Informatica

Informatica® PowerExchange for Salesforce
10.5.9

User Guide

Informatica PowerExchange for Salesforce User Guide
10.5.9
November 2025

The information in this documentation is subject to change without notice. If you find any problems in this documentation, report them to us at infa_documentation@informatica.com.

Publication Date: 2025-09-03

Table of Contents

Preface 5

Informatica Resources 5

- Informatica Network 5
- Informatica Knowledge Base 5
- Informatica Documentation 5
- Informatica Product Availability Matrices 6
- Informatica Velocity 6
- Informatica Marketplace 6
- Informatica Global Customer Support 6

Chapter 1: Introduction to PowerExchange for Salesforce 7

PowerExchange for Salesforce Overview 7

Example of Data Migration from Salesforce 8

Example of Updating Real-time Data to Salesforce 8

Chapter 2: PowerExchange for Salesforce Configuration 9

PowerExchange for Salesforce Configuration Overview 9

- Configuring HTTP Proxy Options at Design Time 9
- Configuring HTTP Proxy Options at Run Time 10

Chapter 3: Salesforce Connections 11

Salesforce Connection Overview 11

Salesforce Connection Properties 11

Creating a Salesforce Connection in the Administrator Tool 13

Creating a Salesforce Connection in the Developer Tool 13

Chapter 4: Salesforce Data Objects 15

Salesforce Data Objects Overview 15

Standard and Custom Salesforce Objects 15

Related Objects 16

- Rules and Guidelines for Related Objects 16

Salesforce Data Object Views 16

Salesforce Data Object Overview Properties 17

Salesforce Data Object Read Operation Properties 17

- Source Properties of the Data Object Read Operation 17
- Output Properties of the Data Object Read Operation 19

Salesforce Data Object Write Operation Properties 21

- Input Properties of the Data Object Write Operation 22
- Target Properties of the Data Object Write Operation 24

Importing a Salesforce Data Object 25

Creating a Salesforce Data Object Read or Write Operation. 26

Chapter 5: Salesforce Mappings. 27

Salesforce Mappings Overview. 27

Salesforce Mapping Read Example. 27

Salesforce Mapping Write Example. 28

Chapter 6: PowerExchange for Salesforce Dynamic Mappings. 29

Salesforce Dynamic Mapping Overview. 29

Developing and Running Dynamic Mappings. 30

Salesforce Dynamic Mapping Example. 30

Chapter 7: Salesforce Run Time Processing. 32

Salesforce Run-time Processing Overview. 32

Parameterization of Salesforce Sources. 32

Parameterization of Salesforce Targets. 33

Filtering Source Data by Using the SOQL Filter Condition. 33

Capturing Deleted and Archived Salesforce Records. 33

Enable Bulk Query. 34

Use SFDC Bulk API. 34

Configuring the Upsert Target Operation. 34

Configuring the Maximum Batch Size. 35

Handling Null Values in Update and Upsert Operations. 35

Override an External ID with an idLookup for Upserts. 35

Appendix A: Data Type Reference. 36

Data Type Reference Overview. 36

Salesforce Data Types and Transformation Data Types. 36

Index. 38

Preface

Use the *Informatica® PowerExchange® for Salesforce User Guide* to learn how to read from or write to Salesforce by using the Developer tool. Learn to create a Salesforce connection and develop and run mappings in an Informatica domain.

Informatica Resources

Informatica provides you with a range of product resources through the Informatica Network and other online portals. Use the resources to get the most from your Informatica products and solutions and to learn from other Informatica users and subject matter experts.

Informatica Network

The Informatica Network is the gateway to many resources, including the Informatica Knowledge Base and Informatica Global Customer Support. To enter the Informatica Network, visit https://network.informatica.com.

As an Informatica Network member, you have the following options:

- Search the Knowledge Base for product resources.
- View product availability information.
- Create and review your support cases.
- Find your local Informatica User Group Network and collaborate with your peers.

Informatica Knowledge Base

Use the Informatica Knowledge Base to find product resources such as how-to articles, best practices, video tutorials, and answers to frequently asked questions.

To search the Knowledge Base, visit https://search.informatica.com. If you have questions, comments, or ideas about the Knowledge Base, contact the Informatica Knowledge Base team at KB_Feedback@informatica.com.

Informatica Documentation

Use the Informatica Documentation Portal to explore an extensive library of documentation for current and recent product releases. To explore the Documentation Portal, visit https://docs.informatica.com.

If you have questions, comments, or ideas about the product documentation, contact the Informatica Documentation team at infa_documentation@informatica.com.

Informatica Product Availability Matrices

Product Availability Matrices (PAMs) indicate the versions of the operating systems, databases, and types of data sources and targets that a product release supports. You can browse the Informatica PAMs at https://network.informatica.com/community/informatica-network/product-availability-matrices.

Informatica Velocity

Informatica Velocity is a collection of tips and best practices developed by Informatica Professional Services and based on real-world experiences from hundreds of data management projects. Informatica Velocity represents the collective knowledge of Informatica consultants who work with organizations around the world to plan, develop, deploy, and maintain successful data management solutions.

You can find Informatica Velocity resources at http://velocity.informatica.com. If you have questions, comments, or ideas about Informatica Velocity, contact Informatica Professional Services at ips@informatica.com.

Informatica Marketplace

The Informatica Marketplace is a forum where you can find solutions that extend and enhance your Informatica implementations. Leverage any of the hundreds of solutions from Informatica developers and partners on the Marketplace to improve your productivity and speed up time to implementation on your projects. You can find the Informatica Marketplace at https://marketplace.informatica.com.

Informatica Global Customer Support

You can contact a Global Support Center through the Informatica Network or by telephone.

To find online support resources on the Informatica Network, click **Contact Support** in the Informatica Intelligent Cloud Services Help menu to go to the **Cloud Support** page. The **Cloud Support** page includes system status information and community discussions. Log in to Informatica Network and click **Need Help** to find additional resources and to contact Informatica Global Customer Support through email.

The telephone numbers for Informatica Global Customer Support are available from the Informatica web site at https://www.informatica.com/services-and-training/support-services/contact-us.html.

CHAPTER 1

Introduction to PowerExchange for Salesforce

This chapter includes the following topics:

- PowerExchange for Salesforce Overview, 7
- Example of Data Migration from Salesforce, 8
- Example of Updating Real-time Data to Salesforce, 8

PowerExchange for Salesforce Overview

PowerExchange for Salesforce provides connectivity between Informatica Developer and Salesforce. You can use PowerExchange for Salesforce to read data from and write data to Salesforce. You can add a Salesforce data object operation as a source or a target in a mapping and run the mapping to read or write data.

