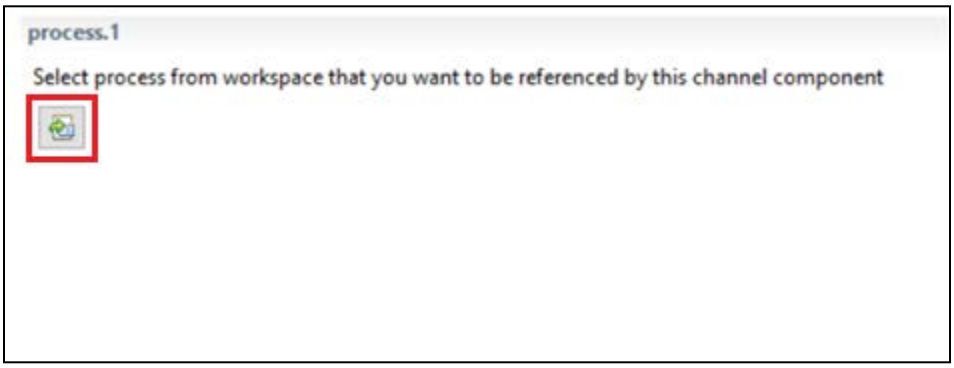
8. Click *Save* (or press *Ctrl+S*).

    You are now ready to associate this channel with the process flow you configured earlier, which configures a connection from Oracle to Salesforce.com.

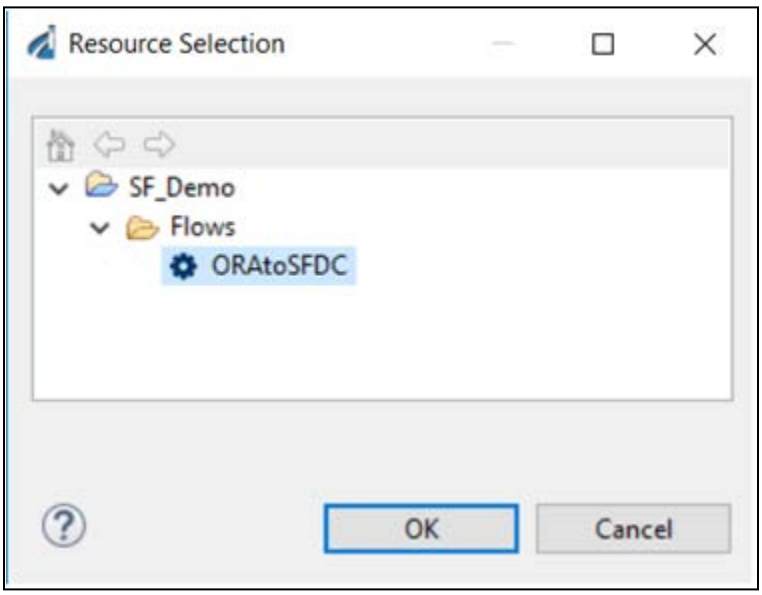
9. In the left pane, expand the *route: route.1 (default)* node and select *process:process.1*, as shown in the following image.



10. In the right pane, click the *Select Process Flow* icon, as shown in the following image.



The Resource Selection dialog opens, as shown in the following image.



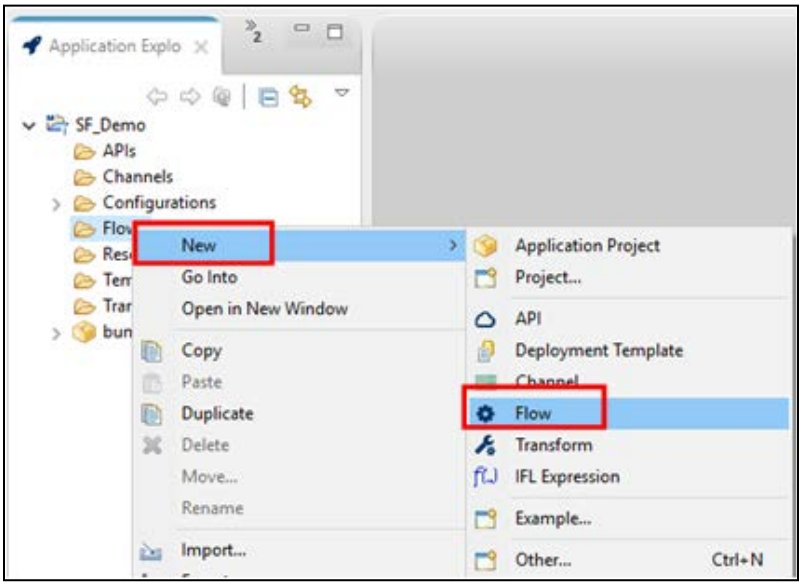
11. Expand the *Flows* subfolder under your application project, select the *ORAtoSFDC* process flow, and then click *OK*.

12. Click *Save* (or press *Ctrl+S*) to save your channel configuration.

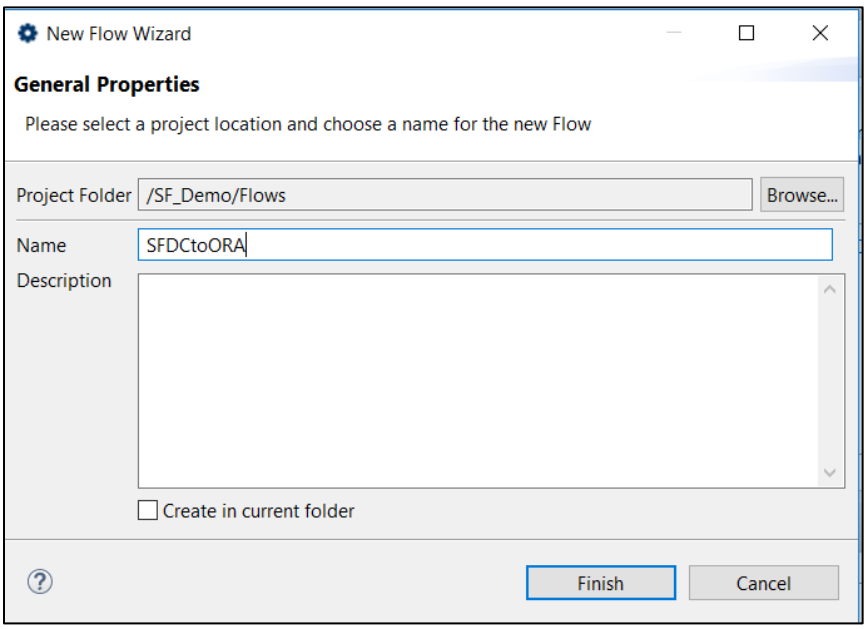
Configuring a Connection From Salesforce.com to Oracle

To configure a connection from Salesforce.com to Oracle:

1. Expand your application project folder (for example, *SF_Demo*), right-click the *Flows* subfolder, select *New* from the context menu, and then click *Flow*, as shown in the following image.



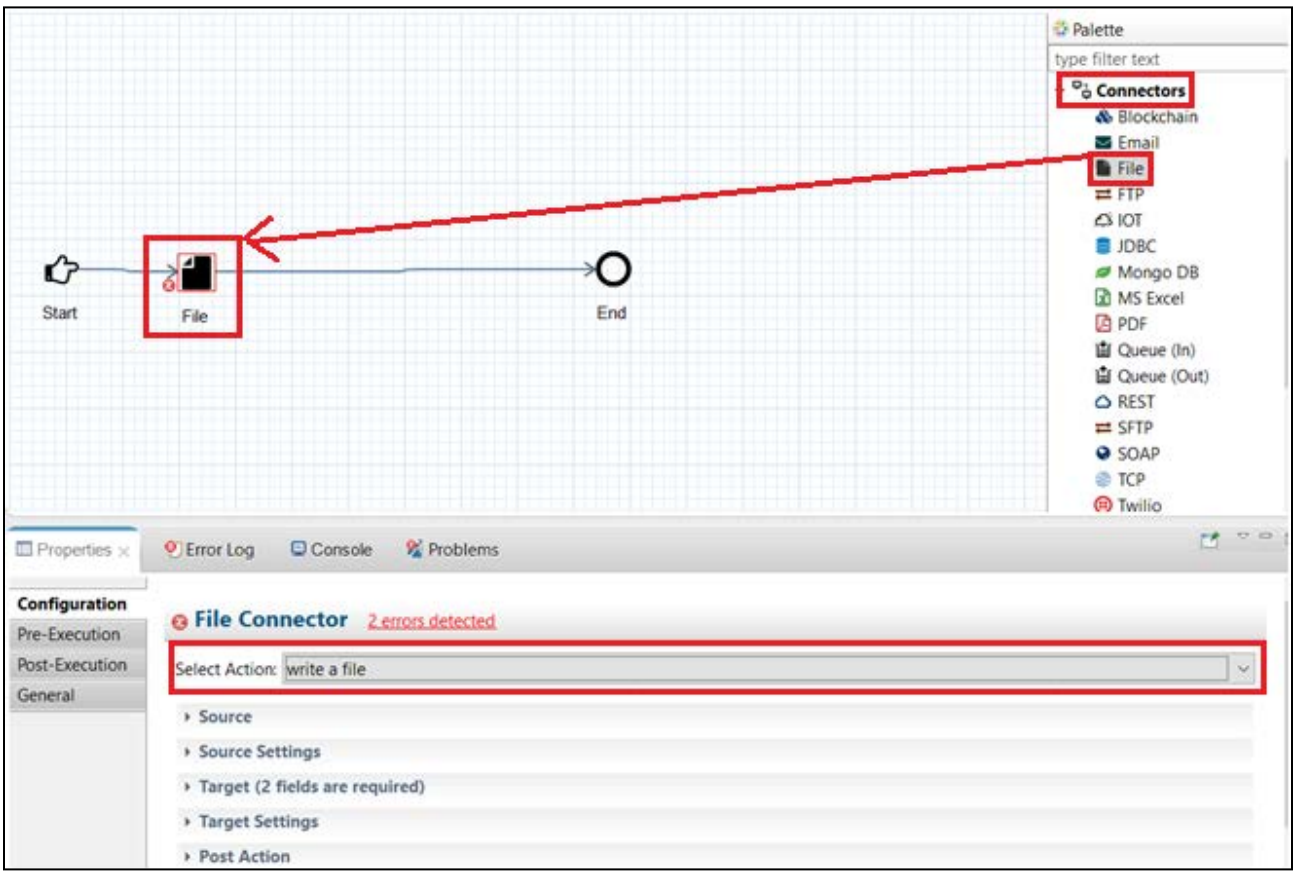
The New Flow Wizard dialog opens, as shown in the following image.



