



BPA
Platform

White Paper

Salesforce Connector Tool v2.0.1

Copyright

The copyright in this document is owned by Orbis Software Ltd T/A Codeless Platforms 2018. All rights reserved.

This publication may not, in whole or part, be reproduced, transmitted, transcribed, stored in a retrieval system or translated into any language in any form or by any means without the prior written consent of Orbis Software Ltd T/A Codeless Platforms.

Head Office:

Codeless Platforms
Suite 1 & 2 Bourne Gate
25 Bourne Valley Road
Poole
BH12 1DY
United Kingdom
Tel: +44 (0) 330 99 88 700
Email: enquiries@codelessplatforms.com

Trademarks

Orbis Software Ltd T/A Codeless Platforms owns the registered Trademark "TaskCentre®".

Microsoft, Windows, Windows 2000 Professional, Windows Vista, Windows 7, Windows 8, Windows 8.1, Windows 10, Windows 2000 Server, Windows 2003 Server, Windows 2008 Server, Windows 2008 Server R2, Windows 2012 Server, Windows 2012 Server R2, Windows 2016 Server, Internet Explorer and Outlook are all trademarks of the Microsoft Corporation.

All other Trademarks used are acknowledged as the property of their respective owners.

The information provided in this publication may contain errors, omissions, or typographical errors or may be out of date. Orbis Software Ltd T/A Codeless Platforms may change, delete, or update any published information at any time and without prior notice. The information published in this document is provided for informational purposes only and is not binding on Orbis Software Ltd T/A Codeless Platforms in any way except to the extent that it is specifically indicated to be so.

Contents

Copyright	2
Trademarks	2
Introduction	4
The Salesforce Connector Tool Pack	4
System Requirements	4
Architecture	5
About the Salesforce Event Agent	6
Event Agent Database — Notification Table	6
About the Salesforce Connector	6
Global Configuration	7
Using Extended Logging	7
About the Connection Settings Panel	8
Step Configuration	9
About the General Tab	10
About the Connection Tab	11
About the Mapping Tab	12
About the Options Tab	14
Working with Other Steps	15
Consuming XML from Other Steps	15
Exposing XML to Other Steps	15
Supported Objects	16
Tool Output	21
Runtime	21
Error Handling	21
Salesforce Accounts Security	22

Introduction

The **Salesforce Connector** tool allows communication between BPA Platform and Salesforce. This enables efficient integration between Salesforce and incompatible systems, via BPA Platform.

The Salesforce Connector Tool Pack

The tool pack consists of:

- ▶ **Salesforce Connector** — The **Salesforce Connector** is a BPA Platform tool, used to communicate with the Salesforce API. It must be installed on the BPA Platform server, and on any BPA Platform client machines. It is this that provides the translation between the BPA Platform data and the Salesforce objects and operations.
- ▶ **Salesforce Event Agent** — The Salesforce Event Agent is a web service, receiving specific events from Salesforce and writing them to a database. These events can then be used by BPA Platform steps for processing.

For more information, see [About the Salesforce Event Agent](#).

System Requirements

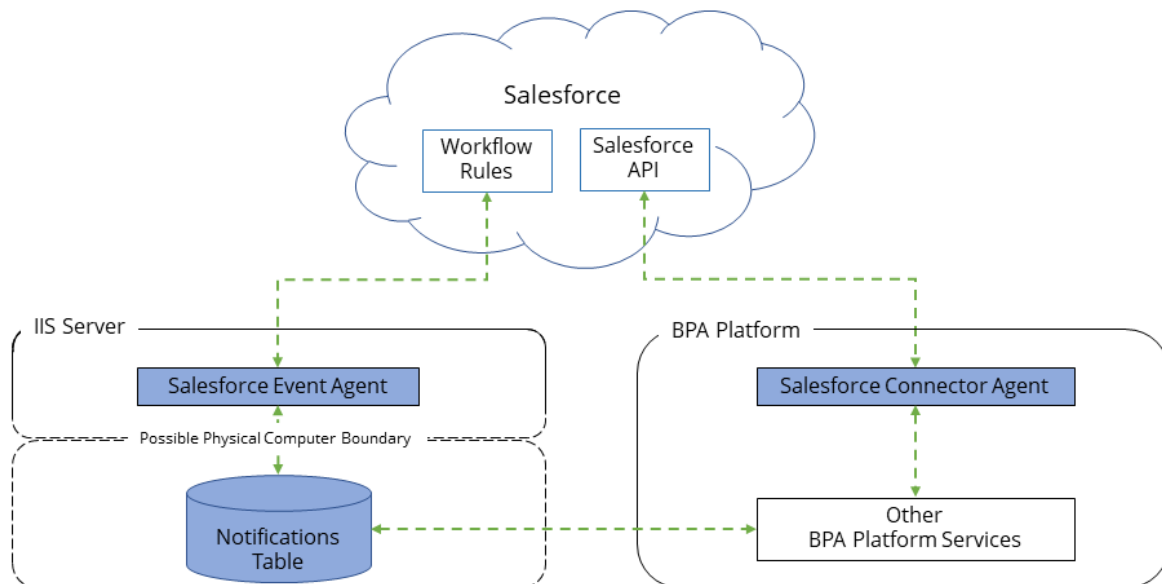
The **Salesforce Connector** tool pack requires BPA Platform (formerly TaskCentre) version 4.6 or above.

