Adeptia Suite Salesforce Integration Accelerator 6.0

DOCUMENT INFORMATION

Adeptia Suite Salesforce Integration Accelerator Guide

Adeptia Suite Version 6.0

Printed April 2013

Printed in USA

Adeptia Support Information

For support queries, please contact us at support@adeptia.com.

Access the Adeptia Web site at the following URL:

www.adeptia.com

Copyright

Trademarks

Confidentiality

Disclaimer

TABLE OF CONTENTS

PREFACE

This document enables the users to understand how to use Salesforce Integration Accelerator. Adeptia Salesforce Integration Accelerator is designed to help speed up implementing a solution for connecting Salesforce.com with the applications and systems of an organization. It uses the Salesforce Enterprise Web Services API to connect to Salesforce.

Target Audience

This document is intended for the users who want to integrate their application with Salesforce CRM.

HOW IS THIS GUIDE ORGANIZED?

This guide is organized into the following sections:

Section	Description
Preface	Introduction to this document.
Overview	About Adeptia Salesforce Integration Accelerator.
Salesforce Data Model	A brief description of Salesforce Data Model.
Installation	Installation and configuration steps of Salesforce Accelerator.
Accelerator Templates, Services and Process Flows	Description of bundled templates, services and process flow.
Using Salesforce Integration Accelerator in Customer Environment	Detailed description about how to use pre-built accelerator services for your implementation.
Handling Salesforce Custom Fields	Description of handling custom fields.
Appendix A: System Requirements and Prerequisites	List of pre-requisites.
Appendix B: Useful Links	Useful links .

CONVENTIONS

The following tables list the various conventions used in Adeptia documentation. We follow these conventions to help you quickly and easily identify particular elements, processes, and names that occur frequently in documents.

Typographical conventions

This guide uses the following typographical conventions:

Convention	Description
Bold text	Indicates one of the following: • Screen element • New terminology • A file or folder name • A control in an application's user interface • A registry key • Important information
Italic text	Indicates a reference or the title of a publication.
Monospaced text	Indicates code examples or system messages.
Monospaced bold text	Indicates system commands that you enter.
Hyperlink	Indicates an Internet link to target material.

Graphical conventions

This guide uses the following graphical conventions:

Convention	Description
	Indicates additional information that may be of interest to the reader.
	Indicates cautions that, if ignored, can result in damage to software or hardware.

CONTACTS/REPORTING PROBLEMS

These sections present contact information for a variety of situations.

Sales

In case of any sales queries, please contact us at sales@adeptia.com.

Support

For support queries, please contact us at support@adeptia.com.

Latest updates and information

For the latest updates and information, please visit us at www.adeptia.com.

Adeptia Web site

Access the Adeptia Web site at the following URL:

www.adeptia.com

OVERVIEW

Adeptia Salesforce Integration Accelerator is designed to help speed up implementing a solution for connecting Salesforce.com with the applications and systems of an organization. It uses the Salesforce Enterprise Web Services API to connect to Salesforce.

It consists of Adeptia Server process flow templates and services that can be easily and quickly configured to create data flows between Salesforce and applications and databases.

The information in Salesforce is located in an offsite location and with SIA data transfer can be simplified into back-end applications to automate business processes such as automatically triggering actions, notifications, workflow tasks for other business users and for generating consolidated reports. Adeptia helps by setting up either real-time or scheduled (batch) integration flows between Salesforce and back-end systems. These "deploy-and-forget" flows run automatically in the background to share information and require no manual intervention except to handle exceptions and errors. This automation is provided by the rich set of triggers, e.g. Calendar based, Timer based etc., within Adeptia Suite. These integration flows can be bi-directional, so data can move in both directions into and out of Salesforce.

Adeptia's Business Process Management capability allows automating of a complete business process where interfacing with Salesforce is only one of the many tasks involved in overall process. So, human workflow tasks can be added into the overall flow for business users to interact with data and allow them to validate information before it is synchronized, correct errors and exception and take decisions. This provides more automation and control than just moving of data. The workflow task feature is available in Adeptia BPM and Integration Suite only.

This solution supports the integration of live environment for production as well as facilitates testing by integrating the Salesforce sandbox environment. This allows you to test and verify your flows before moving to production. Salesforce sandbox environment is basically a replication of your Salesforce live environment, you can use for development, testing and training without disrupting the live environment.

Using Salesforce Integration accelerator, you can:

1.   Move data such as Accounts and Contacts etc from Salesforce live or sandbox environment to your backend application and vice versa.

2.   Map the fields of Salesforce object with the fields of your backend application.

3.   Provides an automated replication capability to copy and save this data behind your firewall from where it can be easily backed-up and recovered.

4.   Do one time migration of data from Salesforce to your backend application. Adeptia offers wide range of application and format support for mapping and transforming data to meet these needs.
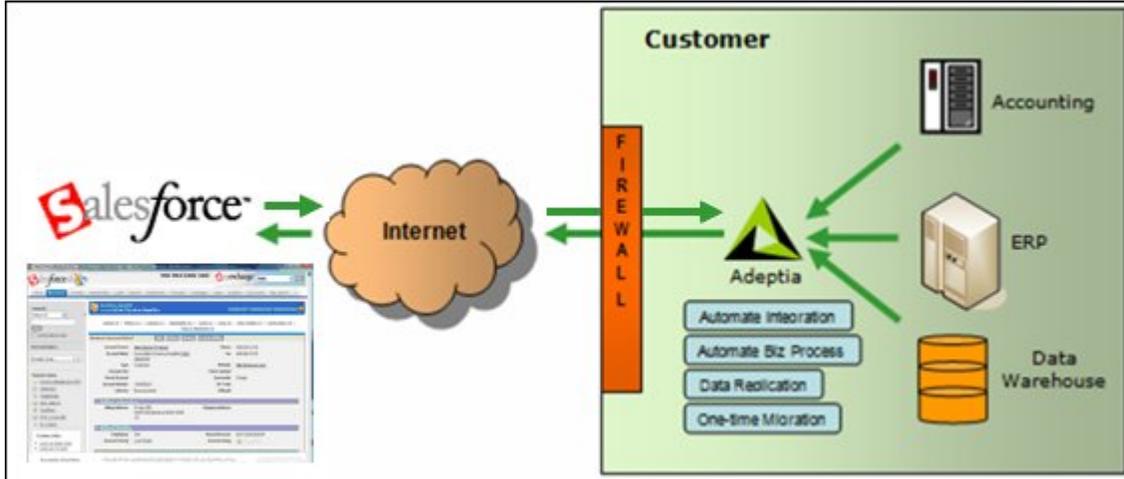

Figure 1

Adeptia Salesforce Integration Accelerator is created using Salesforce API Version 23.0. The Salesforce Enterprise WSDL is already bundled in the Salesforce Integration Accelerator. The Enterprise WSDL is packaged with Adeptia Salesforce Integration Accelerator.

SALESFORCE DATA MODEL

Adeptia Salesforce Integration Accelerator can move the data such as Accounts and contact etc. from Salesforce to other application or vice-versa. In Salesforce, this data is referred as Objects. Salesforce data model has mainly two types of objects:

- Standard Objects: Objects, which are already created by Salesforce are called Standard objects.
- Custom Objects: Objects, which you create in Salesforce for your organization, are called Custom objects.

For details about Salesforce objects refer to Standard Object section of Web Services API Developer's Guide from http://www.salesforce.com/us/developer/docs/api/index.htm

Adeptia Suite provides a Web Service interface which connects to Salesforce SOAP Web Service API, to retrieve, populate, and update the objects.

To connect Adeptia Suite with Salesforce SOAP Web Service API, an Enterprise Edition, Unlimited Edition, or Developer Edition account is required. Developer Edition account is used for building and testing the integration solution. Enterprise or Unlimited edition account is used for production solution. For the list of all pre-requisites of Adeptia Salesforce Integration Accelerator, refer to Pre-requisites section.