2. Type a name for your process flow in the Name field (for example, *SFDCtoORA*) and click *Finish.*

   To retrieve the input structure for your Transform (mapping from Salesforce.com to Oracle) from Salesforce.com, you need to add a File object, which writes a structure file received from Salesforce.com.

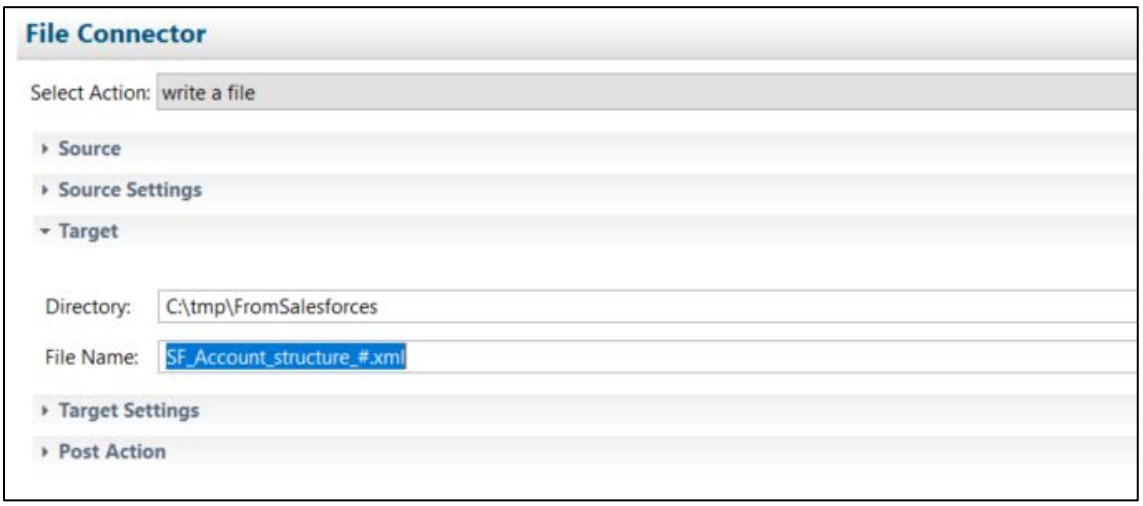
3. From the Palette, which is located in the right pane, expand *Connectors*. Click and drag the *File* object to the workspace area on the line between the *Start* and *End* objects, as shown in the following image.



4. From the Select Action drop-down list in the Properties tab of the process flow, select *write a file*.

5. Expand the *Target* group and specify values for the *Directory* and *File Name* parameters.

   You can use an asterisk character (*) to include a date time stamp in the file name or a hash character (#) to include a single digit counter in the file name. Adding multiple hash characters (###) will increase the counter limit.