This tool is compatible with the following versions of Salesforce:

- ▶ Salesforce Cloud Services

Architecture

The diagram below provides a high-level system architecture overview of the **Salesforce Connector** tool pack, with BPA Platform and Salesforce:



The Salesforce event agent can be installed on any compatible Microsoft Windows Server. If required, you can install the event agent on the same machine running the BPA Platform server instance.

The event agent is independent of BPA Platform. It runs as a service on the server; only one Salesforce event agent can be installed on the server at any one time. The agent writes events to a **Notifications** table. This table can be located on the same server running the agent, or on a remote machine.

The **Salesforce Connector** is used to read and write data directly to Salesforce. It must be installed on the same server as BPA Platform, and on any client installations of BPA Platform.

About the Salesforce Event Agent

The Salesforce event agent receives workflow events from your Salesforce instance. These are stored in a database for later retrieval and processing by a BPA Platform task step. It runs as a passive web service on the server.

For a detailed description of how to install and configure the Event Agent, see [Installing and Configuring the Salesforce Event Agent](#).

Event Agent Database — Notification Table

The Salesforce event agent makes use of a database table — **Notification**. This table holds workflow events' details.

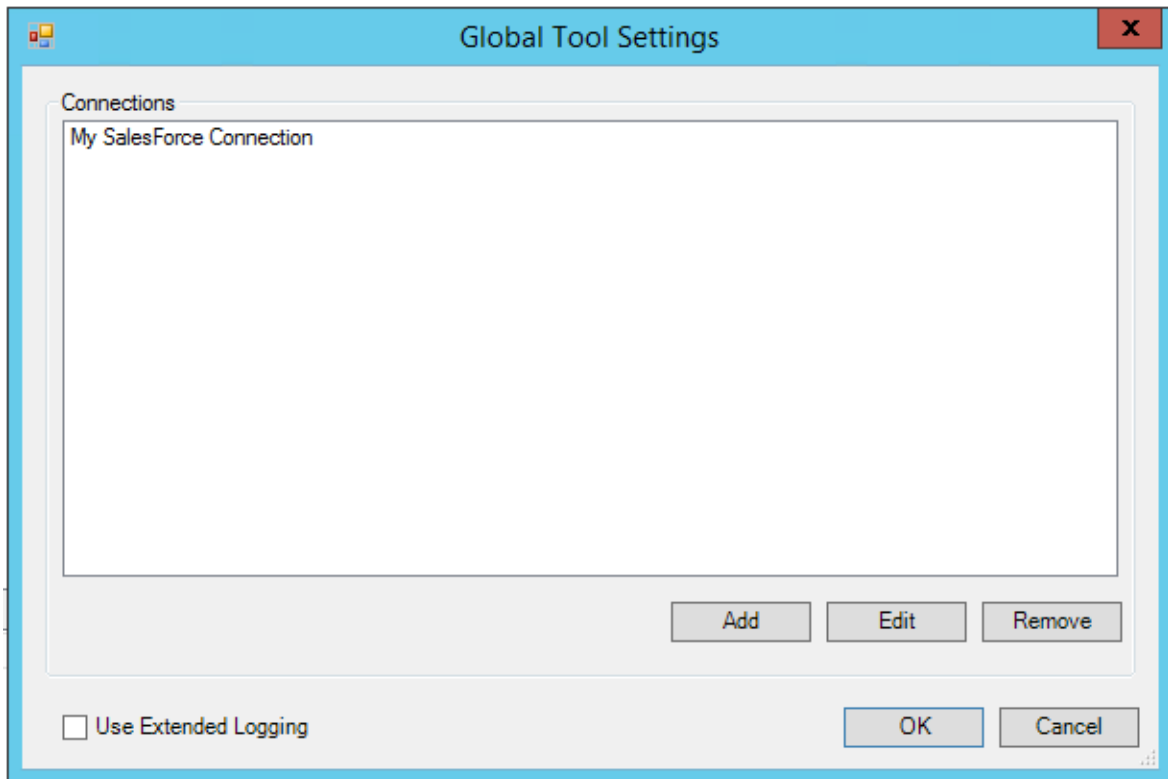
Column Name	Type	Description
<code>Id</code>	<code>int</code>	Primary Key
<code>NotificationId</code>	<code>nchar</code>	
<code>ActionId</code>	<code>nchar</code>	
<code>ObjectId</code>	<code>nchar</code>	
<code>ObjectType</code>	<code>nchar</code>	
<code>DataXML</code>	<code>xml</code>	The full XML string of the workflow event
<code>UserParam</code>	<code>varchar</code>	
<code>EnterpriseURI</code>	<code>nchar</code>	
<code>CreateDateTime</code>	<code>datetime</code>	Timestamp of when the record was added

About the Salesforce Connector

The **Salesforce Connector** provides a means of communication between the BPA Platform server, and your Salesforce instance. You install the connector on the BPA Platform server.