Salesforce sources and targets represent objects in the Salesforce object model. Salesforce objects are tables that correspond to tabs and other user interface elements on the Salesforce website. For example, the Account object contains the information that appears in fields on the Salesforce Account tab. You can view, create, update, and delete data in Salesforce objects.

PowerExchange for Salesforce uses the Salesforce security model to enforce data access controls. You can access data based on the Salesforce organization associated with the user account you use to connect to Salesforce. Your access to data also depends on the user privileges and the field-level and row-level permissions associated with the login.

PowerExchange for Salesforce uses the Simple Object Access Protocol API (SOAP API) to read or write a small volume of data in near real-time mode. PowerExchange for Salesforce uses the Salesforce Bulk API to read large amounts of data from Salesforce sources or write large amounts of data to Salesforce targets. PowerExchange for Salesforce generates a Salesforce Object Query Language (SOQL) query to read data from Salesforce objects. SOQL is a derivative of SQL.

You can run a profile on a Salesforce data object. A Salesforce data object profile discovers information about the column data and metadata in the Salesforce data source.

PowerExchange for Salesforce is listed under the Cloud connection category in the Developer tool and the Administrator tool.

To learn about the supported version for Salesforce API, review the Product Availability Matrix. You can find the Product Availability Matrix on the Informatica Network: Product Availaibility Matrix.

Example of Data Migration from Salesforce

Your organization needs to migrate sales opportunity information from a Salesforce system that the sales team uses to a relational data source that the executive management team uses. You can create a data object in the Model repository and import the Opportunity object. The executive management team can reconcile and analyze the data written to the relational data object.

Example of Updating Real-time Data to Salesforce

Your organization needs to update real-time sales order processing status from an enterprise resource planning (ERP) system that the logistics team uses to a Salesforce system that was used to create the order. You can create a data object and specify the update strategy. You can then create a mapping that reads shipping details from the ERP system and writes those records to the Salesforce data objects. Sales managers can use the updated information to track sales orders.

CHAPTER 2

PowerExchange for Salesforce Configuration

This chapter includes the following topic:

- PowerExchange for Salesforce Configuration Overview, 9

PowerExchange for Salesforce Configuration Overview

Configure PowerExchange for Salesforce before you can extract data from or load data to a Salesforce database.

Configuring HTTP Proxy Options at Design Time

If your organization uses a proxy server to access the Internet, you can configure the HTTP proxy server authentication settings from the `developerCore.ini` file.

Perform the following tasks to configure the HTTP Proxy Options from the Developer tool:

1. Enable the proxy server settings from your web browser.
2. Access the `developerCore.ini` file at the following location:

   ```
   <Informatica Installation Location>\clients\DeveloperClient
   ```
3. Add the HTTP Proxy options to the `developerCore.ini` file.
   The following table describes the properties that you must add to the `developerCore.ini` file:

Property	Description
-Dsfdc.http.proxyHost=	Name of the HTTP proxy server.
-Dsfdc.http.proxyPort=	Port number of the HTTP proxy server.

Property	Description
-Dsfdc.http.proxyUser=	Authenticated user name for the HTTP proxy server. This property is required if the proxy server requires authentication.
-Dsfdc.http.proxyPassword=	Password for the authenticated user. This property is required if the proxy server requires authentication. **Note:** The password is not encrypted.

When you create a connection in the Administrator tool, the test connection might fail if you connect to Salesforce through the proxy server. In this case, you must configure the `ACJVMCommandLineOptions` environment variable with the proxy details and restart the service.

For example, `export ACJVMCommandLineOptions="-Dsfdc.http.proxyHost=proxyhostname -Dsfdc.http.proxyPort=proxyportnumber -Dsfdc.http.proxyUser=username -Dsfdc.http.proxyPassword=password"`

Configuring HTTP Proxy Options at Run Time

If your organization uses a proxy server to access the internet, you must configure the HTTP proxy server authentication settings for the Data Integration Service.

1. Open the Administrator tool.
2. Click the **Administration** tab, and then select the **Data Integration Service**.
3. Click the **Processes** tab.
4. Configure the following properties in the custom properties:

JVM Option	Property Name	Description
JVMOption1	-Dsfdc.http.proxyHost	Name of the HTTP proxy server.
JVMOption2	-Dsfdc.http.proxyPort	Port number of the HTTP proxy server. Default is 8080.
JVMOption3	-Dsfdc.http.proxyUser	Authenticated user name for the HTTP proxy server. This property is required if the proxy server requires authentication.
JVMOption4	-Dsfdc.http.proxyPassword	Password for the authenticated user. This property is required if the proxy server requires authentication. **Note:** The password is not encrypted.

CHAPTER 3

Salesforce Connections

This chapter includes the following topics:

- Salesforce Connection Overview, 11
- Salesforce Connection Properties, 11
- Creating a Salesforce Connection in the Administrator Tool, 13
- Creating a Salesforce Connection in the Developer Tool, 13

Salesforce Connection Overview

Use a Salesforce connection to access objects in a Salesforce application.

Create a connection to import Salesforce metadata to create data objects, preview data, and run mappings.

You can create a Salesforce connection in the Developer tool, the Administrator tool, and through infacmd isp.

Salesforce Connection Properties

Use a Salesforce connection to connect to a Salesforce object. The Salesforce connection is an application connection type. You can create and manage a Salesforce connection in the Administrator tool or the Developer tool.

Note: The order of the connection properties might vary depending on the tool where you view them.

The following table describes Salesforce connection properties:

Property	Description
Name	Name of the connection. The name is not case sensitive and must be unique within the domain. It cannot exceed 128 characters, contain spaces, or contain the following special characters: `~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /`
ID	String that the Data Integration Service uses to identify the connection. The ID is not case sensitive. It must be 255 characters or less and must be unique in the domain. You cannot change this property after you create the connection. Default value is the connection name.
Description	The description of the connection. The description cannot exceed 765 characters.
Location	The Informatica domain where you want to create the connection.
Type	The connection type. You can select the Standard connection type or OAuth connection type.
User Name	Applicable for Standard connection type. Salesforce user name.
User Password	Applicable for Standard connection type. Password for the Salesforce user name. To access Salesforce outside the trusted network of your organization, you must append a security token to your password to log in to the API or a desktop client. To receive or reset your security token, log in to Salesforce and click **Setup** > **My Personal Information** > **Reset My Security Token**. Password is case sensitive.
Service URL	URL of the Salesforce service you want to access. For example, `https://login.salesforce.com/services/Soap/u/<version>` In a test or development environment, you might want to access the Salesforce Sandbox testing environment. For more information about the Salesforce Sandbox, see the Salesforce documentation.
Refresh Token	Applicable for OAuth connection type. Refresh Token of Salesforce.
Consumer Key	Applicable for OAuth connection type. The Consumer Key obtained from Salesforce, required to generate the Refresh Token. For more information about how to generate the Consumer Key, see the Salesforce documentation.
Consumer Secret	Applicable for OAuth connection type. The Consumer Secret obtained from Salesforce, required to generate the Refresh Token. For more information about how to generate the Consumer Secret, see the Salesforce documentation.