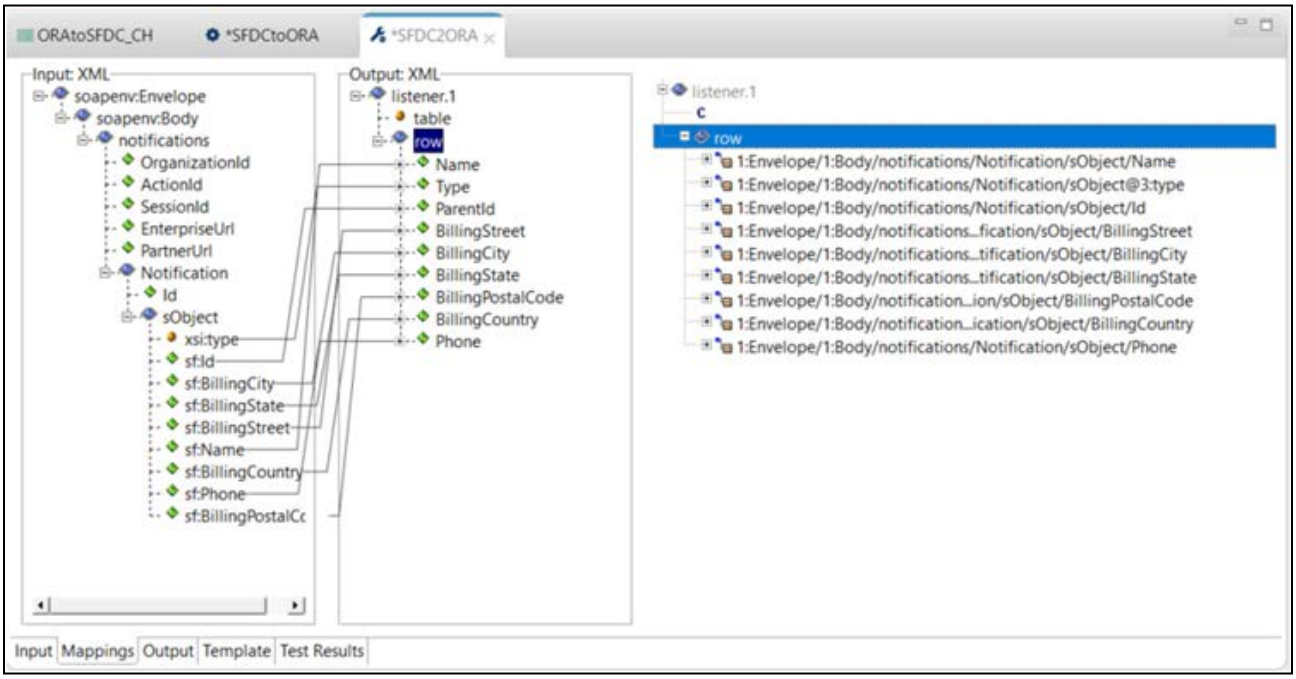


6. Click *Save* (or press *Ctrl+S*) to save your process flow configuration.

7. Follow the steps described in Building a Canonical Transform to create a new Transform named *SFDC2ORA*.

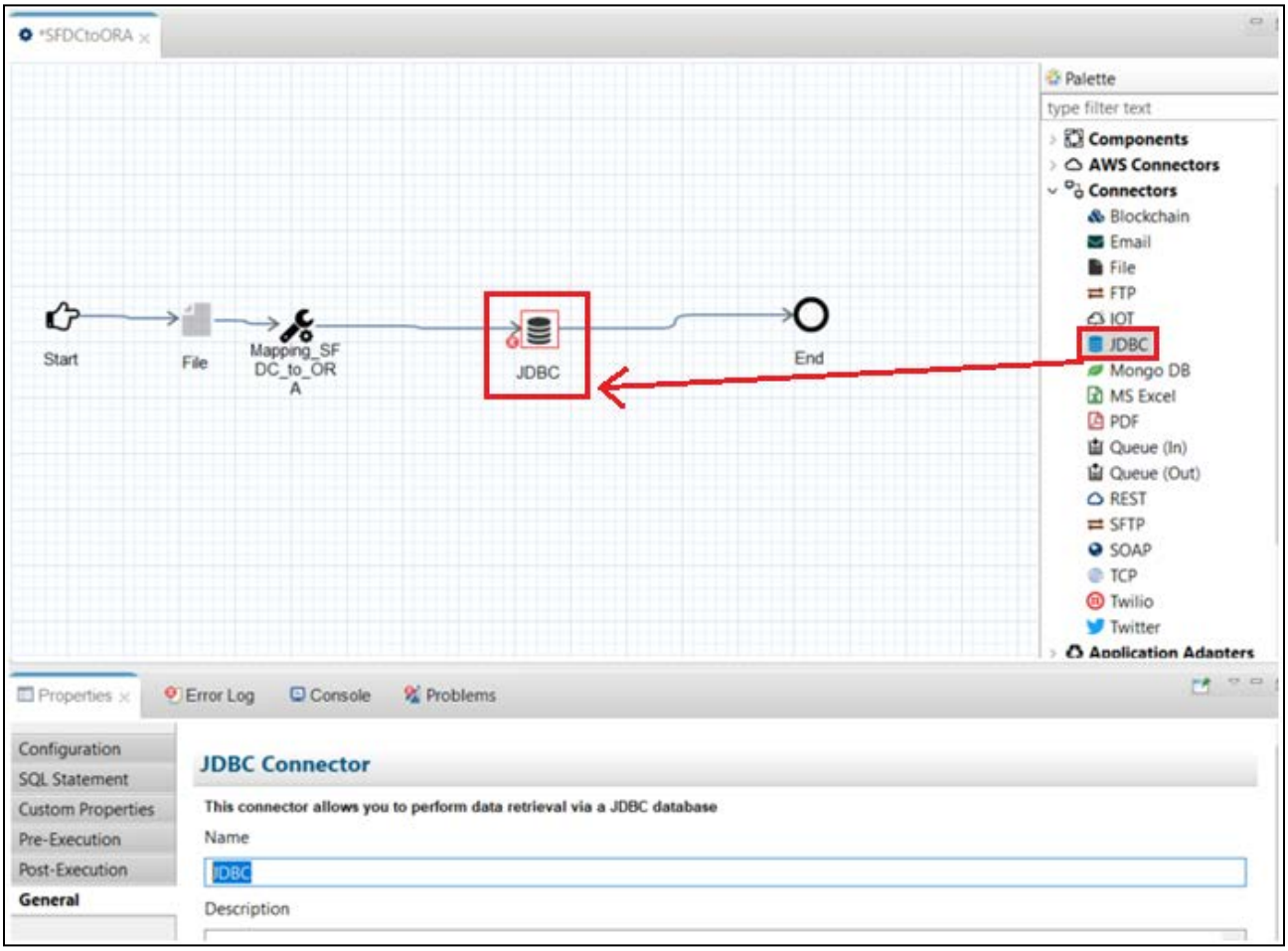
   This new Transform (*SFDC2ORA*) will use the Salesforce.com structure file that is generated using the File Write object in your process flow. The output structure will be the Oracle structure file.

   The new completed Transform should be configured and structured, as shown in the following image.

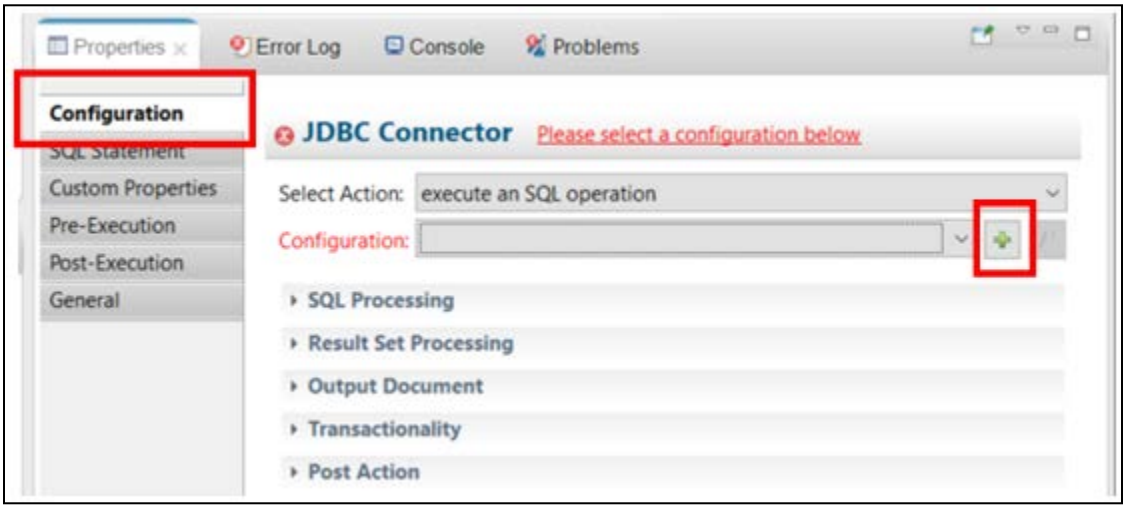


8. Click *Save* (or press *Ctrl+S*) to save your Transform.

9. From the Palette, which is located in the right pane, expand *Connectors*. Click and drag the *JDBC* object to the workspace area on the line after the Transform object, as shown in the following image.

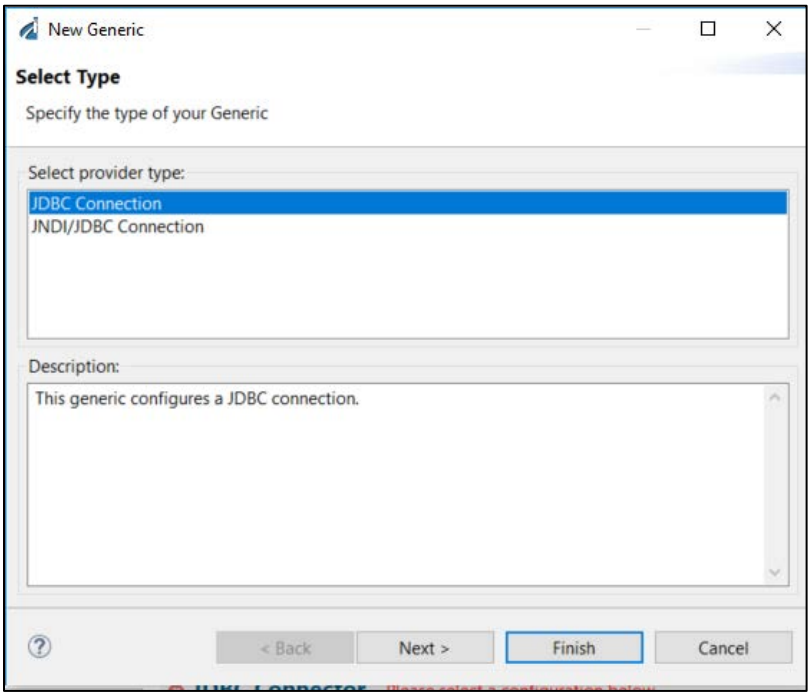


10. In the Properties tab on the lower part of the screen, select *execute an SQL operation* from the Select Action drop-down list, as shown in the following image.



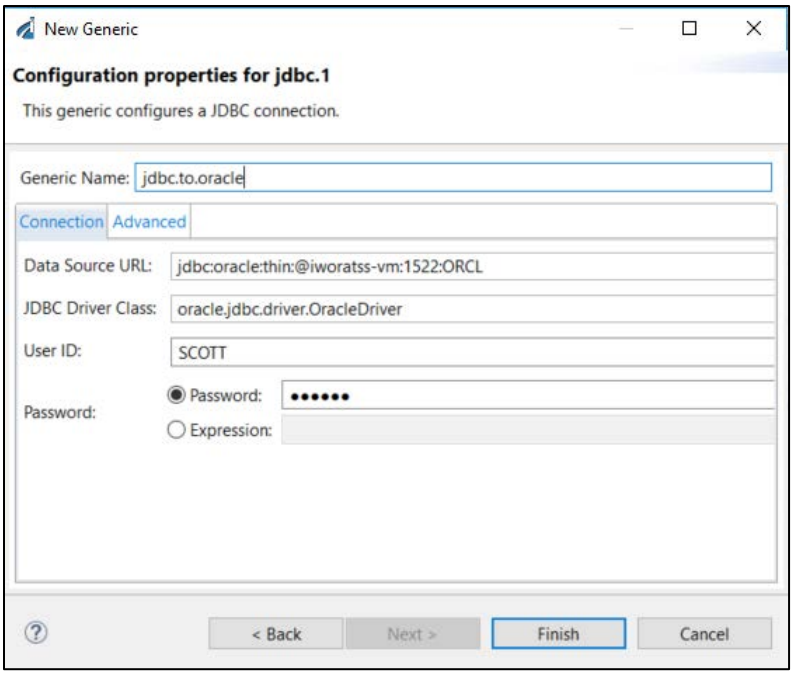
11. Click the green plus sign (+) to the right of the Configuration field.

The New Generic dialog opens, as shown in the following image.



12. From the Select provider type area, select *JDBC Connection*, and then click *Next*.

The Configuration properties for jdbc.1 pane opens, as shown in the following image.