Global Configuration

The global configuration for this tool is used to create the connection to the Salesforce server. You access the **Global Tool Settings** dialog box via **System > Tools > SKIT > Salesforce Connector**.



When a connection is created, the available objects and fields of Salesforce are retrieved.

Click **Add** to create a connection to Salesforce.

Using Extended Logging

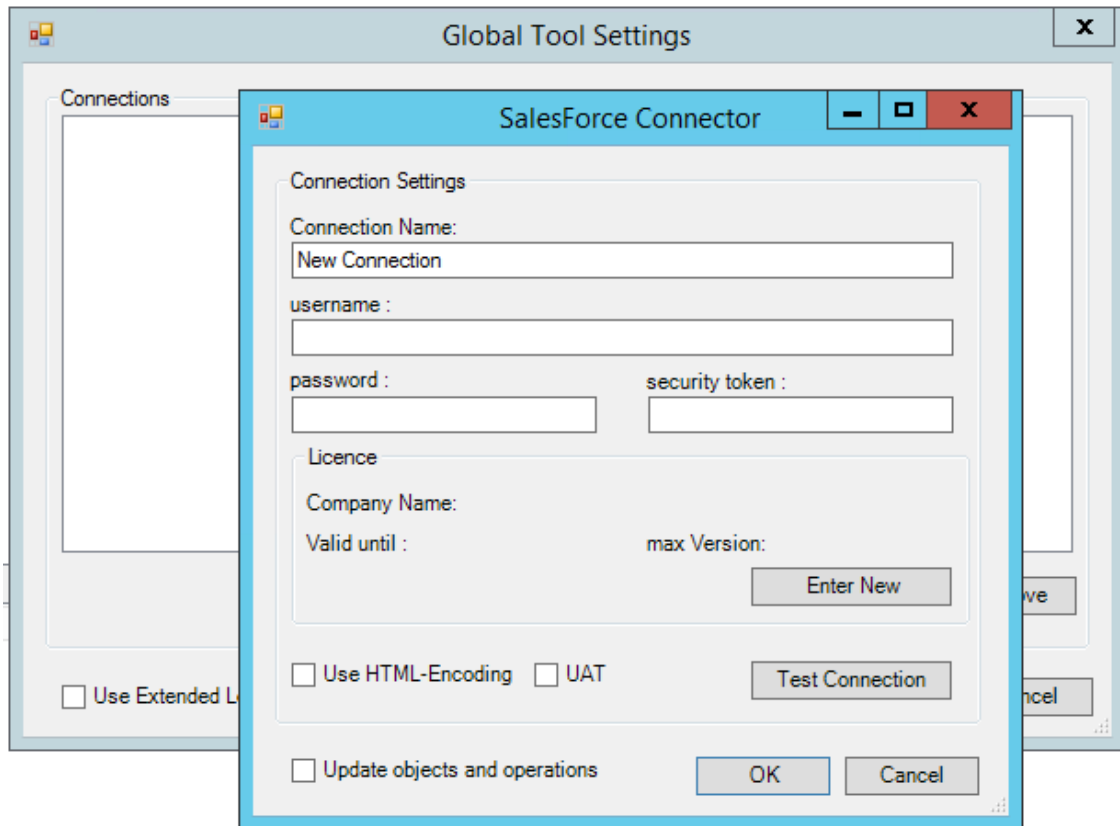
Enabling this option exposes the XML parsed between the **Salesforce Connector** and Salesforce.

Without extended logging, the Event Log only contains start and end of transaction messages, plus any error messages encountered at runtime.

You can view the extended log in the BPA Platform Event Log (**Manage > Event Log**).

About the Connection Settings Panel

You configure the connection to the Salesforce instance here.



Configure a meaningful **Connection Name** for this connection; this is the name displayed when adding tasks for Salesforce.

Enter the credentials of the Salesforce user, created in [Prerequisites](#), allowed to access the API, that is:

- ▶ **username**
- ▶ **password**
- ▶ **security token**

You can choose whether to encode ASCII-compatible characters in the XML passed between BPA Platform and Salesforce — select **Use HTML-Encoding** to enable this; it is recommended you do this.

Select **UAT** if the instance of Salesforce you are using is the Test or Sandbox version. This changes the URL the **Salesforce Connector** tool uses to access Salesforce itself.

Use the **Test connection** button to ensure the **Salesforce Connector** can access Salesforce correctly.

Licence Panel

Each instance of the Salesforce connection must be licensed — for more information, contact your Codeless Platforms' representative. The license is in the form of a `.stg` file. It contains the licensed company name, the expiry date of the license (**Valid until**), and the licensed **Salesforce Connector** tool version (**max Version**).

Click **Enter New**, and locate and open the license file.

Update Objects and Operations

Enabling this option allows the Salesforce schema within BPA Platform to be refreshed every time an update is made in Salesforce itself.

Should you also upgrade Salesforce to a newer, compatible version, all related schemas within BPA Platform will be refreshed.



NOTE:

Existing task steps using the **Salesforce Connector** must be refreshed in order to use the latest schema.

Step Configuration

When creating new tasks, the **Salesforce Connector** tool is located under **SKIT** of the Task Browser.



NOTE:

It is not possible to add the connector tool to the task until at least one Salesforce connection has been defined in the **Global Tools Settings** (see [Global Configuration](#)).