Creating a Salesforce Connection in the Administrator Tool

Create a connection before you import Salesforce data objects, preview data, or run mappings. When you create a Salesforce connection, you enter information such as a connection ID and the URL of the Salesforce service you want to access.

1. In the Administrator tool, click the **Domain** tab.
2. Click the **Connections** view.
3. In the Navigator, select the domain.
4. In the Navigator, click **Actions** > **New** > **Connection**.

   The **New Connection** dialog box appears.
5. In the **New Connection** dialog box, select **Cloud** > **Salesforce**, and then click **OK**.

   The **New Connection** wizard appears.
6. Enter a connection name.
7. Enter an ID for the connection.
8. Optionally, enter a connection description.
9. Enter the connection properties.
10. Click **Test Connection** to verify that you can connect to Salesforce.
11. Click **Finish**.

Creating a Salesforce Connection in the Developer Tool

Use the Salesforce connection to connect to Salesforce from the Developer tool to import Salesforce data objects, preview data, or run mappings.

Perform the following steps to create a Salesforce connection in the Developer tool:

1. Click **Window** > **Preferences**.
2. Select **Informatica** > **Connections**.
3. Expand the domain.
4. Select **Cloud** > **Salesforce** and click **Add**.
5. When you set up a Salesforce connection, you can select the connection type as **Standard** or **OAuth** .The following table lists the properties for a standard connection:

Connection Property	Description
Name	Name of the Salesforce connection.
ID	The Data Integration Service uses the ID to identify the connection.

Connection Property	Description
Description	Optional. The description of the connection.
Location	Informatica domain where you want to create the connection.
Type	Connection type. Select Salesforce.
User Name	User name for the Salesforce account.
User Password	Password for the Salesforce account.
Service URL	URL of the Salesforce service. For example, `https://login.salesforce.com/services/Soap/u/(version)`

The following table lists the properties for an OAuth connection:

Connection Property	Description
Name	Name of the OAuth connection.
ID	The Data Integration Service uses the ID to identify the connection.
Description	Optional. The description of the connection.
Location	Informatica domain where you want to create the connection.
Type	Connection type. Select OAuth.
Service URL	URL of the Salesforce service. For example, `https://login.salesforce.com/services/Soap/u/(version)`
Refresh Token	The Refresh Token of Salesforce.
Consumer Key	The Consumer Key obtained from Salesforce, required to generate the Refresh Token. For more information about how to generate the Consumer Key, see the Salesforce documentation.
Consumer Secret	The Consumer Secret obtained from Salesforce, required to generate the Refresh Token. For more information about how to generate the Consumer Secret, see the Salesforce documentation.

6. Click **Test Connection** to verify that you can connect to Salesforce.
7. Click **Finish**.

CHAPTER 4

Salesforce Data Objects

This chapter includes the following topics:

- Salesforce Data Objects Overview, 15
- Standard and Custom Salesforce Objects, 15
- Related Objects, 16
- Salesforce Data Object Views, 16
- Salesforce Data Object Overview Properties, 17
- Salesforce Data Object Read Operation Properties, 17
- Salesforce Data Object Write Operation Properties, 21
- Importing a Salesforce Data Object, 25
- Creating a Salesforce Data Object Read or Write Operation, 26

Salesforce Data Objects Overview

A Salesforce data object is a physical data object that uses a Salesforce object as a source and a target. A Salesforce data object is a representation of data that is based on a Salesforce object.

Import a Salesforce object into the Developer tool to create a Salesforce data object. After you create a data object, create a data object read or write operation. You can use the data object read operation as a source and the data object write operation as a target in a mapping.

Standard and Custom Salesforce Objects

Use the Developer tool to import Salesforce objects and create a Salesforce data object. You can import both standard and custom Salesforce objects.

Standard object types are objects packaged within Salesforce, such as Account, AccountPartner, and Opportunity.

Custom object types extend the Salesforce data for an organization by defining data entities that are unique to the organization. Salesforce administrators can define custom fields for both standard and custom objects.

When you import a Salesforce object, use a Salesforce login to connect to the Salesforce service. The Developer tool generates a list of objects that are available for import.

Related Objects

You might need to read data from more than one object at a time. The Data Integration Service generates relationship queries through SOQL to read data from related objects.

For example, you can read all accounts created by Tom Smith and the contacts associated with those accounts. You can use PowerExchange for Salesforce to create parent-to-child relationships that connect the objects.
Parent-to-child relationships exist between many types of objects. For example, Account is a parent of Contact, Assets, and Cases.

Use PowerExchange for Salesforce to read related objects. Each object can have one related object. For example, you can create a data object called Account Details. Select Account as the parent object, and either Contact or Opportunity as the child object. The relationship persists while creating a Salesforce data object read operation from the Salesforce data object called Account Details.

Rules and Guidelines for Related Objects

Consider the following rules and guidelines when you import related objects in a Salesforce data object:

- You must select a parent object to create a data object that has related objects.
- You cannot import multiple parent objects in a single data object.
- You can select one related object for each parent object.
- You cannot read data from a related object while using Bulk API. You can read data from one parent object.
- The Relationship dialog box does not display all related entities of the Salesforce object.

Salesforce Data Object Views

The Salesforce data object contains views to edit the object name and the properties.

After you create a Salesforce data object, you can change the data object properties in the following data object views:

- **Overview** view. Edit the Salesforce data object name, description, and object.
- **Data Object Operation** view. View and edit the properties that the Data Integration Service uses when it reads data from or writes data to a Salesforce data object.

When you create a mapping that uses a Salesforce source, you can view the data object read properties in the **Properties** view.

When you create a mapping that uses a Salesforce target, you can view the data object write properties in the **Properties** view.

Salesforce Data Object Overview Properties

The **Overview** view displays general information about the Salesforce data object and detailed information about the Salesforce object that you imported.

The following table describes the general properties that you configure for a Salesforce data object:

Property	Description
Name	Name of the Salesforce data object.
Description	Description of the Salesforce data object.
Connection	Name of the Salesforce connection.