13. Provide values for the connection parameters as required, and then click *Finish*.

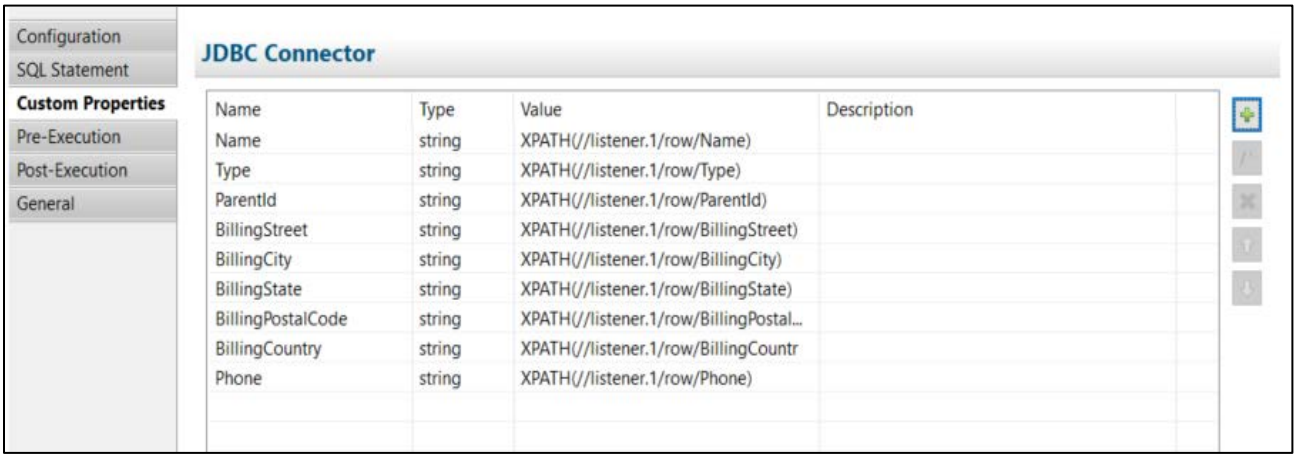
14. Click *Save* (or press *Ctrl+S*) to save your process flow.

15. Click the *SQL Statement* sub-tab under the Properties tab of the JDBC object, as shown in the following image.

This is where your SQL Insert statement can be added. For example, insert the following values into SFDCRECEIVER:

```
('?Name','?Type','?ParentId','?BillingStreet','?BillingCity','?BillingState','?BillingPostalCode','?BillingCountry','?Phone')
```

16. Click the *Custom Properties* sub-tab under the Properties tab of the JDBC object, as shown in the following image.

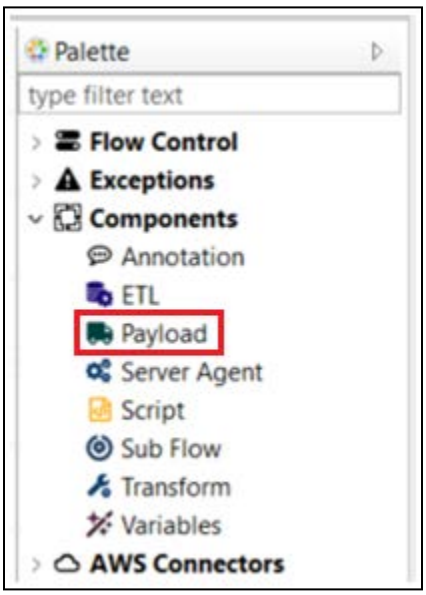


17. Add a custom property for each of the values you will be inserting through the SQL Insert statement (for example, *?Name* requires a custom property called *Name*).

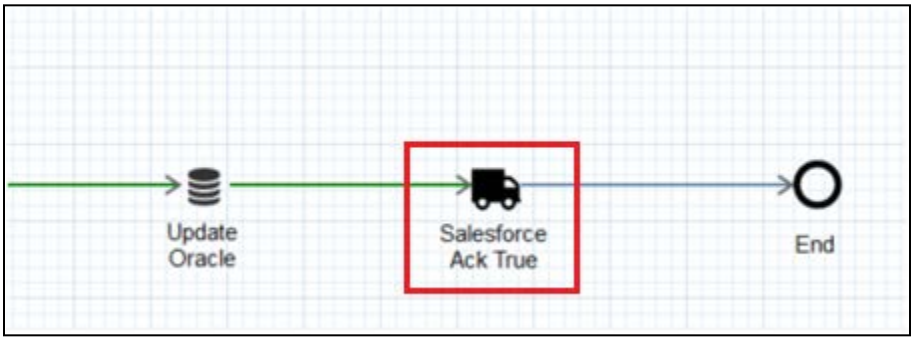
    A custom property must resolve to a corresponding value. In this particular scenario, the value is provided by an XPath.

18. Click *Save* (or press *Ctrl+S*) to save your process flow.

    You are now ready to add a Payload object to your process flow that will provide Salesforce.com with an acknowledgement (Salesforce Ack = true response document) when the Oracle database is successfully updated.

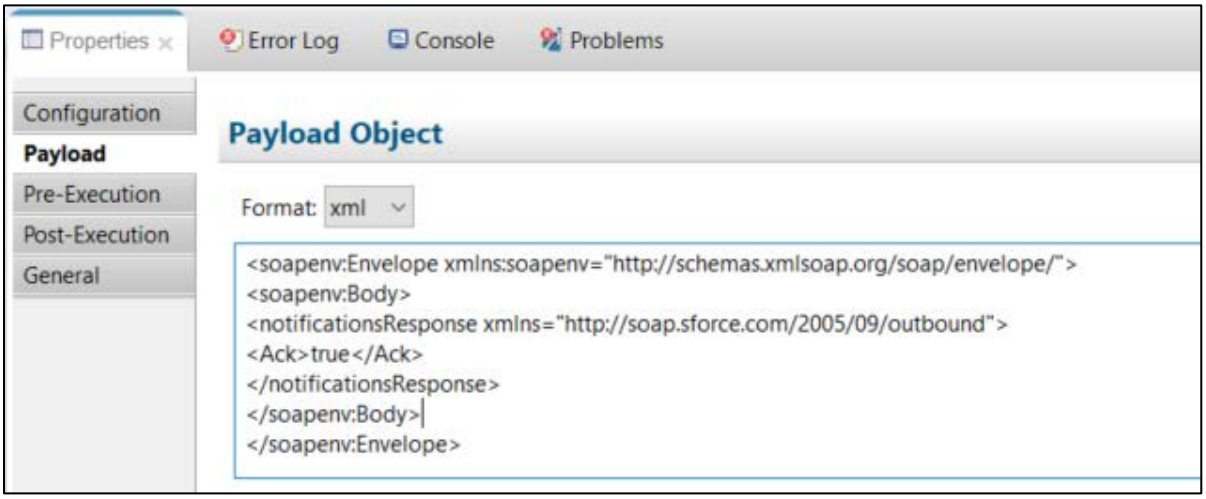


19. From the Palette, which is located in the right pane, expand *Components*. Click and drag the *Payload* object to the workspace area on the line after the Oracle JDBC object, as shown in the following image.



20. Change the behavior of the wire connecting the Oracle JDBC object and Payload object to *On Success*.

21. Click the *Payload* sub-tab under the Properties tab of the Payload object, as shown in the following image.



22. Add the following XML payload:

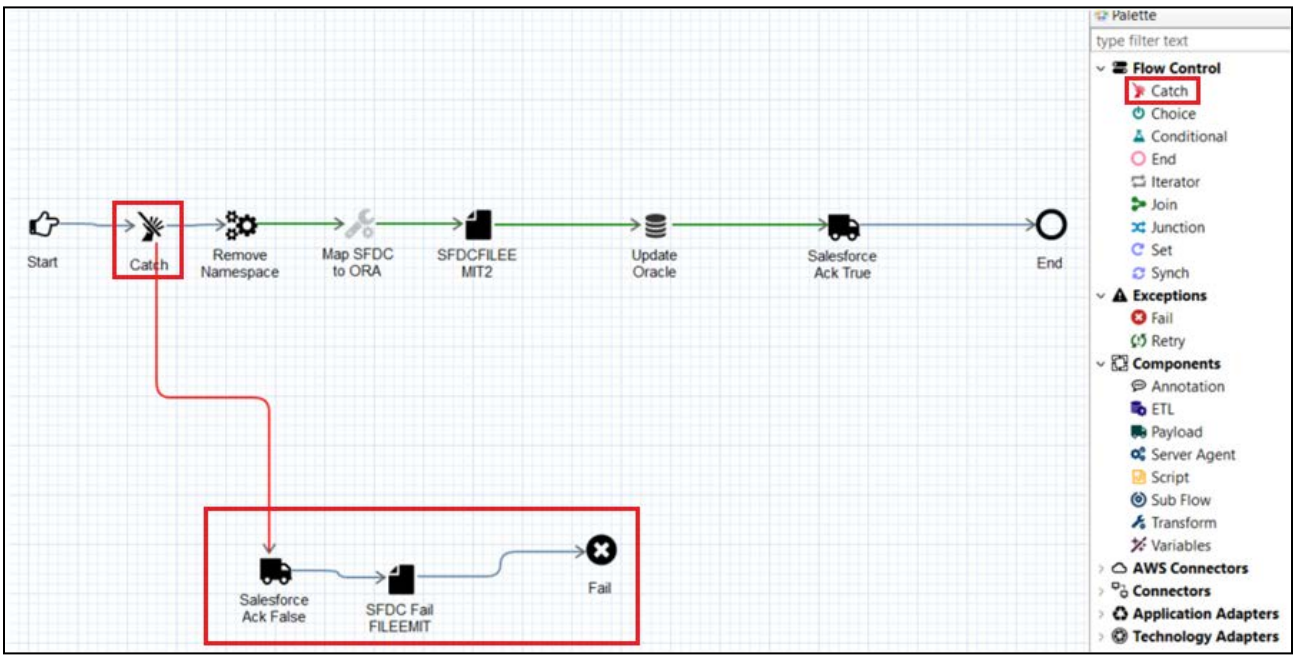
```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
<soapenv:Body>
<notificationsResponse xmlns="http://soap.sforce.com/2005/09/outbound">
<Ack>true</Ack>
</notificationsResponse>
</soapenv:Body>
</soapenv:Envelope>
```

23. In the General sub-tab of the Payload object, specify a unique name for this object (for example, *Salesforce ACK true*).