In case there is a version upgrade in Salesforce SOAP Web Service API or you have create or modified any objects, or added any custom field to any objects, You need to download the latest enterprise WSDL from you Salesforce environment and update the Adeptia Web Service interface. To know how to download the latest enterprise WSDL and use it in Salesforce Integration Accelerator, refer to Handling Salesforce Custom Fields section.

INSTALLATION

Setting up Adeptia Salesforce Integration Accelerator consists of three steps.

1. Installing Adeptia Salesforce Integration Accelerator
2. Configuring the Salesforce Integration Accelerator
3. Configuring Database Tables

The steps mentioned above are the main steps needed to setup the Adeptia Salesforce Integration Accelerator. In case you have defined custom fields in your Salesforce environment, you have to get the latest Enterprise WSDL from you Salesforce Environment and update the Web Service Consumer and Mapping activities. To know how to handle custom fields in Salesforce Integration Accelerator, refer to Handling Salesforce Custom Fields section.

INSTALLING ADEPTIA SALESFORCE INTEGRATION ACCELERATOR

For installing Adeptia Salesforce Integration Accelerator, please refer Adeptia Suite Installation Guide provided with the installation package. Before installing Adeptia Salesforce Integration Accelerator, refer to Appendix A in this document to review the prerequisites.

1. While installing Salesforce Integration Accelerator, the Configure Accelerator screen is displayed (see Figure 2).
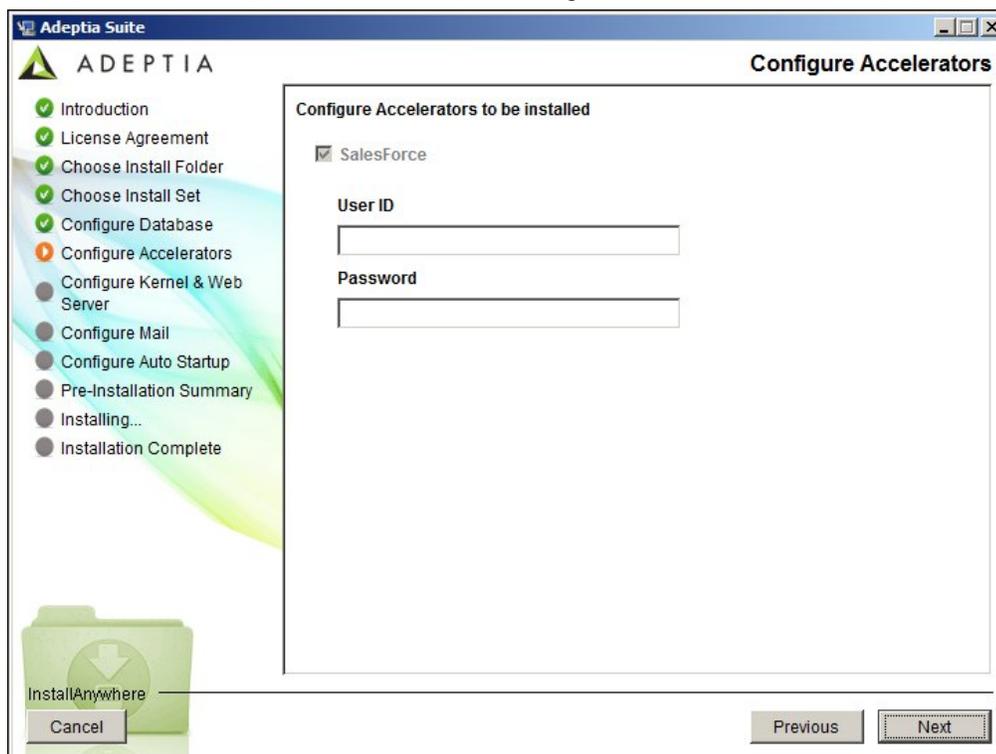

Figure 2

2. In the screen displayed above, enter the User ID and Password, which you use to login into Salesforce.com live environment, in User ID and Password field respectively.

The password must be appended with the Salesforce Security Token. For example if your Salesforce password is 'MyPassword' and Security token is 'xxxxxxxxxxxxxxxxxxxxxxx' then you will be required to enter 'MyPasswordxxxxxxxxxxxxxxxxxxxxxxx'.

- To know how to get the security token from Salesforce, refer to Salesforce Security Token section.
- You can also configure these parameters after installation of Salesforce accelerator. To know how to configure these parameter refer to Salesforce.com account Configuration section.

3.   All other steps are similar to as explained in the Adeptia Suite Installation Guide.

After installing Salesforce Integration Suite, login into it using the following information:

User account: sfuser

Default password: sf1234

where:

'sfuser' is the owner of all the pre-built objects of Salesforce Integration Accelerator. You can access these objects either through 'admin' user or 'sfuser'.

It is recommended that 'sfuser' be used to execute and implement the Salesforce Integration flows. The primary purpose of admin user is to do tasks related to system administration like User and Group management etc.

After login, you will see the Salesforce process flows and services in the repository such as Design -> Process Flows. Refer to Adeptia Suite Developer Guide to learn more about process flows and services.

CONFIGURING THE SALESFORCE INTEGRATION ACCELERATOR

A Salesforce.com user account with integration privileges is required to connect to Salesforce.com through web services. The account should also have the necessary permissions for operations such as to create, update and view objects details based on the integration needs. To configure this user account with Adeptia Server so it can be used at run-time, perform the following steps:

1.   Login into Adeptia server as admin.

2.   The default password of admin user is indigo1.

3.   Navigate to Administer > Setup > Application Settings.

4.   Click the Update System Properties link.

5.   Expand the Solution Properties and then expand Salesforce Integration Accelerator Parameters. (See Figure: 3).
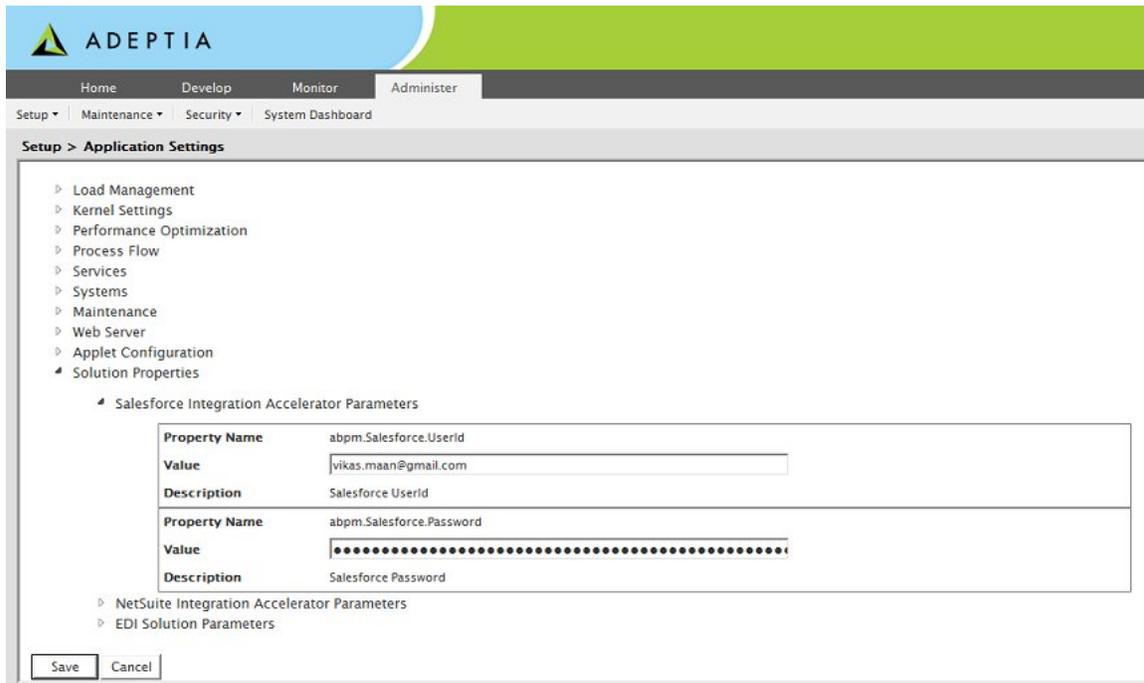

Figure 3

6.   Enter the User ID and Password, which you use to login into Salesforce.com live environment, in abpm.Salesforce.UserID and abpm.Salesforce.Password fields respectively.