The following table describes the Salesforce object properties that you can view:

Property	Description
Name	Name of the Salesforce object.
Type	Native data type of the Salesforce object.
Description	Description of the Salesforce object.

Salesforce Data Object Read Operation Properties

The Data Integration Service reads data from a Salesforce object based on the data object read operation. The Developer tool displays the data object read operation properties of the Salesforce data object in the **Data Object Operation** view.

You can view or configure the data object read operation from the source and output properties.

Source properties

Represents data that the Data Integration Service reads from the Salesforce object. Select the source properties to view data such as the name and description of the Salesforce object and the column properties.

Output properties

Represents data that the Data Integration Service passes into the mapping pipeline. Select the output properties to edit the port properties of the data object read operation. You can also set advanced properties, such as row limit and Salesforce bulk API.

Source Properties of the Data Object Read Operation

The source properties are populated based on the Salesforce object that you added when you created a data object. The source properties of the data object read operation include general and column properties that apply to the Salesforce object.

You can view the source properties of the data object read operation from the **General**, **Column**, and **Advanced** tabs.

General Properties

The general properties display the name and description of the data object object read operation.

Column Properties

The column properties display the data types, precision, and scale of the source property in the data object read operation.

The following table describes the source column properties of the data object read operation:

Property	Description
Name	Name of the column
Type	Native data type of the column
Precision	Maximum number of significant digits for numeric data types, or maximum number of characters for string data types. For numeric data types, precision includes scale.
Scale	Maximum number of digits after the decimal point for numeric values
Description	Description of the column
Creatable	Indicates whether the field allows inserts
Updateable	Indicates whether the field allows updates
ExternalID	Salesforce custom fields only. Indicates whether the field is designated as an external ID field. Each Salesforce object can contain a single custom field designated as the external ID field. Salesforce appends custom field names with "__c". For more information about external ID and custom fields, see the Salesforce documentation.
SforceName	Field name in Salesforce
ReferenceTo	Gets referenced object
IDLookup	Specifies a record in an upsert call. The ID field of each object and some Name fields have this property. There are exceptions, so use Salesforce to check for this property in any object that you want to upsert.
Filterable	Indicates whether the field can be used in the FROM or WHERE clause of an SOQL query.
Label	Field label in Salesforce
Access Type	Indicates whether the field has read and write permissions.

Advanced Properties

The advanced properties display the physical name of the Salesforce object.

Output Properties of the Data Object Read Operation

The output properties represent data that the Data Integration Service passes into the mapping pipeline. Select the output properties to edit the port properties of the data object read operation.

The output properties of the data object read operation include general properties that apply to the data object operation. The output properties also include port, source, query, and advanced properties that apply to the Salesforce object.

You can view and change the output properties of the data object read operation from the **General**, **Ports**, **Sources**, **Query**, and **Advanced** tabs.

General Properties

The general properties display the name and description of the data object object read operation.

Ports Properties

The output ports properties display the data types, precision, and scale of the data object read operation.

The following table describes the output ports properties that you configure in the data object read operation:

Property	Description
Name	Name of the port.
Type	Data type of the port.
Precision	Maximum number of significant digits for numeric data types, or maximum number of characters for string data types. For numeric data types, precision includes scale.
Scale	Maximum number of digits after the decimal point for numeric values.
Description	Description of the port.

Sources Properties

The sources properties list the Salesforce objects in the data object read operation.

Query Properties

Use the **Query** tab to specify a filter condition. You can specify a native filter expression or a platform filter expression.

When you use the native expression, you use the standard Salesforce syntax for filter expressions. You can specify a native expression that use AND, OR, or nested conditions. The expression that you enter becomes the WHERE clause in the query used to retrieve records from the source.

To specify a native expression, use the following syntax:

```
<Object Name.field> <Operator> <Value>
```

For example,

```
Account.Name='John'
```

Platform expression uses Informatica transformation language. Use the platform filter expression to select specific records from Salesforce resources based on the filter condition that you specify.

The following table describes the properties that you can specify when you use the platform expression filter:

Property	Description
Expression Type	Type of filter expression that you want to use to filter records. Select **Platform Expression**.
Left Field	Column on which you want to apply the filter condition.
Operator	Simple operators that you can use to filter records. You can select one of the following operators: `=, !=, <, <=, >, >=`
Right Field	Value based on which you want to filter the records. You can also parameterize the value.

Advanced Properties

Use the advanced properties to specify the data object read operation properties to read data from Salesforce objects.

The following table describes the advanced properties that you configure in the data object read operation:

Property	Description
SOQL Filter Condition	Filters Salesforce source records.
CDC Time Limit	Time period, in seconds, that the Data Integration Service reads changed Salesforce data. When you set the CDC Time Limit to a non-zero value, the Data Integration Service performs a full initial read of the source data and then captures changes to the Salesforce data for the time period you specify. Default is 0.
Flush Interval	Interval, in seconds, at which the Data Integration Service captures changed Salesforce data. Default is 300. If you set the CDC Time Limit to a non-zero value, the agent captures changed data from the source every 300 seconds. Otherwise, the Data Integration Service ignores this value.
CDC Start Timestamp	Start date and time for the time period. The Data Integration Service extracts data added or modified after this time. Must be in the format YYYY-MM-DDTHH:MI:SS.SSSZ.
CDC End Timestamp	End date and time for the time period. The Data Integration Service extracts data added or modified before this time. Must be in the format YYYY-MM-DDTHH:MI:SS.SSSZ.
Row Limit	Specifies the maximum number of rows the Data Integration Service processes. Default is 0, which indicates that the Data Integration Services processes all records.
Use QueryAll	Runs a query that returns all rows, which includes active, archived, and deleted rows. Otherwise, the Data Integration Service returns active rows. **Note:** The Use QueryAll property for Bulk API is available for Salesforce API version 39.0 and later.