24. Click *Save* (or press *Ctrl+S*) to save your process flow.

    Note: Since the *On Success* condition has been specified for the connection between the Oracle JDBC and Payload objects, you assure that only when Oracle is successfully updated will an acknowledgement set to *true* be returned to Salesforce.com to confirm that a record was received.

    You can also add a Catch object to your process flow, which will catch all errors within your process flow and route them down the Error wire.

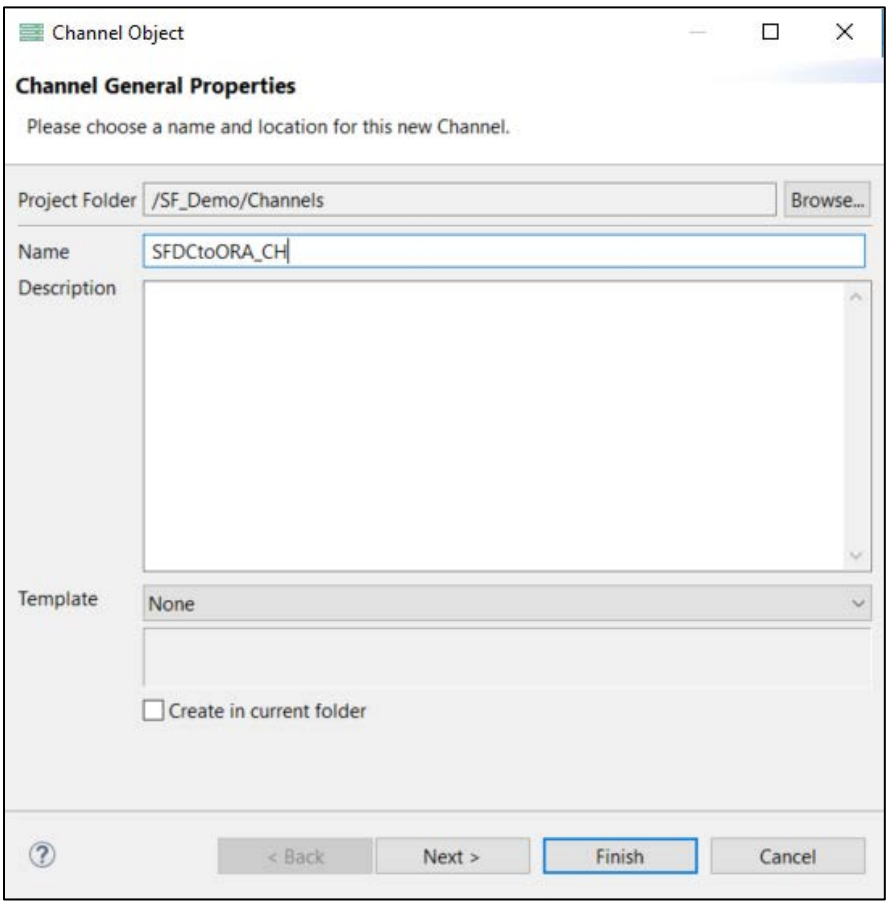


In this scenario, you can add another Payload object to the process flow, but with *Salesforce Ack = false* as a response document.

You are now ready to configure a non-blocking (nHTTP) channel listener to receive messages from Salesforce.com.

25. Expand your application project folder (for example, *SF_Demo*), right-click the *Channels* subfolder, select *New* from the context menu, and then click *Channel*.

    The Channel Object dialog opens, as shown in the following image.

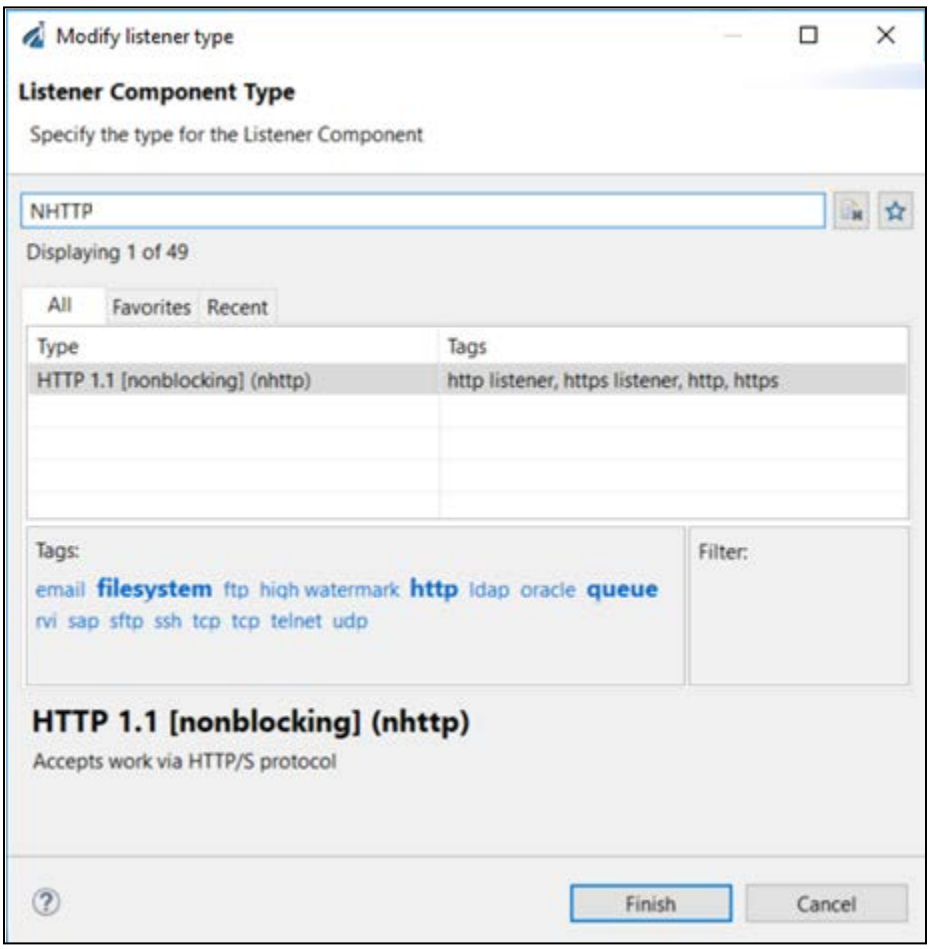


26. Type a name for your channel in the Name field (for example, *SFDCtoORA_CH*) and click *Finish*.

    The Channel Builder opens in your workspace area, as shown in the following image.

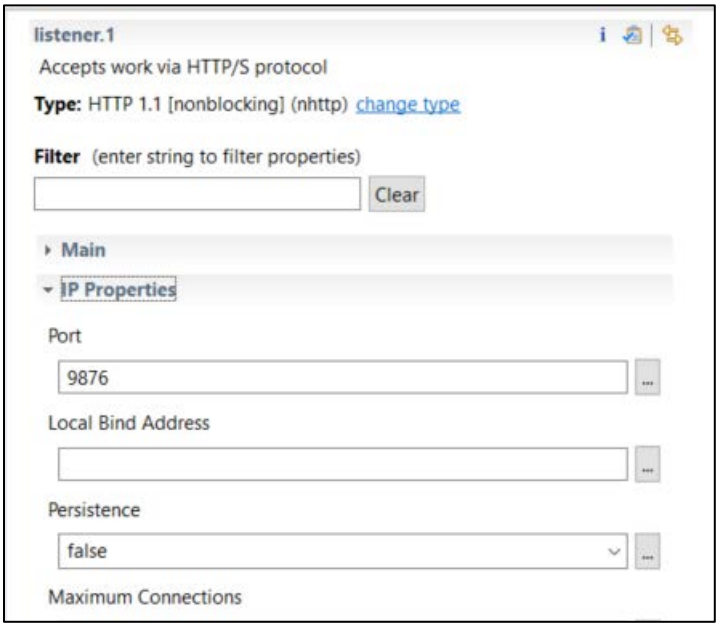
27. In the left pane, expand the *inlet: inlet.1* node, select *listener:listener.1*, and then click *change type* in the right pane.

    The Modify listener type dialog opens, as shown in the following image.



28. Type *NHTTP* in the listener search/filter field, select *HTTP 1.1 [nonblocking] (nhttp)* from the resulting list, and then click *Finish*.