About the General Tab

The screenshot shows a window titled "Step Configuration Form" with a blue header bar containing standard window controls. Below the header is a tabbed interface with four tabs: "General", "Connection", "Mapping", and "Options". The "General" tab is active. It contains a "Name:" label followed by a text box with the value "Salesforce Connection - Get Records". Below this is a "Description:" label followed by a large, empty text area. At the bottom of the form is a "Data source" section. It has three radio buttons: "No data source" (which is selected), "Task step:", and "Custom schema:". The "Task step:" option has a dropdown menu next to it. The "Custom schema:" option has a "Define" button next to it. Below these is an "Input source variable:" label followed by a text box. A small instruction text below the text box reads: "Drag a variable that will contain the XML data from the Task Centre browser window to the text box above." At the bottom right of the form are "OK" and "Cancel" buttons.

Provide a meaningful **Name** and **Description** for this step.



TIP:

If this task instance makes use of two or more **Salesforce Connector** steps, ensure the **Name** used is unique for each individual step.

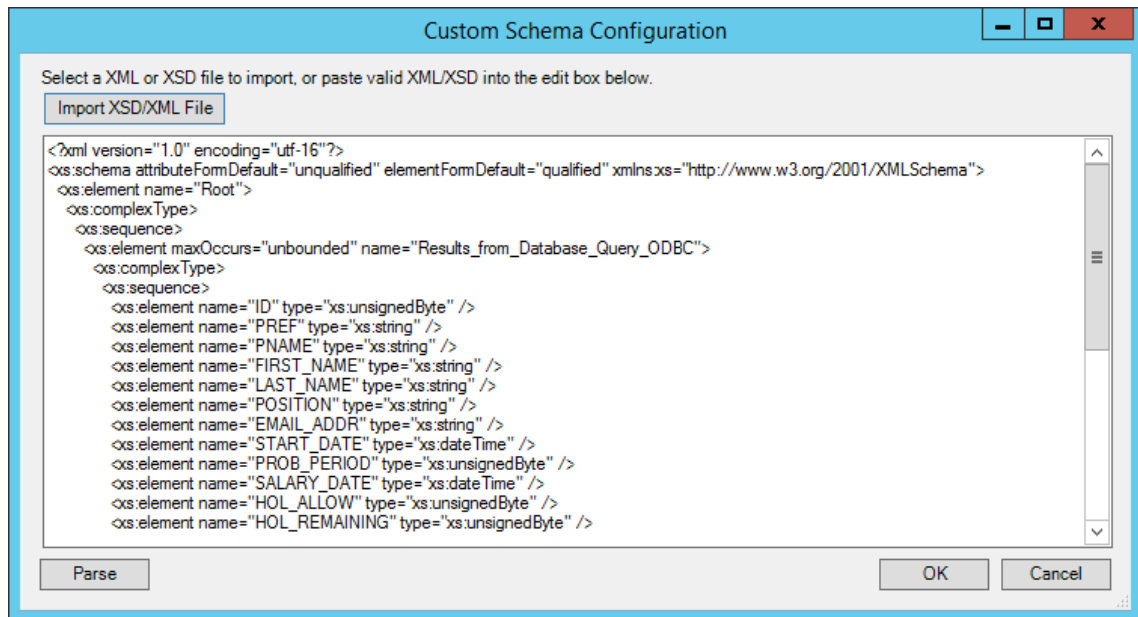
Data source can either be:

- ▶ **No data source** — If you don't make use of a dedicated XML input source, select this option to use BPA Platform variables in place of the XML objects' fields. These can then be mapped to operational fields — see [About the Mapping Tab](#).

For example, you can extract various bits of information from an email and store them in BPA Platform variables. Then, having mapped them to Salesforce objects and operations, these can be stored in the Salesforce database for later use.
- ▶ **Task step** — The data source can be set to an available BPA Platform XML data source. Only those steps that are capable of natively exposing an XML document at runtime are listed. These may be another **Salesforce Connector** step, or a tool, such as **Convert Recordset to XML**.
- ▶ **Custom schema** — An XML schema defines the structure of the parsed XML: what tags are present, and the nesting of the tags. You **Define** the schema of the XML that will be used as the input data source for this step. The **Salesforce Connector** tool uses the industry standard XSD format. Any XML processed by this step must

conform to this schema else an error will be reported.

If the XSD schema is available, either import it into the **Custom Schema Configuration** (use the **Import XSD/XML File** button), or copy and paste it into the configuration box.



The image shows a dialog box titled "Custom Schema Configuration". It has a blue header bar with standard window controls. Below the header, there is a text area with the instruction: "Select a XML or XSD file to import, or paste valid XML/XSD into the edit box below." To the left of the text area is a button labeled "Import XSD/XML File". The text area contains an XSD schema for a root element named "Root". The schema defines a sequence of elements: "ID" (unsigned byte), "PREF" (string), "PNAME" (string), "FIRST_NAME" (string), "LAST_NAME" (string), "POSITION" (string), "EMAIL_ADDR" (string), "START_DATE" (date time), "PROB_PERIOD" (unsigned byte), "SALARY_DATE" (date time), "HOL_ALLOW" (unsigned byte), and "HOL_REMAINING" (unsigned byte). At the bottom of the dialog, there are three buttons: "Parse", "OK", and "Cancel".