Property	Description
Use SystemModstamp for CDC	Uses the SystemModstamp as the time stamp for changed records in Salesforce. Otherwise, the Data Integration Service uses the LastModifiedDate time stamp to identify changed records in Salesforce. Default is to use the LastModifiedDate time stamp.
CDC Flush Interval Offset	The number of seconds that you want to offset the CDC flush interval. Captures real-time data that is submitted within the CDC time limit but not committed by Salesforce within the time limit.
Enable Bulk Query	Enable this feature to use the Salesforce Bulk API to read batch files containing large amounts of data. By default, the Data Integration Service uses the SOAP Salesforce API.
Enable PK Chunking	Use to extract from Salesforce sources when you use the Bulk API. Enables primary key chunking to optimize performance while replicating data from large data sets. Salesforce splits the data set into a number of chunks based on the record ID, creates multiple queries to extract data, and combines the result. Salesforce supports primary key chunking for custom objects and certain standard objects. For more information about supported objects for primary key chunking, see the Salesforce documentation.
PK Chunking Size	The number of records in a chunk. Default is 100,000. The maximum value is 250,000. Applicable if you select Enable PK Chunking.
PK Chunking startRow ID	The record ID from which you want to chunk the data set. By default Salesforce applies chunking from the first record. Applicable if you select Enable PK Chunking.
PK Chunking Parent Object	Specify the parent object to enable PK chunking for queries on a shared object. For example, for CaseHistory, specify Case as the parent object. You can perform PK chunking on shared objects only if PowerExchange for Salesforce supports PK chunking on the parent objects. **Note:** PK Chunking Parent Object name is case sensitive. The name must start with an uppercase letter followed by lowercase letters.

Salesforce Data Object Write Operation Properties

The Data Integration Service writes data to a Salesforce object based on the data object write operation. The Developer tool displays the data object write operation properties for the Salesforce data object in the **Data Object Operation** section.

You can view the data object write operation from the Input and Target properties.

Input properties

Represent data that the Data Integration Service reads from an enterprise resource planning (ERP) system or a relational data object. Select the input properties to edit the port properties and specify the advanced properties of the data object write operation.

Target properties

Represent data that the Data Integration Service writes to Salesforce. Select the target properties to view data, such as the name, description, and the relationship of the Salesforce object.

Note: Information about rejected rows in a SOAP writer session is written to the Data Integration Service logs.

Input Properties of the Data Object Write Operation

Input properties represent data that the Data Integration Service reads from an enterprise resource planning (ERP) system or a relational data object. Select the input properties to edit the port properties of the data object write operation. You can also specify advanced data object write operation properties to write data to Salesforce objects.

The input properties of the data object write operation include general properties that apply to the data object write operation. They also include port, source, and advanced properties that apply to the data object write operation.

You can view and change the input properties of the data object write operation from the **General**, **Ports**, **Sources**, and **Advanced** tabs.

General Properties

The general properties list the name and description of the data object write operation.

Ports Properties

The input ports properties list the data types, precision, and scale of the data object write operation.

The following table describes the input ports properties that you must configure in the data object write operation:

Property	Description
Name	Name of the port.
Type	Data type of the port.
Precision	Maximum number of significant digits for numeric data types, or maximum number of characters for string data types. For numeric data types, precision includes scale.
Scale	Maximum number of digits after the decimal point for numeric values.
Description	Description of the port.

Sources Properties

The sources properties list the Salesforce objects in the data object write operation.

Advanced Properties

The advanced properties allow you to specify data object write operation properties to write data to Salesforce objects.

You can configure the following advanced properties in the data object write operation:

Property	Description
Treat Insert as Upsert	Upserts any record flagged as insert. By default, the Data Integration Service treats all records as insert.
Treat Update as Upsert	Upserts any record flagged as update. Select this property when you use the Update Strategy transformation in the mapping to flag records as update.
Max Batch Size	Maximum number of records the Data Integration Service writes to a Salesforce target in one batch. Default is 200 records. Not used in Bulk API target sessions.
Set Fields to NULL	Replaces values in the target with null values from the source. By default, the Data Integration Service does not replace values in a record with null values during an update or upsert operation. The Data Integration Service retains the existing values.
Use SFDC Error File	Generates error log files for a Bulk API target session. You must specify the SFDC error file directory and select the Monitor Bulk Job Until All Batches Processed session properties to generate the error log files. By default, the Data Integration Service does not generate error log files. Not applicable to a Standard API target session.
Use SFDC Success File	Generates success log files for a Bulk API target session. You must specify the SFDC success file directory and select the Monitor Bulk Job Until All Batches Processed session properties to generate the success log files. By default, the Data Integration Service does not generate success log files. Not applicable to a Standard API target session.
SFDC Success File Directory	Mandatory, if you want to create success log file for the Bulk API target. Specify a directory on the machine that hosts the Data Integration Service.
Use Idlookup Field for Upserts	Uses the Salesforce idLookup field to identify target records that need to be upserted. If you do not select this property, use an external ID for the upsert operation. If you do not select this property and do not provide an external ID, the session fails.
Use this ExternalId/ IdLookup field for Updates	The exact name of the external ID or idLookup field to use for updates. By default, the Data Integration Service uses the first external ID or idLookup field in the target. Use this property when you want to use a different field for updates.
Use SFDC Bulk API	Uses the Salesforce Bulk API to load batch files containing large amounts of data to Salesforce targets. By default, the Data Integration Service uses the standard Salesforce API.
Monitor Bulk Job Until All Batches Processed	Monitors a Bulk API target session. When you select this property, the Data Integration Service logs the status of each batch in the session log. If you do not select this property, the Data Integration Service does not generate complete session statistics for the session log.

Property	Description
Override Parallel Concurrency with Serial	Instructs the Salesforce Bulk API to write batches to targets serially. By default, the Bulk API writes batches in parallel.
Disable Bulk Success and Error file creation	Disables the creation of success and error log files for a Bulk API target session. Overrides the Use SFDC Error File and Use SFDC Success File target properties.
Enable Field Truncation Attribute	Allows Salesforce to truncate target data that is larger than the target field. When you select this property, Salesforce truncates overflow data and writes the row to the Salesforce target.
Set prefix for BULK success and error files	Adds the prefix to the BULK success and error files names when created.
Enable Hard Deletes for BULK API	Permanently deletes rows from Salesforce targets in a Bulk API target session.
Set the location of the BULK error files	Mandatory, if you want to create error log file for the Bulk API target. Specify a directory on the machine that hosts the Data Integration Service.
Set the Interval for Polling BULK Job Status	Number of seconds the Data Integration Service waits before polling Salesforce for information about a Bulk API target session. Enter a positive integer. By default, the Data Integration Service polls every 10 seconds.
Assignment Rule Selection	Applicable only to Salesforce Case and Lead target objects using the standard API. Assignment rule to reassign attributes in records when inserting, updating, or upserting records: - None. If you do not want to specify an assignment rule, select None. - Default. Select to use the default assignment rule set for the organization. - Custom. Select to specify and use a custom assignment rule. **Note:** You cannot use assignment rules for Bulk API.
Specify Assignment Rule	Applicable for custom assignment rule. Enter a valid assignment rule.

Target Properties of the Data Object Write Operation

The target properties represent the data that is used to populate the Salesforce data object that you added when you created the data object. The target properties of the data object write operation include general and column properties that apply to the Salesforce objects. You can view the target properties of the data object write operation from the **General**, **Column**, and **Advanced** tabs.