> The password must be appended with the Salesforce Security Token. For example if your Salesforce password is 'MyPassword' and Security token is 'xxxxxxxxxxxxxxxxxxxxxxx' then you have to enter 'MyPasswordxxxxxxxxxxxxxxxxxxxxxxxxx'.

7.   Enter the required parameters and click Save. The following screen is displayed.

Application setting changes have been saved.

Note: To commit this change, click on Reload configuration
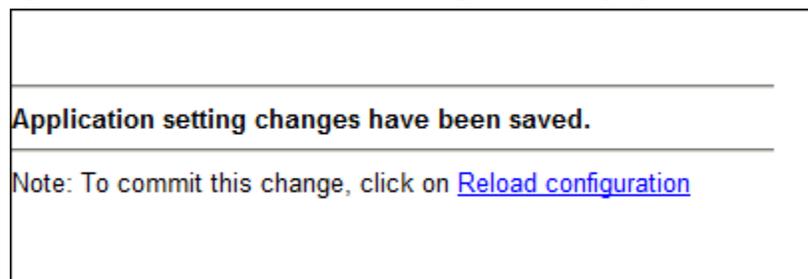
Figure 4

8.   Click the **Reload Configuration** hyperlink to commit the change.

9.   In case you also want to move data from Salesforce Sandbox environment, you need to configure User ID and Password of Salesforce Sandbox environment.

10. To configure the Sandbox User ID and password, go to ./Serverkernel/etc folder.

11. Open the Server-Configure.properties file in a text editor and add the properties, highlighted in the figure below, under Salesforce Integration Accelerator Parameters group.

```
<Category name="Solution Properties">
    <Group name="Salesforce Integration Accelerator Parameters">
        <Property comment="Salesforce UserId" dynamic="yes" name="abpm.Salesforce.UserId"></Property>
        <Property comment="Salesforce Password" dynamic="yes" encrypt="y" name="abpm.Salesforce.Password"></Property>
        <Property comment="Salesforce Sandbox UserId" dynamic="yes" name="abpm.Sandbox.Salesforce.UserId"></Property>
        <Property comment="Salesforce Sandbox Password" dynamic="yes" encrypt="n" name="abpm.Sandbox.Salesforce.Password"></Property>
    </Group>
```

12.   Configure the user id and password, which you use to login into salesforce.com sandbox environment, in abpm.Sandbox.Salesforce.UserID and abpm.Sandbox.Salesforce.Password properties respectively.

> ℹ️ The password must be appended with the Salesforce Security Token. For example if your Salesforce password is 'MyPassword' and Security token is 'xxxxxxxxxxxxxxxxxxxxxxxx' then you have to enter 'MyPasswordxxxxxxxxxxxxxxxxxxxxxxxx'.

13.   Save the file and restart the Kernel and WebRunner.

CONFIGURING DATABASE TABLES

To execute the template flows, which are used to move the objects from Salesforce to internal database and vice-versa, you have to create some tables in your database. To create these tables, a SQL script is provided along with the Adeptia Salesforce Integration Accelerator.

Following are the high-level steps needed to configure this table:

Create Database Tables

To create the tables, run the provided SalesforceScript.sql file in your database.

Modify Database Info Activities

The database info activity DB_Info is used in all the Database Schema, Source and Target activities, which are the part of Salesforce Accelerator. Once you create the table, you need to modify this database info activity to point to the database where you have created the table.

Navigate to Develop > Services > Connector > Database Info.

1.   Edit the DB_Info info activity (See Figure 5)

Figure 5

2.   Change the Server URL, User and Password required to connect to the database where you have created the tables.

3.   Save this activity.

ACCELERATOR TEMPLATES, SERVICES AND PROCESS FLOWS

TEMPLATES

Adeptia provides process templates to make it easy for designing the flows that may be required for customer's integration needs. Most implementations of process flows, to either move the data from or to Salesforce.com, follow a common pattern that is pre-built into the templates. The template process flows are designed to guide through these patterns and to quickly and easily implement a process flows. These templates have used Account and Contact Salesforce objects and explain the process from high level.

Name: SF_Login_PF

This process flow is used to connect and login to Salesforce.com and open a session. The login process flow is designed to connect to Salesforce.com and get session ID and End Point URL for further web service communication. The process flow can be called from other process flows before making a web service call to Salesforce.com.
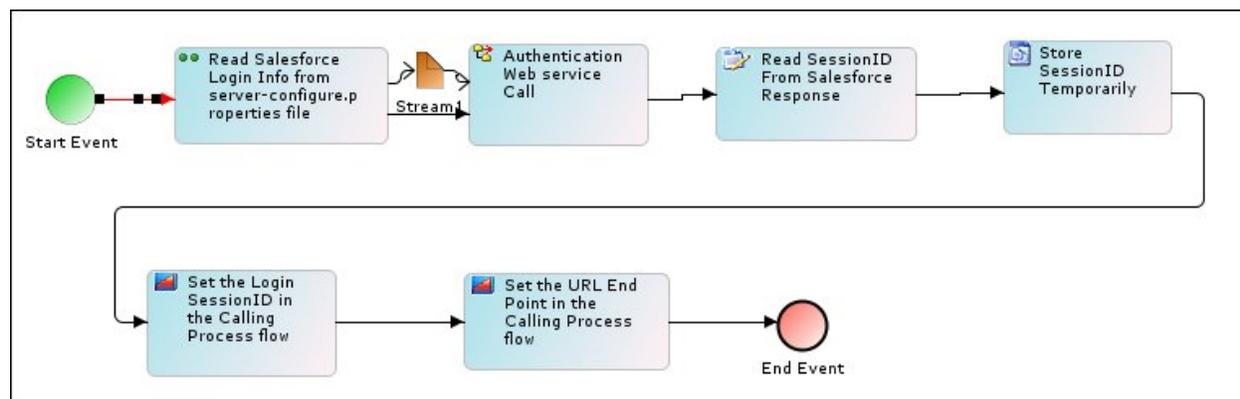

Figure 6

The steps (also called services or activities) in this process flow are:

- Read Salesforce Login Info from server-configure.properties file: This is a custom plug-in, which reads the login information which was set at the time of accelerator installation.
- Authentication Web Service Call: Login call to Salesforce.com
- Read SessionID From Salesforce.com Response: Reads the Session ID from the login Web Services response message.
- Store SessionID: Stores the Session ID for passing it to the calling process flow
- Set the Login SessionID in the Calling Process flow: Passes the session ID to the calling process flow. This session id is set in 'inputHeader' variable in the calling flow.
- Set the End Point URL in the Calling Process flow: Passes the End Point URL to the calling process flow to be used later in that process flow. This End Point URL is set in 'serverURL' variable in the calling flow.

Template Name: SF_Account_To_Salesforce_PF

The process flow template listed in the process flow repository with the name "SF_Account_To_Salesforce_PF" is to help implement a process flow for moving the accounts data from internal system to Salesforce.com.