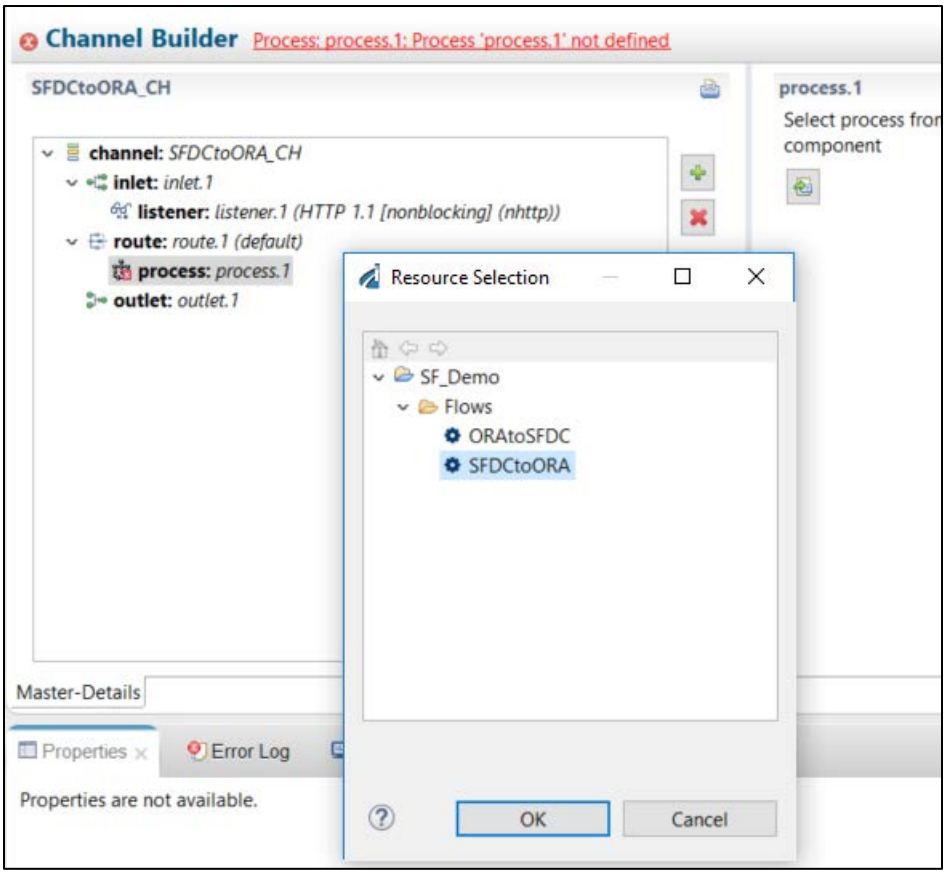
29. In the right pane, expand the *IP Properties* group and specify a port number Salesforce.com will communicate with by sending data, as shown in the following image.



30. In the left pane, expand the *route: route.1 (default)* node and select *process:process.1*.

31. In the right pane, click the *Select Process Flow* icon.

    The Resource Selection dialog opens, as shown in the following image.



32. Expand the *Flows* subfolder under your application project, select the *SFDCtoORA* process flow, and then click *OK*.

33. Click *Save* (or press *Ctrl*+*S*) to save your process flow.

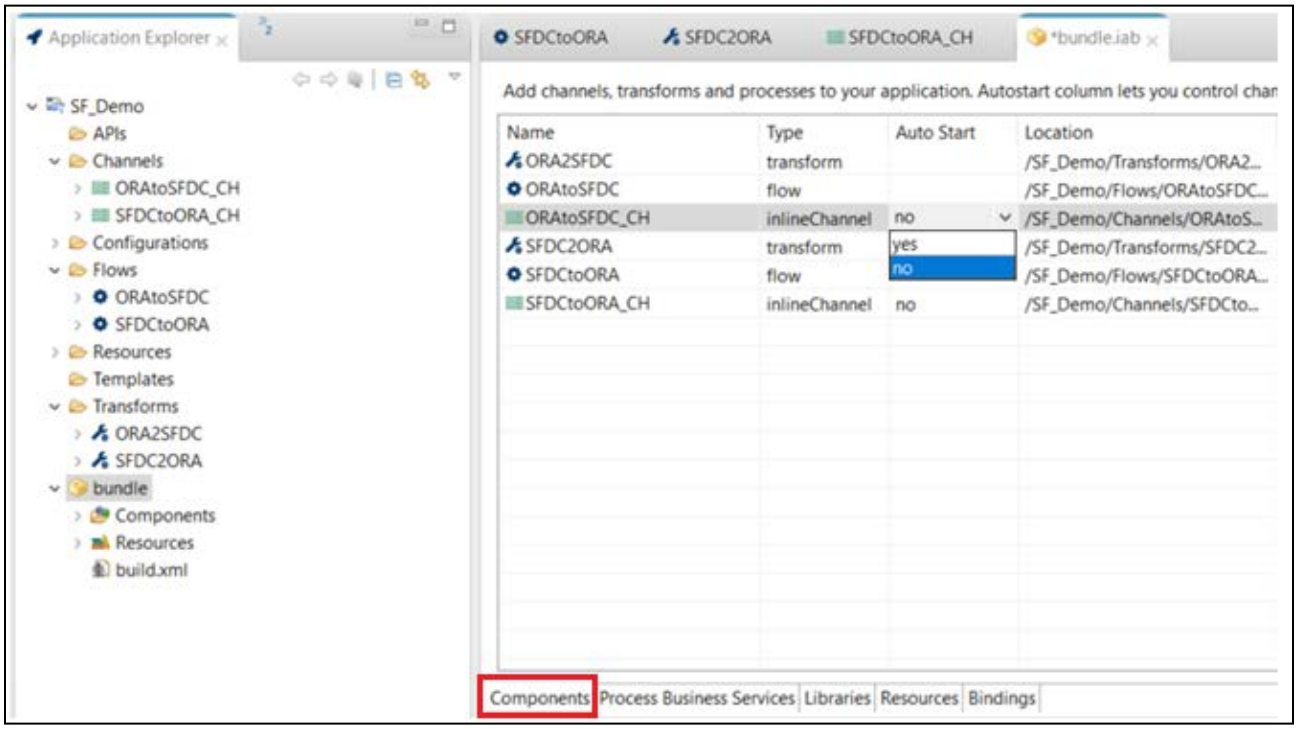
Deploying an Application From iWay Integration Tools to iWay Service Manager

This section provides suggested guidelines and recommendations when deploying an application from iWay Integration Tools (iIT) to iWay Service Manager (iSM).

Stopping and Starting Application Channels Using Auto Start

By default, all channels in an application start when the application is started. To stop channels from starting automatically, double-click on the application project *bundle* subfolder.

The application bundle opens as a new tab (bundle.iab) in your workspace area, as shown in the following image.

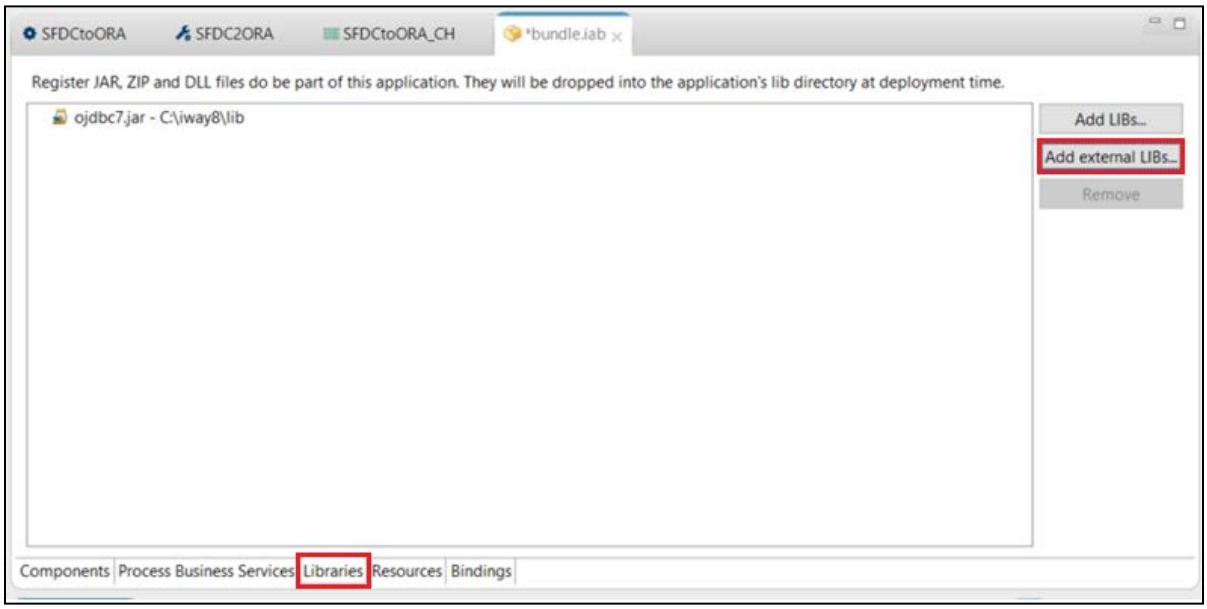


Ensure the *Components* sub-tab is selected. Click the drop-down list in the Auto Start column that corresponds to your channel and select *no*. You can determine which channels should start automatically based on your application requirements.

Adding Required Third-Party Libraries and Drivers to Your Application

You can add any third-party libraries and drivers (for example, .jar files, JDBC drivers, and so on) to your application bundle as required.