General Properties

The general properties display the name and description of the Salesforce objects.

Column Properties

The column properties display the data types, precision, and scale of the target property in the data object write operation.

You can view the following target column properties of the data object write operation:

Property	Description
Name	Name of the column
Type	Native data type of the column property
Precision	Maximum number of significant digits for numeric data types, or maximum number of characters for string data types. For numeric data types, precision includes scale.
Scale	Maximum number of digits after the decimal point for numeric values
Primary Key	Determines whether the column property is a part of the primary key
Description	Description of the column property

Advanced Properties

The advanced properties displays the physical name of the Salesforce objects.

Importing a Salesforce Data Object

Import a Salesforce data object to read data from a Salesforce object.

1. Select a project or folder in the Object Explorer view.
2. Click **File** > **New** > **Data Object**.
3. Select **Salesforce Data Object** and click **Next**.

   The **New Salesforce Data Object** dialog box appears.
4. Enter a name for the data object.
5. Click **Browse** next to the **Location** option and select the target project or folder.
6. Click **Browse** next to the **Connection** option and select the Salesforce connection from which you want to import the Salesforce object.
7. To add an object, click **Add** next to the **Selected Resource(s)** option.

   The **Add Resource** dialog box appears.
8. Select a Salesforce object. You can search for it or navigate to it.
   - Navigate to the Salesforce object that you want to import and click **OK**.
   - To search for the Salesforce object, enter the name of the Salesforce object you want to add. Click **OK**.
9. If required, add additional objects to the Salesforce data object.

   You can also add objects to a Salesforce data object after you create it.

10. Click **Finish**.

The data object appears under Physical Data Objects in the project or folder in the Object Explorer view.

Creating a Salesforce Data Object Read or Write Operation

You can add a Salesforce data object read or write operation to a mapping or mapplet as a source. You can create the data object read or write operation for one or more Salesforce data objects.

Before you create a Salesforce data object read or write operation, you must create at least one Salesforce data object.

1. Select the data object in the Object Explorer view.
2. Right-click and select **New** > **Data Object Operation**.

   The **Data Object Operation** dialog box appears.
3. Enter a name for the data object read or write operation.
4. Select **Read** or **Write** as the type of data object operation.
5. Click **Add**.

   The **Select Resources** dialog box appears.
6. Select the Salesforce object for which you want to create the data object read or write operation and click **OK**.
7. Click **Finish**

The Developer tool creates the data object read or write operation for the selected data object.

CHAPTER 5

Salesforce Mappings

This chapter includes the following topics:

- Salesforce Mappings Overview, 27
- Salesforce Mapping Read Example, 27
- Salesforce Mapping Write Example, 28

Salesforce Mappings Overview

After you create a Salesforce data object read or write operation, you can develop a mapping.

You can define the following objects in the mapping:

- Salesforce data object read operation as the input to read data from Salesforce metadata.
- Relational, flat file, or any supported data object as the output.
- Relational, flat file, or any supported data object as the input.
- Salesforce data object write operation as the output to write data to Salesforce data objects.

Validate and run the mapping to read data from Salesforce sources, and write to a Salesforce object.

Salesforce Mapping Read Example

Your organization needs to migrate real-time sales opportunity information from a Salesforce system that is used by the sales team to a relational data source that is used internally by the executive sales management team.

Create a mapping that reads opportunity information in real time and writes those records to a table.

You can use the following objects in a Salesforce mapping:

Mapping Input

The mapping source is a Salesforce data object that contains the Opportunity object. Add the Opportunity object to the physical data object.

Create a data object read operation. Add the data object read operation to the mapping.

Mapping Output

Add a relational data object to the mapping as an output.

After you run the mapping, the Data Integration Service writes the extracted opportunity information to the target table. Sales managers can use the information to track sales opportunities.

Salesforce Mapping Write Example

Your organization needs to update real-time sales order processing status from an ERP system that is used by the logistics team to a Salesforce system that was used to create the order.

Create a mapping that reads real-time sales order processing status from the ERP system, and writes those records to Salesforce.

You can use the following objects in a Salesforce mapping:

Mapping Input

Add a relational data object to the mapping as an input.

Mapping Output

Add a Salesforce data object write operation to the mapping as an output.

The mapping target is a Salesforce data object that contains the Order object. Add the Order object to the physical data object.

Create a data object write operation and specify the update strategy in the data object write operation. Add the data object write operation to the mapping.

After every mapping run, the Data Integration Service writes the extracted order status information to the target table. Sales managers can use the updated information to track sales orders.

CHAPTER 6

PowerExchange for Salesforce Dynamic Mappings

This chapter includes the following topics:

- Salesforce Dynamic Mapping Overview, 29
- Developing and Running Dynamic Mappings, 30
- Salesforce Dynamic Mapping Example, 30

Salesforce Dynamic Mapping Overview

You can use Salesforce data objects as dynamic sources and targets in a mapping in the native environment.

Use the Salesforce dynamic mapping to accommodate changes to source, target, and transformation logics at run time. You can use a Salesforce dynamic mapping to manage frequent schema or metadata changes or to reuse the mapping logic for data sources with different schemas. Configure rules, parameters, and general transformation properties to create the dynamic mapping.

If the data source for a source or target changes, you can configure a mapping to dynamically get metadata changes at runtime. If a source changes, you can configure the Read transformation to accommodate changes. If a target changes, you can configure the Write transformation accommodate target changes.

You do not need to manually synchronize the data object and update each transformation before you run the mapping again. The Data Integration Service dynamically determines transformation ports, transformation logic in the ports, and the port links within the mapping.

You can select one of the following options to enable dynamic mapping:

- In the **Data Object** tab of the data object read or write operation, select the **At runtime, get data object columns from data source** option when you create a mapping.
  When you enable the dynamic mapping using this option, you can refresh the source and target schemas at runtime.

  You cannot refresh the schema for Salesforce lookups at runtime.
- In the **Ports** tab of the data object write operation, select the value of the **Columns defined by** property as **Mapping Flow** when you configure the data object write operation properties.

For information about dynamic mappings, see the *Informatica Developer Mapping Guide*.

Developing and Running Dynamic Mappings

Perform the following tasks to develop and run a dynamic mapping to read from or write to Salesforce. The tasks and the order in which you perform the tasks depend on the mapping scenario and the transformations that you plan to use in the mapping.