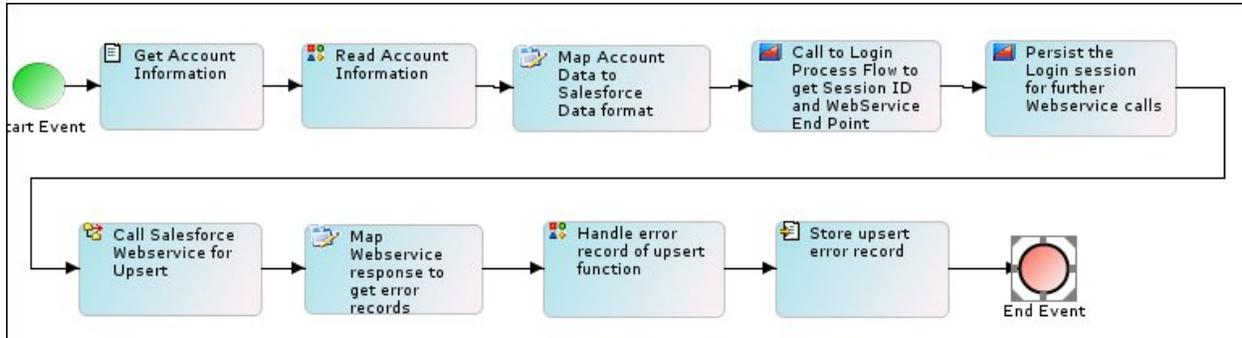Account_To_Salesforce - Account Information from other system to Salesforce.com

Figure 7

The steps (also called services or activities) in this process flow are:

- Get Account Information: SF_ReadAccountInformation_FileSource – This is a file source activity used to read the list of accounts, which are to be updated to Salesforce, from a file.

- Read Account Information: SF_AccountInfo_TextSchema - This is a text schema to parse account information from the source file.

- Map Account to Salesforce Data format: SF_AccountInfo_To_SalesForce_MAP - This maps the account information to Salesforce upsert data format.

- Call to Login Process Flow to get Session ID and Web service End Point: This is a call to SF_Login_PF process flow that logs into salesforce.com with the user account information configured initially, gets the login session ID and uses it for further web services communication. Except for the login web service call, all other web services call End Point with Salesforce.com is dynamic. This End Point is passed to the integration application in the response message in the login call. This process flow also gets the End Point and passes on to the calling process flow for further web services communication.

- Persist the Login session for further Web services Call: This is a put-context-var activity which sets the web services End Point URL received through the login process flow in a context variable.

- Call Salesforce.com Web services: SFUpsertWebservice - This is a web service consumer activity, which connects to Salesforce.com web services, delivers the objects data and receives the response message.

- Map Webservice response to get error records: SF_AccountUpdate_CSV_Response_MAP – This maps the Salesforce response to store data into file target.

- Handle error record of upsert function: SF_Handle_ErrorRecords_TextSchema – This is a text schema, which parses the mapping output and converts it into text format.

- Store upsert error record: SF_Account_SalesforceError_FileTarget - This is a file target service, stores the web services response to a file. In cases where the response needs to be processed, a mapping services can be put in place of this and do further processing.

Template Name: SF_Account_From_Salesforce_PF

The process flow template listed in the process flow repository with the name "SF_Account_From_Salesforce_PF" helps implement a process flow for moving the data from Salesforce.com to internal system.

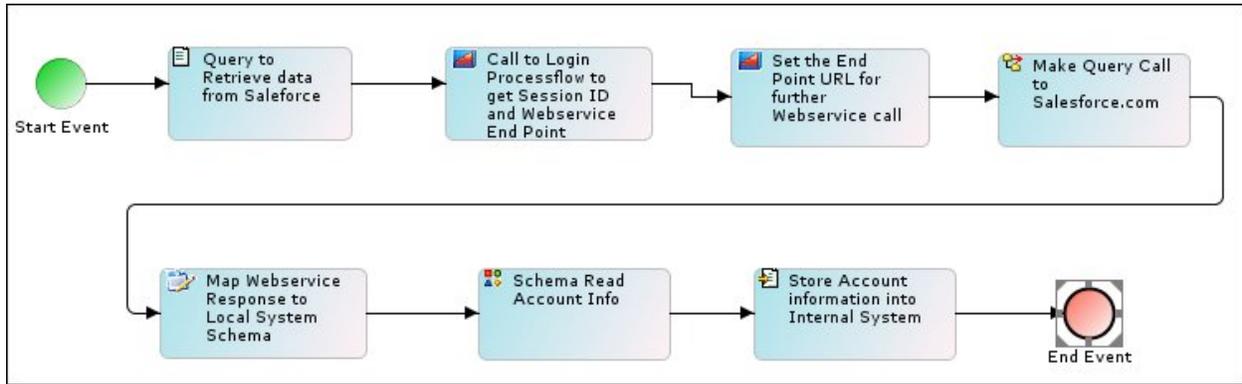Account_From_Salesforce - Account Information from Salesforce.com to other System

Figure 8

The steps (also called services or activities) in this template are:

- Query to retrieve data from Salesforce: SF_AccountQuery_FileSource - This is a file source service, contains the query to retrieve the Account data from Salesforce.com.

- Call to Login Process Flow to get Session ID and Web service End Point: This is a call to SF_Login_PF process flow that logs into salesforce.com with the user account information configured initially, gets the login session ID and uses it for further web services communication. Except for the login web service call, all other web services call End Point with Salesforce.com is dynamic. This End Point is passed to the integration application in the response message in the login call. This process flow also gets the End Point and passes on to the calling process flow for further web services communication.

- Set the End Point URL for further WebService call: This is a put-context-var activity which sets the web services End Point URL received through the login process flow in a context variable.

- Make Query Call to Salesforce.com: SFQuery_WebService - This is a web service that connects to Salesforce.com web services to run the query and receives the response message.

- Map Webservice Response to Local System Schema: - SF_Account_From_SalesForce_MAP - This is a mapping service to process the received query response and a target system. The target system in this case could be a file or database. This uses the pre-packaged XML Schema service SFSchema_OperationResult as the source schema for the mapping.

- Update Response Account Data to Local database: SF_Account_Salesforce2Internal - This is a database service to update the received response data into the local database. In some cases the response data can be put in a file and may even be sent in an email put in a FTP site.

SERVICES

Adeptia Salesforce Integration Accelerator has several process flow services pre-built to speed up the implementation process.

Salesforce.com XML Schemas to push and pull data to and from Salesforce.com

XML schemas for mapping to Salesforce.com format are pre-built as part of the Salesforce accelerator. Here is a list of XML schemas available for Salesforce objects.

Object Name	Description
SFSchema_OperationResult	It's a schema for request and response XML for all Web Service call (WS-Consumer activity). Note: This is a generic schema. For specific function and object type, you need to select appropriate extension element in data mapper for this schema. To know how to create mapping using this Schema, refer the Configure the Services section.
SFSchema_OperationHeader	It's a schema for soap header which is set in all web service call (WS-Consumer activity).

Web Services Operations

Adeptia Salesforce Integration Accelerator has the following pre-built Web Services consumer activities.

Web Service Operation Type	Web Service Name	Description
Login	SFLogin_Webservice	Used for Login to Salesforce live environment.
Upsert	SFUpsert_Webservice	Used for update/insert data into Salesforce live environment.
Query	SFQuery_WebService	Used for getting data from Salesforce live environment.
Login	SFLogin_SandboxWebservice	Used for Login to Sandbox Salesforce environment.
Upsert	SFUpsert_SandboxWebservice	Used for update/insert data into Salesforce sandbox environment.
Query	SFQuery_SandboxWebService	Used for getting data from Salesforce sandbox environment.

For other operations like Insert, Update, Query All, you can also define Web Service consumer.

Following are the steps to define Web Service Consumer:

1. Navigate to Develop → Services → Web Services → Consumer.
2. Click New. The Create Web Service Consumer screen is displayed (see Figure 9).

Figure 9

3. Enter then name and description in the textbox Name and Description.
4. Select the radio button URI.
5. Select the radio button Local/LAN.
6. Click the Browse Wsdl button and select the WSDL provided by Salesforce.
7. Click Next. (See Figure 10).

Figure 10

8. Select the required operation from the combo box Operation.

9. Enter any variable name in the field Input Document. For example, enter $input.

10. Enter '$inputHeader' in the field Input Header Document. This variable contains session ID set by SFPF_Login process flow.

11. Enter the any variable name in the field Output Parameter Name. Do not include $ while defining variable name in the Output Parameter Name field (see Figure 11).

	$input
Input Document	
	Show Input Schema
	$inputHeader
Input Header Document	
	Show Input Header Schema
Output Parameter Name	out
	Show Output Schema

Figure 11

> In Adeptia Suite version 5.2 SP1, the fields Input Document and Output Parameter Name fields have been removed. These variables are set implicitly.