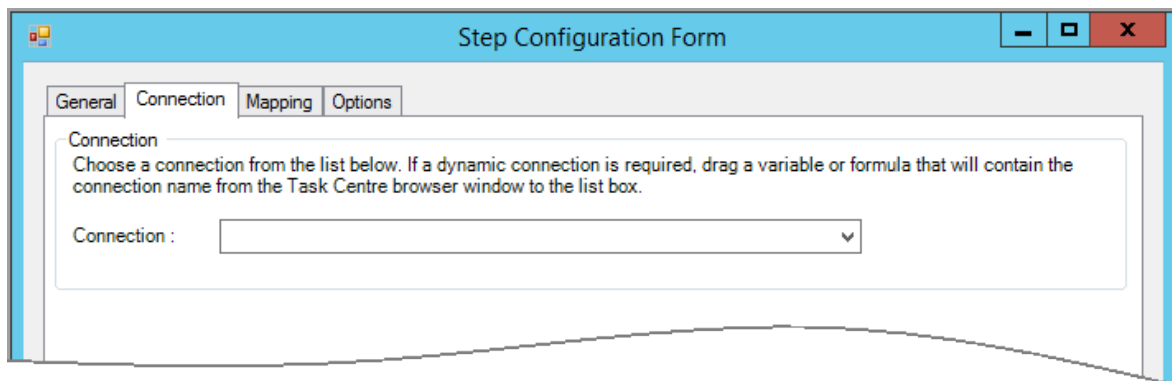
```
<?xml version="1.0" encoding="utf-16"?>
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="Root">
    <xs:complexType>
      <xs:sequence>
        <xs:element maxOccurs="unbounded" name="Results_from_Database_Query_ODBC">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ID" type="xs:unsignedByte" />
              <xs:element name="PREF" type="xs:string" />
              <xs:element name="PNAME" type="xs:string" />
              <xs:element name="FIRST_NAME" type="xs:string" />
              <xs:element name="LAST_NAME" type="xs:string" />
              <xs:element name="POSITION" type="xs:string" />
              <xs:element name="EMAIL_ADDR" type="xs:string" />
              <xs:element name="START_DATE" type="xs:dateTime" />
              <xs:element name="PROB_PERIOD" type="xs:unsignedByte" />
              <xs:element name="SALARY_DATE" type="xs:dateTime" />
              <xs:element name="HOL_ALLOW" type="xs:unsignedByte" />
              <xs:element name="HOL_REMAINING" type="xs:unsignedByte" />
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

If the XSD schema is not available, you can import an example of the runtime XML (**Import XSD/XML File**), or copy and paste it into the configuration box. Use the **Parse** button to create the schema.

- ▶ **Input source variable** — As well as defining the schema, specify the BPA Platform variable that contains the XML data at runtime.

About the Connection Tab

You specify the Salesforce **Connection** this step must use.



The image shows a dialog box titled "Step Configuration Form". It has a blue header bar with standard window controls. Below the header, there are four tabs: "General", "Connection", "Mapping", and "Options". The "Connection" tab is selected. Inside the "Connection" tab, there is a text area with the instruction: "Choose a connection from the list below. If a dynamic connection is required, drag a variable or formula that will contain the connection name from the Task Centre browser window to the list box." Below this text area, there is a label "Connection :" followed by a dropdown menu.

All **Connections** created in **Global Tools Settings** (see [About the Connection Settings Panel](#)) are presented here.

Alternatively, you can use a BPA Platform variable to create a dynamic connection, where the connection used is determined by runtime circumstances. At runtime, the contents of the variable must match the name of one of the configured Salesforce connections (see [Global Configuration](#)) — this is case sensitive.

About the Mapping Tab

Here you define links between the incoming XML (left-side XML tree control) and the outgoing data that is sent to Salesforce (right-side XML tree control). This defines how, at runtime, the incoming XML is to be translated into the XML required for the relevant object and operation.

The screenshot shows the 'Step Configuration Form' with the 'Mapping' tab selected. The 'Object' dropdown is set to 'Contact' and the 'Operation' dropdown is set to 'Get'. The 'Transaction Mode' section has 'Transaction per Object' selected. The 'Design' tab is active, showing two XML tree controls. The left tree has a 'Root' node. The right tree is expanded for the 'Contact' object, showing fields: '_externalId', 'Id', 'IsDeleted', 'MasterRecordId', 'AccountId', 'LastName', 'FirstName', 'Salutation', 'Name', and 'OtherStreet'. A 'Clear All Links' button is at the bottom right of the right tree. 'OK' and 'Cancel' buttons are at the bottom of the form.