Click the *Libraries* sub-tab at the bottom of the screen and then *Add external LIBs* in the right pane, as shown in the following image.



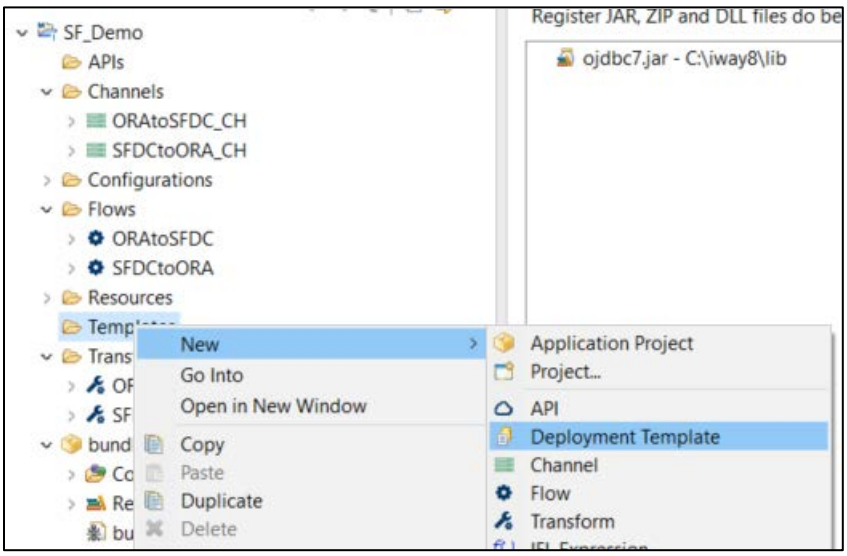
Browse to select any .jar file(s) that you want to include with your application.

Configuring a Deployment Template

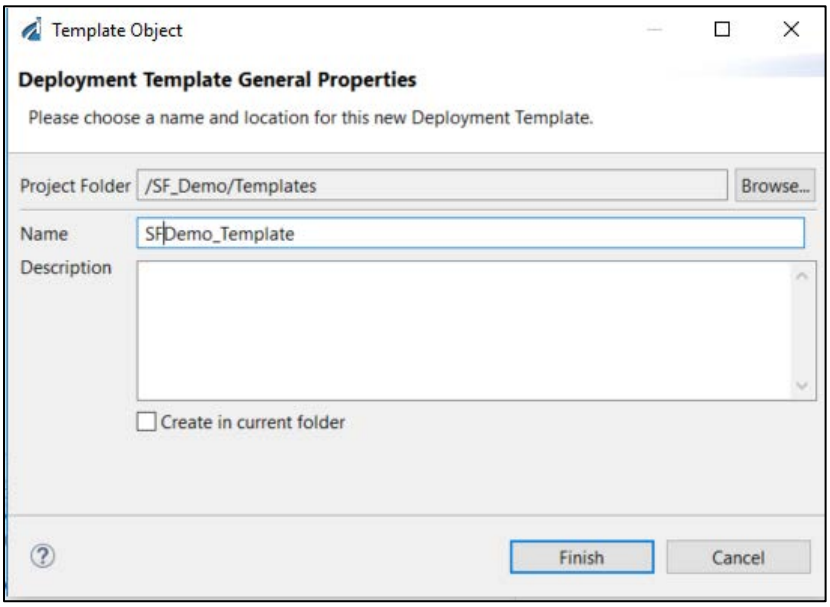
Configuring a deployment template for your application project allows you to set up a customized runtime environment prior to application deployment. Logging and tracing are enabled in the template to debug runtime issues.

To configure a deployment template:

1. Expand your application project folder (for example, *SF_Demo*), right-click the *Templates* subfolder, select *New* from the context menu, and then click *Deployment Template*, as shown in the following image.

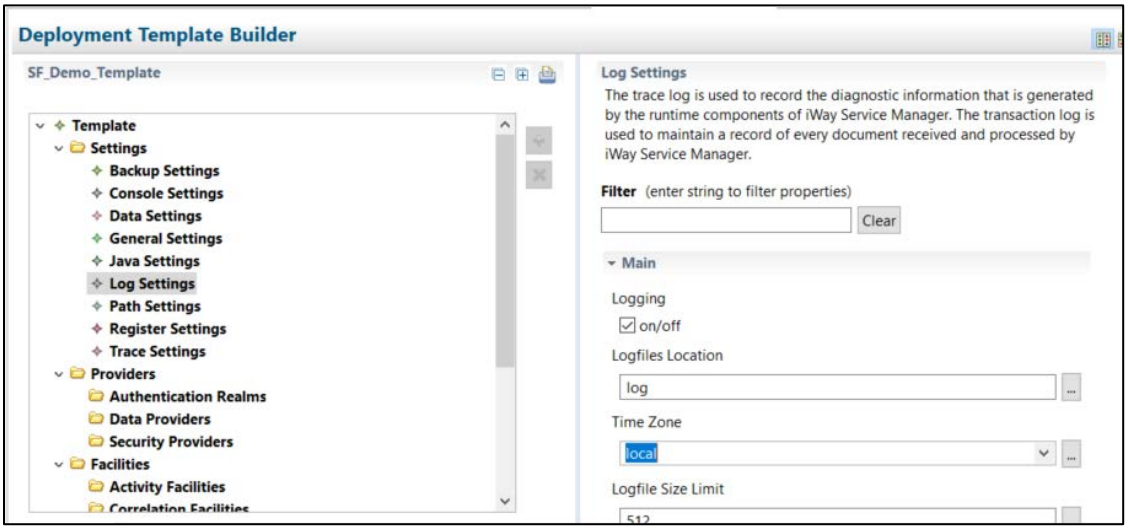


The Template Object dialog opens, as shown in the following image.



2. Type a name for your template in the Name field (for example, *SFDemo_Template*) and click *Finish*.

The Deployment Template Builder opens, as shown in the following image.

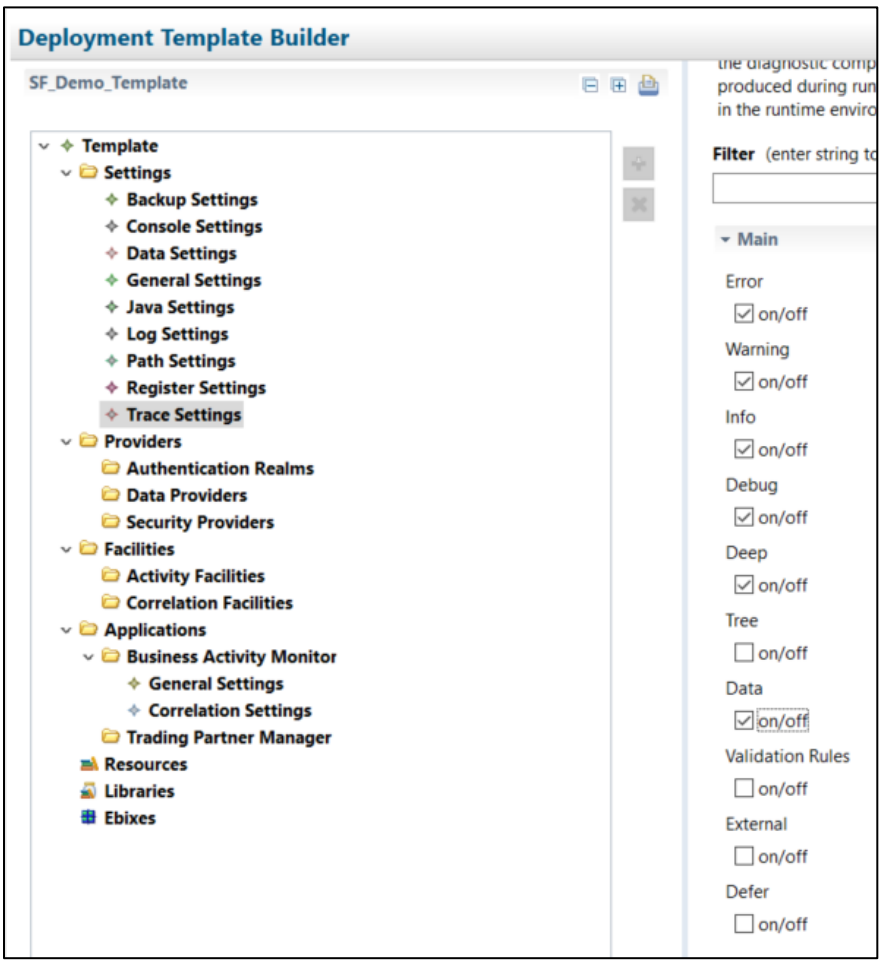


3. Expand *Settings* in the left pane and click *Log Settings*.

4. In the Log Settings area in the right pane, ensure that the *on/off* check box is selected to enable logging.

   Note: You may want to select *local* from the Time Zone drop-down list to make it easier to view and track the logs.