1. Create a Salesforce mapping and add the Salesforce objects.
2. Configure a Salesforce dynamic source for the Read or Lookup transformation to get metadata changes from the Salesforce source at run time. Select the Salesforce source object and perform one of the following tasks based on your requirement:
   - Use a parameter as a source for a dynamic mapping source object.
   - Configure data sources for source objects in mappings to get metadata changes at run time. To dynamically get columns from the data source file at run time, select **At run time, get data object columns from the data source**.
3. Create dynamic ports in transformations and link ports.
4. Define input rules for dynamic ports to determine which generated ports to create.

   Note: Do not add a lookup condition for dynamic ports in a dynamic mapping.
5. Configure a Write transformation to write to a Salesforce dynamic target. Select the Salesforce target object and perform one of the following tasks based on your requirement:
   - Use a parameter as the data object for the transformation and then change the parameter at run time.
   - To dynamically get data object columns from the data source at run-time, enable the option **At run time, get data object columns from the data source**.
   - Define target object columns by mapping flow to enable upstream mapping objects to update the incoming ports for the Write transformation.
     To do this, select **Columns defined by: Mapping flow** in the **Ports** tab of the Properties view.
6. Create and configure a run-time link to determine which ports to link at run time.
7. Validate and run the mapping.

Salesforce Dynamic Mapping Example

Your organization has metadata that changes frequently and you need to incorporate all the updated metadata in a short span of time. Create a dynamic mapping where you can refresh the source schema dynamically to fetch the updated metadata. Add all the dynamic ports to the target to override the metadata of the existing ports.

1. Import the Salesforce read and write data objects.
2. Select a project or folder in the **Object Explorer** view.
3. Click **File** > **New** > **Mapping**.
   The **Mapping** dialog box appears.
4. Enter the name of the mapping in the **Name** field.
5. Click **Finish**.
6. Drag the data object into a mapping.
   The **Salesforce Data Object Access** dialog box appears.

7. Select the **Read** option and click **OK**.
8. In the **Data Object** tab, select the **At runtime, get data object columns from the data source** check box.
9. Drag the data object into a mapping.
   The **Salesforce Data Object Acess** dialog box appears.
10. Select the **Write** option and click **OK**.
11. In the **Ports** tab, select the value of the **Columns defined by** as **Mapping Flow**.
12. Select all the source incoming ports and add the ports to the target.
13. Save and run the mapping.

CHAPTER 7

Salesforce Run Time Processing

This chapter includes the following topics:

- Salesforce Run-time Processing Overview, 32
- Parameterization of Salesforce Sources, 32
- Parameterization of Salesforce Targets, 33
- Filtering Source Data by Using the SOQL Filter Condition, 33
- Capturing Deleted and Archived Salesforce Records, 33
- Enable Bulk Query, 34
- Use SFDC Bulk API, 34
- Configuring the Upsert Target Operation, 34
- Configuring the Maximum Batch Size, 35
- Handling Null Values in Update and Upsert Operations, 35
- Override an External ID with an idLookup for Upserts, 35

Salesforce Run-time Processing Overview

When you develop a Salesforce mapping, you define the data object operation read or write properties. The data object read operation determines how the Data Integration Service reads data from Salesforce sources, and the data object write operation determines how the Data Integration Service writes data to Salesforce targets.

Parameterization of Salesforce Sources

You can parameterize the Salesforce connection and data object read operation properties to override the mapping properties at run time.

You can parameterize the following read operation properties for a Salesforce source:

1. SOQL Filter Condition
2. CDC Start Timestamp
3. CDC End Timestamp
4. PK Chunking Size

5. PK Chunking startRow ID
6. PK Chunking Parent Object

Parameterization of Salesforce Targets

You can parameterize the Salesforce connection and data object write operation properties to override the mapping properties at run time.

You can parameterize the following write operation properties for a Salesforce target:

1. Set prefix for BULK success and error files
2. SFDC Success File Directory
3. Set the location of the BULK error files
4. Specify Assignment Rule

Filtering Source Data by Using the SOQL Filter Condition

When you configure a mapping that reads data from a Salesforce source, you can enter a filter condition to filter records read from the source. When you enter a filter condition, the Data Integration Service adds the WHERE clause to the SOQL query and generates an SOQL query based on the objects and fields included in the Salesforce source.

To filter records from a Salesforce source, set the SOQL filter condition in the data object read operation. For example, enter the following filter condition to read records from the Salesforce Account object that were created before October 30, 2012:

```
CreatedDate < '2012-10-30T00:00:00.000Z'
```

Enter a filter condition based on the SOQL syntax defined in the Salesforce documentation. The Salesforce API validates the SOQL syntax at run time. If you enter a filter condition that is not valid, the mapping fails.

Capturing Deleted and Archived Salesforce Records

The Data Integration Service can capture active, deleted, and archived records from a Salesforce object. By default, mappings do not capture deleted and archived records.

To capture deleted and archived records, configure the Use queryAll data object read operation property.

Enable Bulk Query

The Data Integration Service can read data from Salesforce sources using the Salesforce Bulk API. Use the Bulk API to read large amounts of data from Salesforce while generating a minimal number of API calls.

With the Bulk API, each batch of data can contain up to approximately 1 GB of data in CSV format. When the Data Integration Service creates a batch, it adds any required characters to properly format the data, such as adding quotation marks around text.

You can also monitor the progress of batches in the log file.

To configure a mapping to use the Salesforce Bulk API, select the Enable Bulk Query data object operation read property.

Use SFDC Bulk API

The Data Integration Service can use the Salesforce Bulk API to write data to Salesforce objects. Use the Bulk API to write large amounts of data to Salesforce with a minimal number of API calls.

With a Bulk API write, each batch of data can contain up to 10,000 records or one million characters of data in CSV format. When the Data Integration Service creates a batch, it adds required characters such as, quotation marks around text, to format the data.

You can configure a Bulk API target session to load batches serially or in parallel. By default, the data load is in parallel mode, but you can override the data load to serial mode. You can also monitor the progress of batches in the session log.

To configure a session to use the Bulk API for Salesforce targets, select the Use SFDC Bulk API session property. When you select this property, the Data Integration Service ignores the Max Batch Size session property.

Configuring the Upsert Target Operation

The Salesforce upsert operation creates a new record or updates an existing record in a Salesforce object. You must provide one of the following types of fields to upsert records to a Salesforce object:

External ID field

You can use a custom Salesforce field to uniquely identify each record in a Salesforce object. You can create a custom external ID field for each object in Salesforce. You can view the properties of a Salesforce object to check whether the object includes an external ID field.

idLookup field

You can use a Salesforce idLookup field to identify each record in a Salesforce object. Salesforce creates idLookup fields for each standard Salesforce object. For example, the Email field is an idLookup field for the Contact object. Custom Salesforce objects do not contain an idLookup field. For more information about idLookup fields, see the Salesforce documentation.