The **Object** drop-down shows the available objects. The **Operations** drop-down shows the operations available for the selected **Object**; for all objects, these are:

- ▶ **Get** — The **get** operation is used to retrieve records.
- ▶ **Update** — The **update** operation updates fields in existing Salesforce records. To successfully carry out this operation, you must ensure the **Id** field of the outgoing data object is mapped correctly.
- ▶ **Create** — The **create** operation adds new records to the selected outgoing data object.
- ▶ **Upsert** — The **upsert** operation creates new records and updates existing ones based on the value of the **externalId** field: if matched, then the record is updated; if not, then a new record is created.
- ▶ **Delete** — The **delete** operation deletes existing records. To successfully carry out this operation, you must ensure the **Id** field of the outgoing data object is mapped correctly.

For a detailed list of Salesforce objects, go to [Supported Objects](#).

User-defined objects, if exposed, are also made available for mapping here — these have a **__c** suffix.

You create the links by clicking and dragging the data source's field (left) onto its corresponding tool input data field (right). Note that only linked fields are used in the output XML. BPA Platform formulas and variables can be included in the incoming data — drag them from the BPA Platform Browser to the data source's incoming XML tree control. These can then be linked to tool input data fields. Note that BPA Platform recordsets must first be converted to XML using either the **Convert Recordset to XML** or **Transform Data** tool before they can be used here.

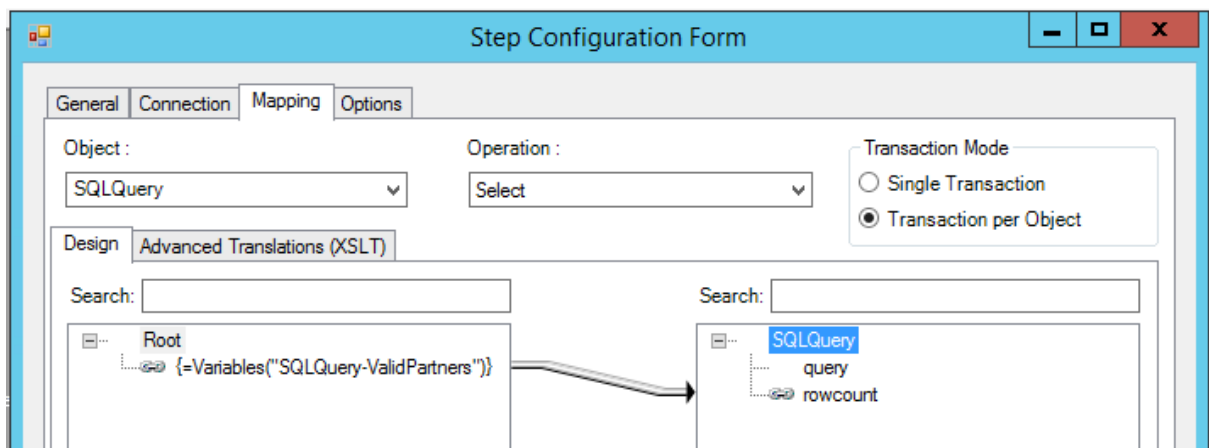
The **Salesforce Connector** tool uses eXtensible Stylesheet Language (XSLT) to translate the received XML. The **Advanced Translations (XSLT)** tab shows the XSLT generated for the links created for the object and operation. Use **Enable Free Type Mode** to directly edit the XSLT — this is particularly useful when translating a non-standard translation requirement.

Using the SQLQuery Object

An additional object is made available for use — **SQLQuery** — with a single operation of **Select**. You can use this object to query the Salesforce database, to either retrieve data or count the number of records that match specified criteria.

The **SQLQuery** object does not come with any predetermined fields. Instead, you create a variable (global or task) containing the SQL, then add it to **SQLQuery**'s incoming XML tree control. Depending on the expected results, this can then be mapped to the relevant outgoing data field — **query** for data retrieval, or **rowcount** for the number of records.

For example:



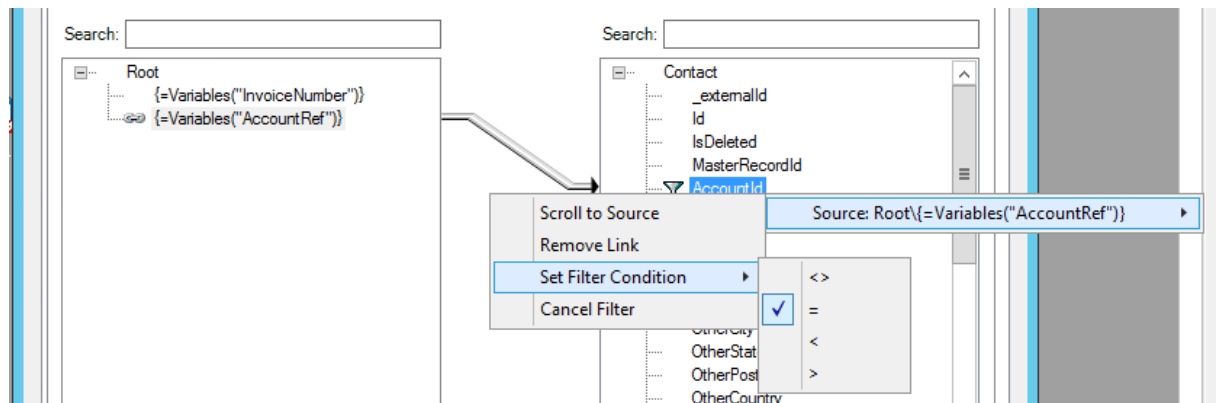
Using Filters with Objects

When using **get** and **upsert** operations, you can make use of filters to limit the data involved. At runtime, these fields are treated as “where clauses”. Available filter conditions are:

- <> — Not equal to
- = — Equal to