5. Click *Trace Settings* in the left pane of the Deployment Template Builder, as shown in the following image.



6. In the right pane, enable the following trace settings by selecting the corresponding *on/off* check boxes:

    - Error
    - Warning
    - Info

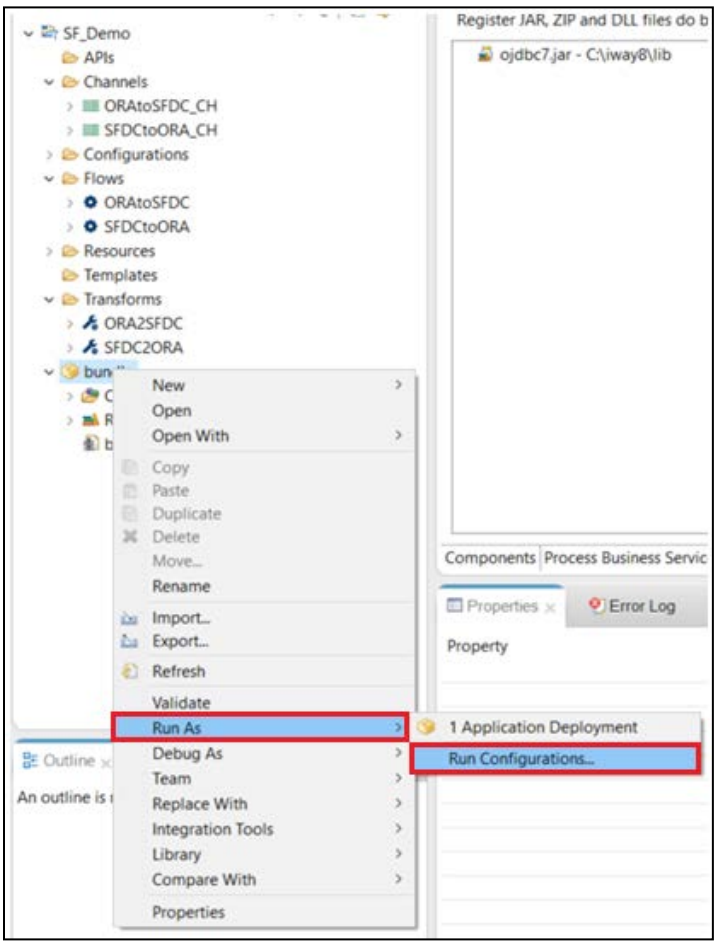
    These first three trace settings return minimal traces. Enabling *Debug*, *Deep*, and *Data* returns more extensive and detailed traces.

7. Click *Save* (or press *Ctrl*+*S*) to save your deployment template.

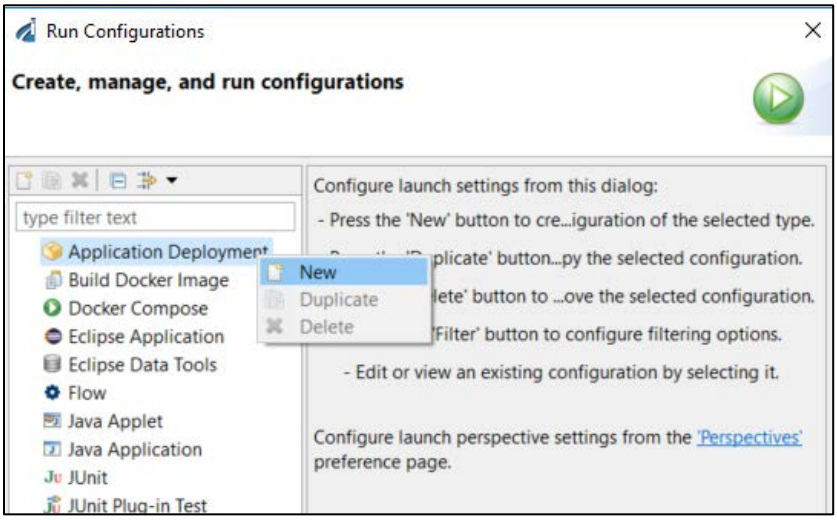
Deploying an Application

To deploy an application from iIT to iSM:

1. Expand your application project folder (for example, *SF_Demo*), right-click the application project *bundle* subfolder, select *Run As* from the context menu, and then click *Run Configurations*, as shown in the following image.

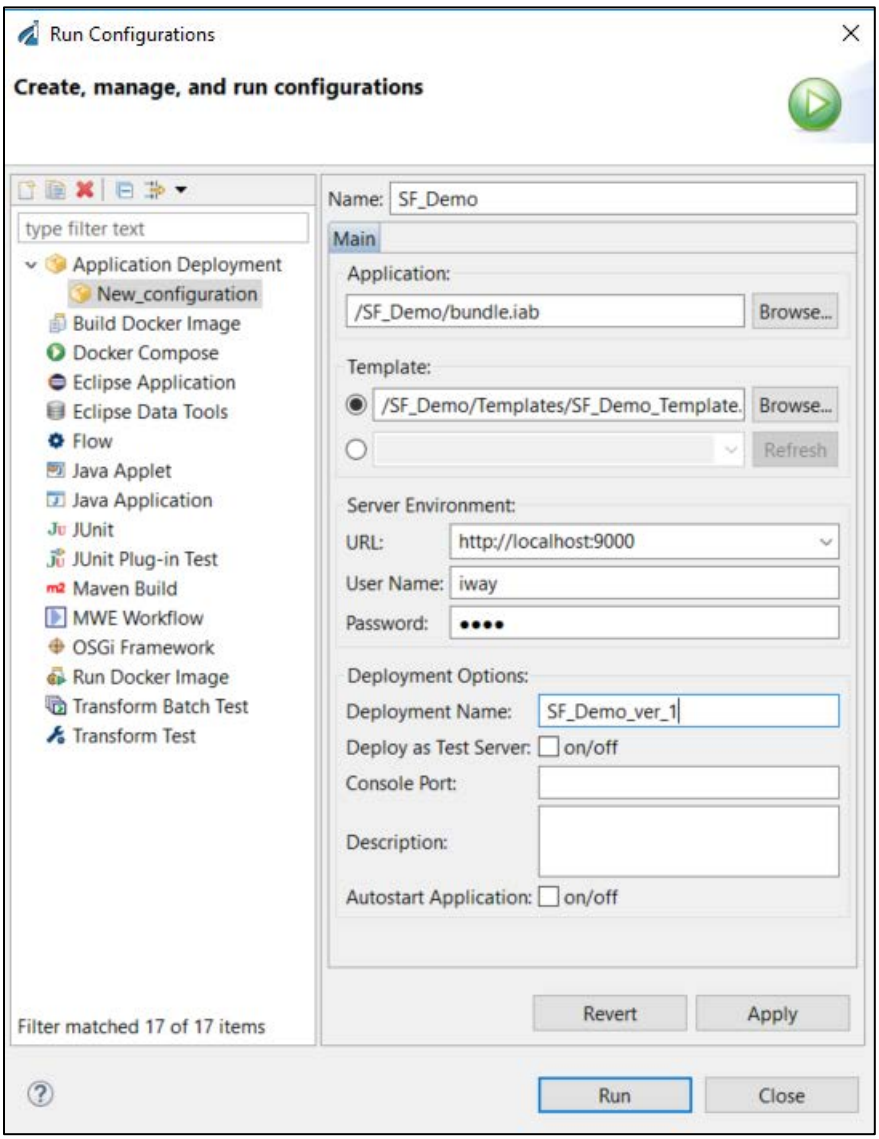


The Run Configurations dialog opens, as shown in the following image.



2. Right-click *Application Deployment* in the left pane and select New from the context menu.

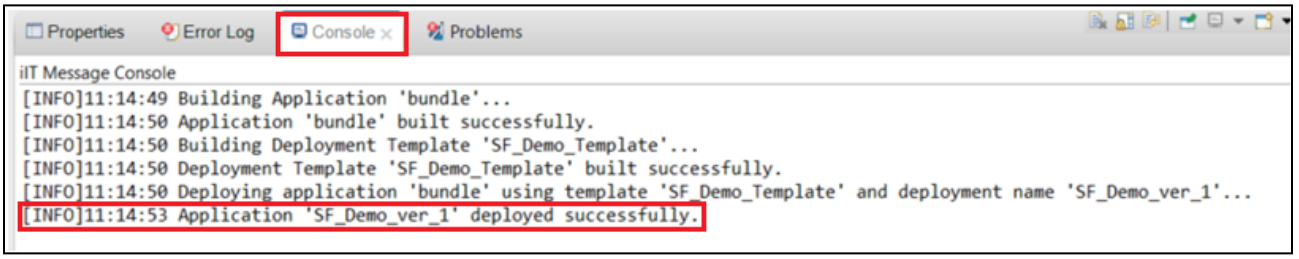
The Create, manage, and run configurations pane opens, as shown in the following image.



3. Enter the required information for your application deployment as shown. You can also add the deployment template that you configured earlier to be used during run time.

4. Click *Apply* and then *Run*.

   If you are prompted to save your changes, click *Yes*. If the application deployment was successful, you will see corresponding messages under the Console tab in iIT, as shown in the following image.



Congratulations! You have completed all of the required steps to build and deploy an application that retrieves data from an Oracle data source and propagates it to Salesforce.com.

If you have any further questions or technical issues, you can open a support case using the Technical Support Center.

For more technical content, including additional how-to's, and videos, visit the iWay and Omni Information Center.