12. Click Next and follow the wizard.

> For detailed instruction to create Web Service Consumer, refer to Adeptia Suite Developer Guide.

PROCESS FLOWS

This section lists the process flows, which are pre-built with Salesforce accelerator. These process flows are ready to use. The SF_Login_PF process flow is called by the other process flows to login into Salesforce and get the Session ID. You do not have to execute this process flow explicitly.

For other process flows, there is no particular execution order. You can execute these process flows in any order unless the objects, which you are moving, are related to each other. For example, if you have to move contacts and accounts, the accounts should be moved first and then contact. The order in which you are moving objects depends upon the relationship between these objects in Salesforce and you should know the relationship between

these objects. Following table displays the the list of pre-built process flows:

Process Flow Name	Description
SF_Login_PF	This process flow logs in into Salesforce live environment and gets the session ID. This process flow is called by other process flows as sub process. You do not have to execute this process flow.
SF_Account_From_Salesforce_PF	Gets Accounts data from Salesforce live environment and stores them in a file.
SF_Account_To_Salesforce_PF	Gets Accounts data from a file and Updates or Adds them to Salesforce live environment.
SF_Accounts_DB_From_Salesforce_PF	Gets Account data from Salesforce live environment and stores them to the internal database.
SF_Accounts_DB_To_Salesforce_PF	Gets Accounts data from internal database and Updates or Adds them to Salesforce live environment.
SF_Contacts_DB_From_Salesforce_PF	Gets Contacts data from Salesforce live environment and store them to internal database.
SF_Contacts_DB_To_Salesforce_PF	Gets Contacts data from internal database and Updates or Adds them to Salesforce live environment
SF_SandBox_Login_PF	This process flow logs in into Salesforce sandbox environment and gets the session ID. This process flow is called by other process flows as sub process. You do not have to execute this process flow.
SF_SandboxAccounts_DB_From_Salesforce_PF	Gets Accounts data from Salesforce sandbox environment and stores them to the internal database.
SF_SandboxAccounts_DB_To_Salesforce_PF	Gets Accounts data from internal database and Updates or Adds them to Salesforce sandbox environment.
SF_SandboxContacts_DB_From_Salesforce_PF	Gets Contacts data from Salesforce sandbox environment and stores them to the internal database.
SF_SandboxContacts_DB_To_Salesforce_PF	Gets Contacts data from internal database and Updates or Adds them to Salesforce sandbox environment.

To know how to use these template flows to further implement the process flows to meet your requirements, refer to "Using Salesforce Integration Accelerator in Customer Environment".

USING SALESFORCE INTEGRATION ACCELERATOR IN CUSTOMER ENVIRONMENT

This section explains how you can use pre-built accelerator services for your implementation.

IMPLEMENTING PROCESS FLOW TO PULL DATA FROM SALESFORCE TO OTHER SYSTEM

We will create a process flow to pull data from Salesforce.com. In this step a copy of the template process flow is made because we don't want to edit or overwrite the template since we want it to be available for future use. The following steps are performed:

Design

1. Navigate to **Develop > Process > Process Flow**, and select **SF_Account_From_Salesforce_PF**.
2. Click the **Edit** button.
3. Edit the Name field by replacing SF_Account with Object Name. For example ERP_Contact_From_Salesforce. In addition, edit the Description field. (See Figure 12).

Figure 12

4.   Click Save AS.

5.   Now the process flow can be used as is or can be modified to fit your business needs.

Configure the Services

Configuring is the second step in completing the process flow. Services are the individual steps of the process flow. After creating the source services, source schema and the mapping objects to suit the business requirements, edit the process flow to attach the activities to the process flow. Most web service and target schemas are pre-built as part of the Salesforce accelerator. In many only the source activities needs to be created, mapped to the existing target Web Service schemas and attached to the template process flow. For more information about creating and using services, refer to Adeptia Suite Developer Guide.

For example if you want to retrieve Contacts from Salesforce to your internal database:

Create an XML file to write the search query. Sample files for this purpose are stored in ../../Solutions/SalesForce/input/Objects/Account folder.

Create a file source to read the XML file.

Create a Database Schema. To know how to create database schema, refer the Adeptia Suite Developer Guide.

Create mapping activity.

a.   Open Data Mapper. To know how to create data mapping activity, refer to Adeptia Suite Developer Guide.

b.   Load SFSchema_OperationResult XML Schema at source side and Database Schema at the target side.

c.   Map your source with Salesforce elements as per your requirement.

Configure the Process Flow

Once you have created all the required activities, attach them to the process flow. This is done in the Process Designer by opening the flow in Edit mode.

1.   In the Process Designer, navigate to the Service in the Activities Panel.

2.   Select the activity and then drag and drop it on the step in the process flow.

3.   Before the Web Service consumer activity, use Call action to call the SFPF_Login flow.

4.   After the call action, use put-context-var action to set the login session to the Web Service consumer activity. (See Figure 13).
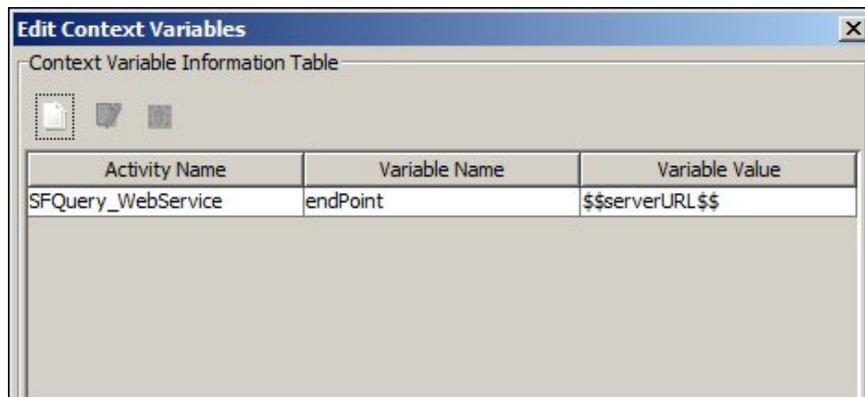

Figure 13

5.   Repeat until all the steps in the process flow are filled with an activity.

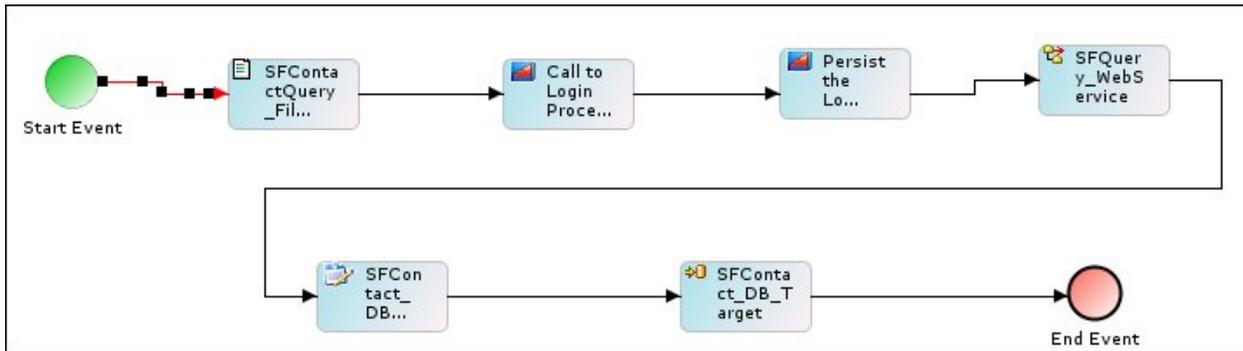6. Now the process flow is complete and it is ready for execution.

Figure 14

Deploy the Process flow

Event and Triggers

Adeptia server offers several types of events and triggers such as database, calendar, File, HTTP, Mail, Timer, and Web services to tie to the process flow so they can be completely executed automatically from various sources.

For example, to read the account information from Salesforce.com to synchronize the data with the internal system, the file can be scheduled to run the process flow every day.