- < — Less than
- > — Greater than

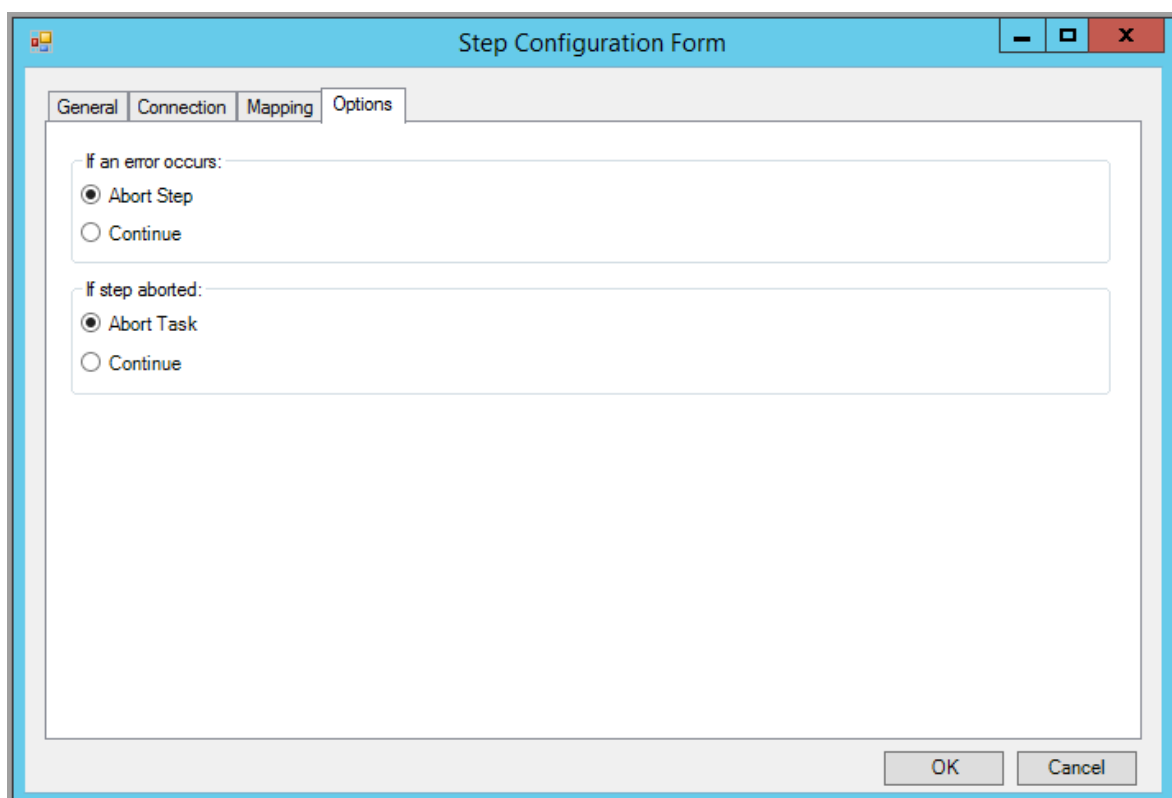
The presence of multiple filter fields implies that all conditions must be met.

Nominated fields must be mapped to be included in the output XML. You add filter conditions by right-clicking the output XML field, and selecting **Set as Filter**. By default, all filters are set to “equal to”. To change the filter condition, right-click the field again, and select **Set Filter Condition**. Change the filter condition as required.



About the Options Tab

The **Options** tab allows you to define how errors in this step are handled at task runtime.










If an error occurs, you can decide whether the step should **Continue** processing, or terminate the step immediately (**Abort Step**).







If the step is aborted, you can choose to **Continue** processing onto the next step in the task, or terminate the whole task immediately (**Abort Task**).

Working with Other Steps

Consuming XML from Other Steps

Icon	Tool Name	Tool Category
	Import Flat File	Input
	Import XML Document	Input
	Convert Recordset to XML	Format
	Transform Data	Format
	Call Task	Execute
	Web Service Connector	Data Connectors
	Salesforce Connector	SKIT

Exposing XML to Other Steps

Icon	Tool Name	Tool Category
	Convert XML to Recordset	Format
	Transform Data	Format
	Save File	Output
	Call Task	Execute
	Web Service Connector	Data Connectors
	Salesforce Connector	SKIT

Supported Objects

Provision of supported objects and operations are handled independently to the tool pack. This means subsequent changes to the API's metadata can be made available to instances of the **Salesforce Connector** tool without the need to reinstall or upgrade. At the time of writing, the following objects are supported:



NOTE:

Should you add any new, custom objects to your Salesforce instance, it is recommended you enable [Update objects and operations](#) in the **Global Tool Settings** dialog box to refresh the supported objects.

[A](#) [B](#) [C](#) [D](#) [E](#) [F](#) [G](#) [H](#) [I](#) [J](#) [K](#) [L](#) [M](#) [N](#) [O](#) [P](#) [Q](#) [R](#) [S](#) [T](#) [U](#) [V](#) [W](#) [X](#) [Y](#) [Z](#)

A

Account

AccountContactRole

AccountFeed

AccountHistory

AccountPartner

AccountShare

AdditionalNumber

AggregateResult

ApexClass

ApexComponent

ApexLog

ApexPage

ApexTestQueueItem

ApexTestResult

ApexTrigger

Approval

Asset

AssetFeed

AssetHistory

AssignmentRule

AsyncApexJob

Attachment

B

BrandTemplate

BusinessHours

BusinessProcess

C

CallCenter

Campaign

CampaignFeed

CampaignHistory

CampaignMember

CampaignMemberStatus

CampaignShare

Case

CaseComment

CaseContactRole

CaseFeed

CaseHistory

CaseShare

CaseSolution

CaseStatus

CaseTeamMember

CaseTeamRole

CaseTeamTemplate

CaseTeamTemplateMember

CaseTeamTemplateRecord

CategoryData

CategoryNode

ChatterActivity

ClientBrowser

CollaborationGroup

CollaborationGroupFeed

CollaborationGroupMember

CollaborationGroupMemberRequest

CollaborationInvitation

Community

Contact

ContactFeed

ContactHistory

ContactShare

ContentDocument

ContentDocumentFeed

ContentDocumentHistory

ContentDocumentLink

ContentVersion

ContentVersionHistory

ContentWorkspace

ContentWorkspaceDoc

Contract

ContractContactRole

ContractFeed

ContractHistory

ContractStatus