A Salesforce target object might have multiple external ID or idLookup fields. By default, the Data Integration Service uses the first external ID or idLookup field it encounters. However, you can specify the external ID or idLookup field to use for the upsert operation in the run-time properties.

To configure the upsert operation to write to a Salesforce object, perform the following steps:

1. Map the external ID or idLookup field from the source to the target in the mapping. If you are using an external ID, map the external ID to the external ID field in the Salesforce object. If you are using an idLookup field, map the field to the appropriate target field. For example, map the email source field to the Email field in the Salesforce Contact object.
2. Configure the Treat Insert as Upsert or Treat Update as Upsert run-time property to configure a Salesforce run-time property to upsert records.
3. To use the idLookup field instead of an external ID field, enable the Use idLookup Field for Upserts run-time property. By default, the Data Integration Service uses an external ID for upserts. You can configure the run-time property to override the external ID and use the idLookup, instead.
4. To specify an external ID or idLookup field, enter the external ID or idLookup field name in the Use this ExternalId/idLookup Field for Upserts run-time property.

   Note: If you do not enter the name of the external ID or idLookup field, the Data Integration Service selects the first external ID or idLookup field it encounters. If you specify a field that is not an external ID or idLookup field, or if you misspell the field name, the run-time property fails.

Configuring the Maximum Batch Size

The Data Integration Service writes data to a Salesforce target as a batch. The Max Batch Size attribute in the session properties determines the maximum number of records the Data Integration Service can write to a Salesforce target in a batch. The Salesforce service can receive a maximum of 200 records in a single insert, update, or delete operation.

To minimize the number of calls made to the Salesforce service, each batch must accommodate the maximum number of records as configured in the Max Batch Size property.

Handling Null Values in Update and Upsert Operations

By default, the Data Integration Service does not replace existing values in a Salesforce record with null values from the source during an update or upsert operation. To replace existing values with null values, configure the Set Fields to NULL session property for the Salesforce target.

You cannot set the value of an external ID field in a Salesforce target to NULL. The session fails if you enable the Set Fields to NULL session property and the session tries to replace the value in an external ID field with a null value.

Override an External ID with an idLookup for Upserts

The Data Integration Service can use the external ID or idLookup fields when performing an upsert operation to identify records in a Salesforce target. By default, the Data Integration Service uses the external ID field for upserts. You can configure the session to override the external ID field and use the idLookup field, instead.

APPENDIX A

Data Type Reference

This appendix includes the following topics:

- Data Type Reference Overview, 36
- Salesforce Data Types and Transformation Data Types, 36

Data Type Reference Overview

The Developer tool uses the following data types in PowerExchange for Salesforce mappings.

Salesforce native data types

Salesforce native data types appear in the physical data object column properties.

Transformation data types

Set of data types that appear in the transformations. They are internal data types based on ANSI SQL-92 generic data types, which the Data Integration Service uses to move data across platforms. Transformation data types appear in all transformations in a mapping.

When the Data Integration Service reads source data, it converts the native data types to the comparable transformation data types before transforming the data. When the Data Integration Service writes to a target, it converts the transformation data types to the comparable native data types.

Salesforce Data Types and Transformation Data Types

The following table lists the Salesforce data types that Data Integration Service supports and the corresponding transformation data types

Salesforce Data Type	Range and Description	Transformation Data Type
AnyType	Polymorphic data type that returns string, picklist, reference, boolean, currency, integer, double, percent, ID, date, datetime, URL, or email data	String
Base64	Base64 encoded binary data	String

Salesforce Data Type	Range and Description	Transformation Data Type
Boolean	Boolean (true/false) values	Integer
Byte	A set of bits	String
Combobox	Enumerated values	String
Currency	Currency values	Decimal
DataCategoryGroupReference	Types of category groups and unique category names	String
Date	Date values	Date/Time
DateTime	Date and time values	Date/Time
Double	Double values	Decimal
Email	Email addresses	String
Encrypted String	Encrypted text fields contain any combination of letters, numbers, or symbols that are stored in encrypted form	String
ID	Primary key field for a Salesforce object	String
Int	Fields of this type contain numbers with no fraction portion	Integer
Master record	ID of the merged record	String
Multipicklist	Multiple-selection picklists, which provide a set of enumerated values from which you can select multiple values	String
Percent	Percentage values	Decimal
Phone	Phone numbers	String
Picklist	Single-selection picklists, which provide a set of enumerated values that you can select one value from	String
Reference	Cross-references to another Salesforce object	String
String	Character strings	String
Textarea	String that appears as a multiple-line text field	String
Time	Time values	Date/Time
URL	URL values	String

INDEX

A

advanced properties
  input 23

B

batch size
  description for Salesforce 35
bulk API 34
Bulk API target session
  configuring for Salesforce 34

C

capture deleted and archived records 33
column properties 18
configuring for Salesforce
  Bulk API target sessions 34
configuring HTTP proxy options
  Developer tool 9
creating
  Salesforce connection 13
  Salesforce data object read operation 26
custom Salesforce objects 15

D

data object read operation
  creating 26
datatype reference overview 36
DTM Buffer Size
  configuring for Salesforce 35
dynamic mappings
  developing and running 30

E

external ID
  description 34
  overriding with idLookup 35

G

general properties
  input 22

I

idLookup
  description for Salesforce 34
  overriding the external ID 35
importing
  Salesforce data object 25
input properties 22

M

mapping output 27, 28

N

nulls
  handling in upserts and updates 35

O

overview
  Salesforce data object 15

P

performance
  configuring buffer block size for Salesforce 35
PowerExchange for Salesforce
  overview 7
properties
  Salesforce data object 17
  Salesforce data object read operation 17

Q

query properties
  data object read operation 19

R

related objects 16
rules and guidelines
  related objects 16
run-time processing
  overview 32

S

Salesforce
  dynamic mapping 29
  mapping example 27, 28
  mapping input 27, 28
  mapping overview 27
Salesforce connection
  creating 13
  overview 11
Salesforce data object
  importing 25
Salesforce data object read operation
  creating 26
  properties 17
Salesforce dynamic mapping
  example 30
session conditions
  DTM Buffer Size for Salesforce 35
Set Fields to NULL
  session property 35
source properties 17
standard Salesforce objects 15

U

upsert
  configuring for Salesforce 34
  description for Salesforce 34
  external ID 34
  overriding external ID with idLookup 35
  Salesforce idLookup field 34
  Salesforce session configuration 34
use query all 33