This creates a Calendar event to run the process flow every day at 6:00 AM.

Figure 15

| | To know more about other possible events, refer to Adeptia Suite Developer Guide. |

Event Registry

After the event or trigger is created, the event must be registered to automate executing the process flow.To register Create an Even Registry from menu "Develop > Events > Event Registry".
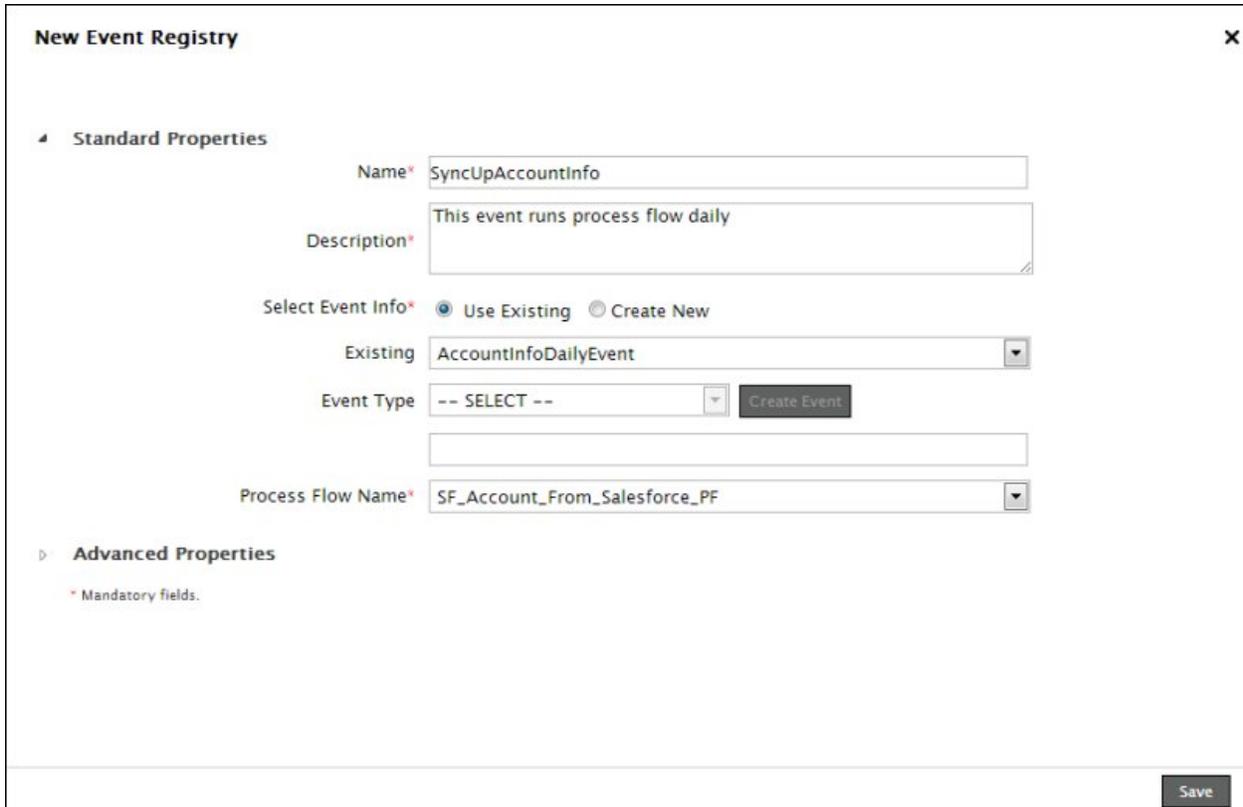

Figure 16

The Final step is to activate the event.

    📂🖎 Navigate to Develop > Events > Calender(Select Event Type which is created) and click on Calendar Event.

Select the event and click on the **Activate** link.

Figure 17

Now the event is active and the process flow will run every day at 6:00 AM.

IMPLEMENTING PROCESS FLOW TO PUSH DATA FROM INTERNAL SYSTEM TO SALESFORCE

We will create a process flow to send data from an ERP or any other local system to Salesforce.com. In this step a copy of the template process flow is made because we don't want to edit or overwrite the template since we want it to be available for future use. The following steps are performed:

Design

1. Navigate to **Develop > Process > Process Flow**, select SF_Account_To_Salesforce_PF.
2. Click Edit. Edit the Name field by replacing SF_Account with Object Name. For example, ERP_Contact_To_Salesforce. In addition, edit the Description field.

Edit Process Flow: SF_Account_To_Salesforce_PF

◢ Standard Properties

Name*	SF_Account_To_Salesforce_PF
Description *	Update and Add the Account Data to Salesforce
Logging Level *	ERROR
Repository File Retention *	DONT DELETE

Process Flow Designer

▷ Advanced Properties

* Mandatory fields.

Figure 18

3. Click Save As.
4. Now the process flow can be used as is or can be modified to fit your business needs.

Configure the Services

Configuring is the second step in completing the process flow. Services are the individual steps of the process flow.

For example if your source is a database, then you need to:

1. Create a Database Schema. To know how to create database schema, refer to Adeptia Suite Developer Guide.

 Create mapping activity.

d. Open the Data Mapper. To know how to create data mapping activity, refer to Adeptia Suite Developer Guide.

e. Load Database Schema at source side and SFSchema_OperationResult XML schema at the target side.

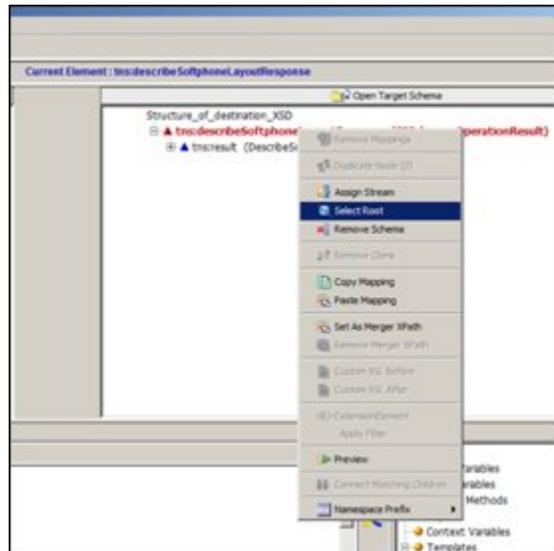f.   Right click the root element on the target schema and then click Select Root option. (See figure below.)

Figure 19

g.   In the Select Root Element dialog box, select the required operation. For example: upsert. (See figure below)
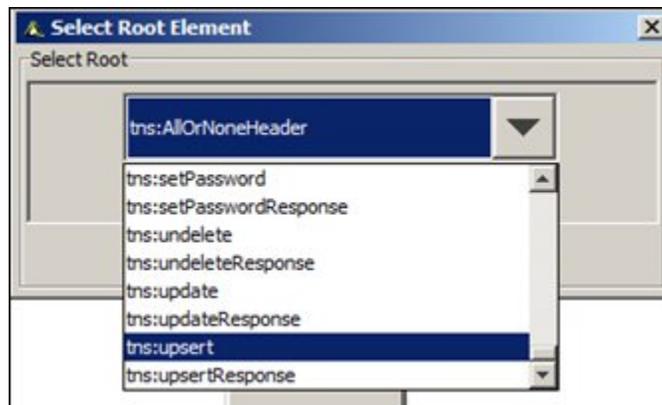

Figure 20

h.   A dialog box is displayed to for your confirmation to change the root node of the schema.

i.   Click Yes. Sub elements of selected operation are displayed.