CronTrigger

CustomConsoleComponent

D

Dashboard

DashboardComponent

DashboardComponentFeed

DashboardFeed

Document

DocumentAttachmentMap

Domain

DomainSite

E

EmailMessage

EmailServicesAddress

EmailServicesFunction

EmailStatus

EmailTemplate

EntitySubscription

Event

EventFeed

EventRelation

ExternalDataSource

ExternalDataUserAuth

F

FeedComment

FeedItem

FeedLike

FeedTrackedChange

FieldPermissions

FiscalYearSettings

Folder

ForecastShare

G

Group

GroupMember

H

Holiday

I

Idea

IdeaComment

L

Lead

LeadFeed

LeadHistory

LeadShare

LeadStatus

LoginHistory

LoginIp

M

MailmergeTemplate

N

Name

NewsFeed

Note

NoteAndAttachment

O

ObjectPermissions

OpenActivity

Opportunity

OpportunityCompetitor

OpportunityContactRole

OpportunityFeed

OpportunityFieldHistory

OpportunityHistory

OpportunityLineItem

OpportunityPartner

OpportunityShare

OpportunityStage

Organization

OrgWideEmailAddress

P

Partner

PartnerRole

Period

PermissionSet

PermissionSetAssignment

Pricebook2

Pricebook2History

PricebookEntry

ProcessDefinition

ProcessInstance

ProcessInstanceHistory

ProcessInstanceStep

ProcessInstanceWorkitem

ProcessNode

Product2

Product2Feed

Profile

PushTopic

Q

QueueSubject

R

RecordType

Report

ReportFeed

S

Scontrol

SelfServiceUser

SetupAuditTrail

SetupEntityAccess

Site

SiteFeed

SiteHistory

Solution

SolutionFeed

SolutionHistory

SolutionStatus

StaticResource

T

Task

TaskFeed

TaskPriority

TaskStatus

Topic

U

User

UserFeed

UserLicense

UserPreference

UserProfileFeed

UserRecordAccess

UserRole

UserShare

V

Vote

W

WebLink

Tool Output

The **Salesforce Connector** tool outputs two documents:

- ▶ `OutputData`
- ▶ `ErrorData`

Both outputs can be:

- ▶ Used directly by subsequent tool steps which consume XML data inputs. If required, use the **Convert XML to Recordset** tool to convert the XML to a BPA Platform recordset first
- ▶ Accessed using the output (`OutputXmlString`) or error (`ErrorXmlString`) objects for response data present in the **Task Browser**.

Runtime

The incoming XML is translated into the XML format for the object and operation selected in the configuration. The data for the linked fields is brought across into the output XML — only those fields that were linked are brought across. The XML is passed to the connector tool, which then:

- ▶ Processes the data
- ▶ Performs the operation requested
- ▶ Sends back an XML document containing the response

The XML can then be used by other BPA Platform steps.

Error Handling

Errors are written to the BPA Platform Event Log (**Manage > Event Log**). You define how errors are handled in the **Options** tab of the **Salesforce Connector** tool.

Reasons for the errors could include:

- ▶ Web service connection errors

- ▶ User privilege errors
- ▶ Errors from the Salesforce API
- ▶ Warnings and messages from the SalesforceAPI

Salesforce Accounts Security

The Salesforce API is the only method used to connect to a Salesforce database, and perform read / write tasks — all security present in the API is used.

Want to learn more?

Discover how Codeless Platforms can help your business by improving performance, boosting efficiency and cutting costs.



+44 (0) 330 99 88 700



enquiries@codelessplatforms.com



www.codelessplatforms.com