j. To select the object/noun, right click the tns:Objects element, and select the click the Extension Element option. (See Figure 21)
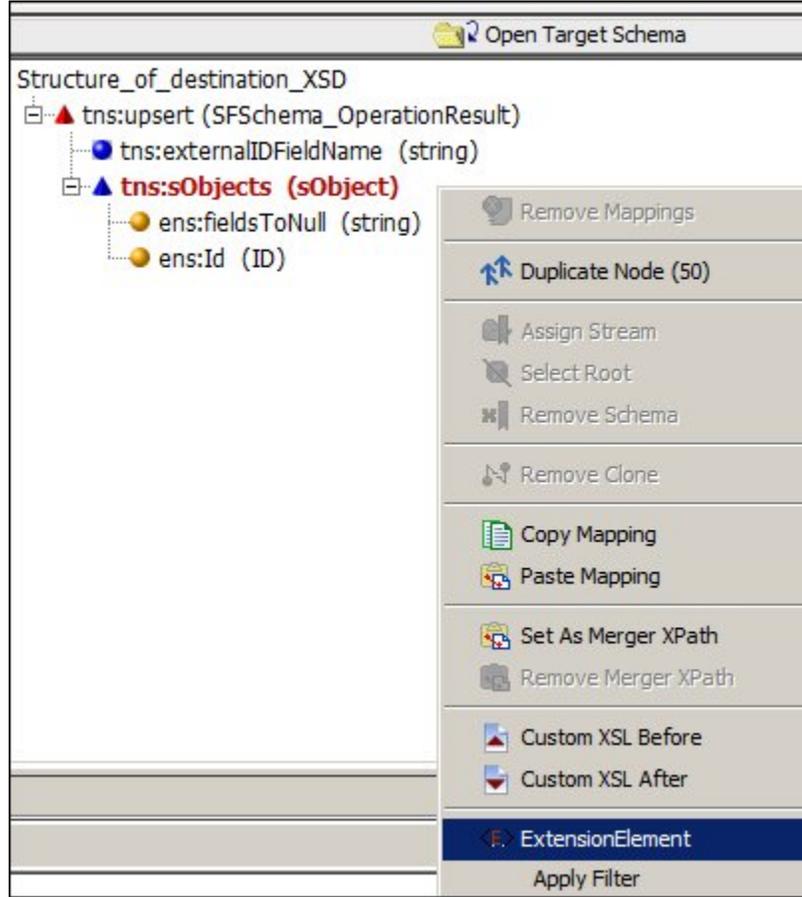

Figure 21

k. The Select Extension Element dialog box is displayed is displayed to select the extension element. (See Figure 22)
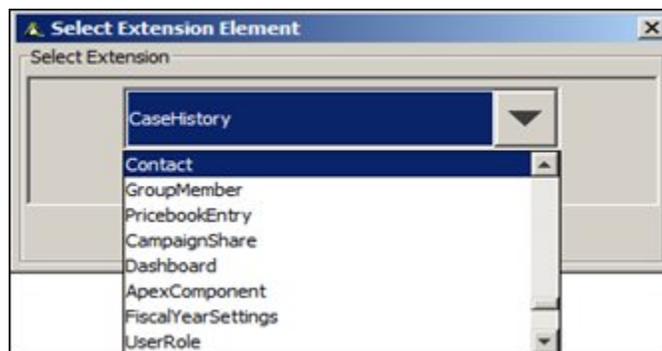

Figure 22

l. In the Select Extension Element, select the required object. For example: Contact.

m. Click OK.

n. Now map the source elements with Salesforce elements as per your requirement.

> While defining mapping activity to push data to Salesforce, you need to configure External ID. To know how to configure External ID, refer to Configuring External ID section.

Configuring External ID

Upsert operation is used to "updated existing records" and if not found then to "Insert new records" in Salesforce. For update, one must know the field based on which it will be updated, e.g. primary key in the database. Salesforce provides External ID mechanism to achieve the upsert operation. You can define any field in your record as an external Id if it is unique.

To know how to define External ID in Salesforce, refer to Configuring External ID in Salesforce section.

To know how to configure External ID in Adeptia Suite, refer to Configuring External ID in Adeptia Suite section.

Configuring External ID in Salesforce

This section explains how to configure External ID in Salesforce.

1. Login to www.salesforce.com.
2. Click on the Setup link at the top.
3. Navigate to APP Setup ⯈ Customize.
4. Expand the objects, for which you want to configure the External ID. For example: Contacts.
5. Click the Fields link.
6. In the Custom Fields and Relationship section, click the New button.
7. Select the Data Type and click Next.
8. Enter the Field Label and Field Name. For example: ContactName.
9. Enter the other required information.
10. Select the External ID checkbox.
11. In the default value field, enter $User:<FieldName>.
12. Here <FieldName> is the name of the field, which you want to configure as External ID.
13. For example: $User:FirstName.
14. Click Save. When this field is saved, an API name is given to this custom field. (See figure below)

Figure 23

For example in the figure shown above, ContactName__c is given as API Name. This API Name is used while creating mapping activity in Adeptia Suite.

Configuring External ID in Adeptia Suite

1. Navigate to Develop > Services > Data Transform > Data Mapping.
2. Click New to create a new mapping activity, or edit the existing mapping activity.
3. Load source schema.
4. Load Database Schema at source side and SFSchema_OperationResult XML schema at the target side.
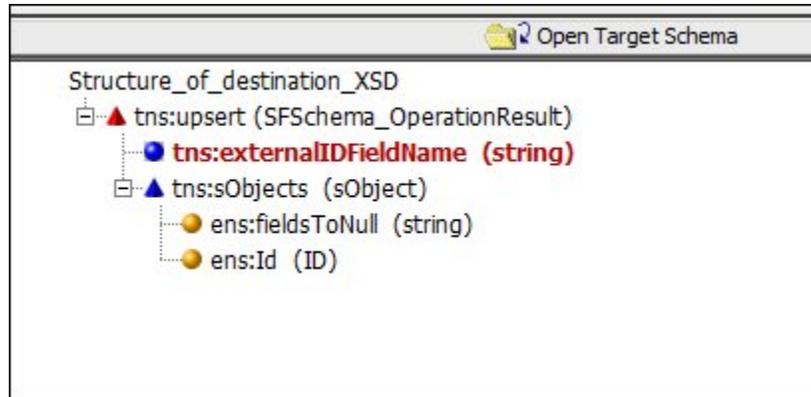5. Expand the hierarchy of the target schema. (See Figure 24).

Figure 24

6. Create a constant and enter the API Name of the External ID field within the quotes. For example: 'ContactName__c'. (See Figure 25.)

Figure 25

7. Now map this constant with externalIDFieldName displayed in the figure above.

8. Map the other fields.

9. Save this activity.

Configure the Process flow

Once you have created all the required activities, attach them to the process flow. This is done in the Process Designer by opening the flow in Edit mode.

1. In the Process Designer, navigate to the Service in the Activities Panel.

Select the activity and then drag and drop it on the step in the process flow.

Before Web Service consumer activity, use Call action to call the SFPF_Login flow.

After the call action, use put-context-var action to set the login session to the Web Service consumer activity (See Figure 26).

Figure 26

Repeat until all the steps in the process flow are filled with an activity.

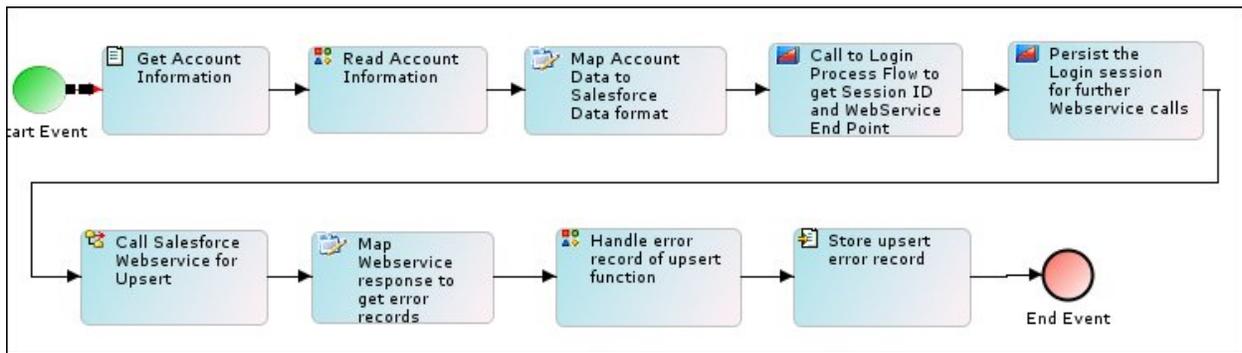Now the process flow is complete and it is ready for execution.

Figure 27

Once you have created the process flow, the next step is to bind this process flow with event so that the process flow can be executed automatically. To know how to bind a process flow with event, refer the section Deploy the Process Flow.

HANDLING SALESFORCE CUSTOM FIELDS

Salesforce allows you to define the custom fields in your Salesforce environment. In case you have defined any custom field in your Salesforce environment, you need to do some modification in the Salesforce Accelerator so that you can pull data from these custom fields or you can populate data to these custom fields.

The high level steps to update the Salesforce Integration Accelerator to handle custom fields are as below:

- Download latest Enterprise WSDL
- Update Web Service Consumer activity to use the latest WSDL.
- Update the Data Mapping activity

DOWNLOAD LATEST ENTERPRISE WSDL

Steps to download the Enterprise WSDL from Salesforce.com:

Login to www.salesforce.com.

Click on the Setup link at the top.

Navigate to App Setup ☐ Develop ☐ API.

Click on Generate Enterprise WSDL below the Partner WSDL section.

The link would open up the WSDL file in the browser.

From the file menu, click Save As and save the file with the name enterprise.wsdl in any folder.

UPDATE WEB SERVICE CONSUMER ACTIVITY

Once you have downloaded the latest Enterprise WSDL, you need to update Web Service Consumer activities to use this WSDL.

Steps to update Web Service Consumer activity:

1. Navigate to Develop > Services > Web Services > Consumer.

2. Select the Web Service consumer activity and click Edit (See Figure 28).

Edit Web Service Consumer: SF_Upsert_WebService

Standard Properties

Specify web service definition location.

Name*	SF_Upsert_WebService
Description*	Used for passing the query in the Upsert Operation
Character Set Encoding*	ISO-8859-1
Consumer Type*	● SOAP ○ REST
Web Service Definition Location*	○ UDDI ● URI
URI Location	○ HTTP ● Local/LAN
WSDL URL (HTTP URL)*	
WSDL File Path(Local/LAN)*	[Browse WSDL]

enterprise.wsdl

[Next]

Figure 28

3. Click the Browse Wsdl button and select the latest enterprise WSDL and follow the instruction. All other parameters should remain as is.

4. Save this activity.

UPDATE MAPPING ACTIVITY

When you update the Web Service consumer activity, the list of fields gets updated in the XML schemas, which are created using this consumer activity. The next step is to map these fields in the data mapping activities.

Steps to update the data mapping activities:

1. Navigate to Develop > Services > Data Transform > Data Mapping.

2. Select the Data mapping activity that you want to update and click Edit. The custom fields are displayed in the XML schema hierarchy.

3. Map these fields as per your requirement.

4. Save this activity.

6

APPENDIX A: SYSTEM REQUIREMENTS AND PREREQUISITES

SYSTEM REQUIREMENTS

Hardware

- CPU: Intel Pentium IV or newer
- RAM: 2GB min, 4GB recommended
- Hard Disk: 2GB

Software

- Operating System: Windows NT/2000+/XP, UNIX, Linux
- Database: Embedded DB included. Commercial RDBMS recommended for production environments
- Java: J2SE v1.6 SDK (Bundled with Adeptia Server)
- Browser: IE7.0 or newer
- Misc.: Adobe Acrobat Reader
- Client End Applets: JRE 1.6

PRE-REQUISITES

Salesforce Account

To integrate with Salesforce.com, an Enterprise Edition, Unlimited Edition, or Developer Edition account is required. Developer Edition account is used for building and testing the integration solution. Enterprise or Unlimited edition account is used for production solution.

User should have the knowledge of Salesforce Data Model.

Salesforce Security Token

To access Salesforce via Adeptia Suite, you need to replace your current password with a combination of your password and a security token. If you already know your security token, you can use it, while installing Salesforce accelerator. In case you do not know the security token you need to reset the current security token. Salesforce doesn't allow to view the security token.

Steps to reset and generate new security token:

1. Log into your Salesforce instance.
2. Go to Setup > My Personal Information > Reset My Security Token.

3. Click the Reset Security Token button, you will then be sent an email containing your Security Token.

ADEPTIA IMPLEMENTATION APPROACH

Adeptia's approach for implementing any solution is the following 3 simple steps: Design, Configure and Deploy.
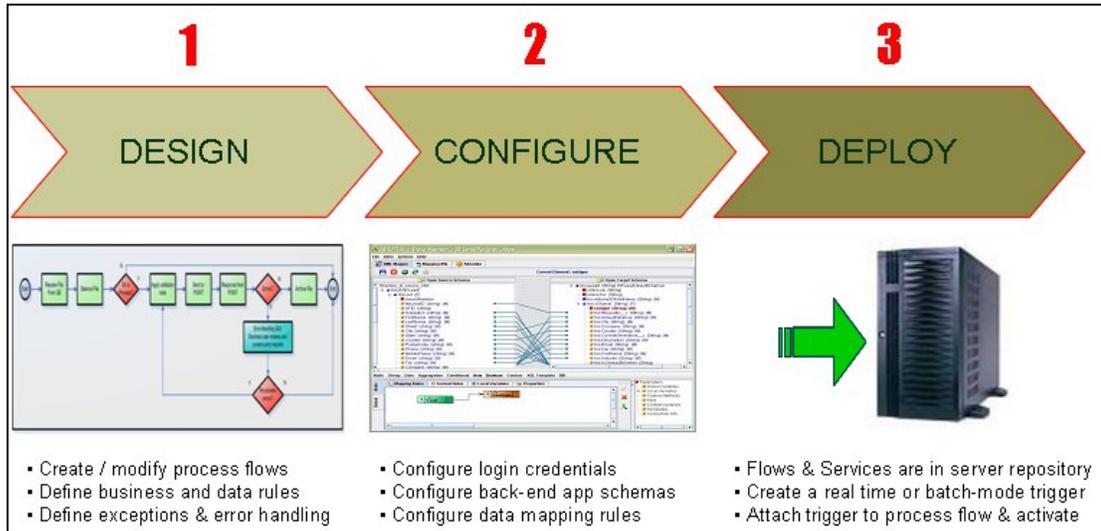
DESIGN, CONFIGURE AND DEPLOY STEPS

Figure 29

Design

In the Design step, the high-level process flow is designed in the Process Designer graphically to describe the entire flow. This may be done either by a Business Analyst or an IT Developer by using the Process Designer. The process flow is not functional yet it is just a representation. The Adeptia Salesforce Integration Accelerator provides template process flows so this step can be made even easier. Users may use these templates or the sample process flows by easily modifying them to design the flows that meet their needs.

Configure

In the configure step, the individual steps in the process flow are created. This is done by specifying the rules in the Adeptia services; this creates an instance of the Adeptia service which is called an "activity". For example: Services like Database source, Database Schema, Mapping etc. are configured. Then, in the Process Designer these activities are associated with the steps in the process flow designed earlier. Adeptia Salesforce Integration Accelerator provides many pre-built activities for using in the creation of Salesforce integration solutions.

Deploy

In the Deploy step, the process flow that has been designed and configured is "activated" by attaching it to a trigger that would start the execution of the flow. Various batch and real-time triggers are available.

6

APPENDIX B: USEFUL LINKS

ADEPTIA

Technical Documentation
http://www.adeptia.com/products/documentation.html

Software Downloads
http://www.adeptia.com/products/downloads.html

Technical Support: support@adeptia.com

Salesforce Integration Accelerator Solution Page
http://www.adeptia.com/solutions/salesforce_integration_accelerator.html

Salesforce Integration Demonstration (25 minutes video)
http://www.adeptia.com/cgi-local/adp_demo_salesforce_integration.cgi

SALESFORCE

Salesforce.com Web Services Developers Guide
http://www.salesforce.com/us/developer/docs/api/index.htm

INDEX