Using the Configuration Manager for Enhanced Integration of IBM Campaign and IBM Marketing Cloud

Alice Guidotti
Giulia Carnevale
Raimondo Castino

The IBM® Campaign integration with IBM Marketing Cloud (also called IBM Silverpop Engage®) provides digital marketers with the ability to personalize and track each customer interaction, communicate over multiple channels, and protect sensitive personal data. The Campaign integration with the Engage solution set is based on a package of downloadable scripts. These scripts are configured to enable a secure, automated exchange of segmentation, contact, and tracking data between Campaign and Engage. The configuration manager for the enhanced integration of IBM Campaign and IBM Marketing Cloud is a web-based application tool that uses the basic integration scripts. This tool provides an interface for easily managing and simplifying the upload of Campaign segmentation and contact data to Engage databases and contact lists.

Overview

The IBM Campaign base integration with IBM Marketing Cloud (also called IBM Silverpop Engage) provides digital marketers with the ability to personalize and track each customer interaction, communicate over multiple channels, and protect sensitive personal data.

The basic Campaign integration with the Engage solution set is based on a package of downloadable scripts. The scripts are configured to enable a secure, automated exchange of segmentation, contact, and tracking data between Campaign and Engage. You can run data upload and download scripts from a command line or automate the data exchange by adding the scripts to Campaign flowcharts. For more details, download the base integration guide, which is linked to in the Resources section.

Using a Campaign flowchart and a stored trigger to automatically upload data from Campaign to Engage through these scripts involves a series of procedures that might be difficult for marketers to implement. For instance, a user needs to specify several IDs to identify contact lists, the parent database, a mailing template, and a mailing recipient list. To determine these IDs, the user must log in to Engage, go to the appropriate section, and search for the corresponding item.

The configuration manager for the enhanced integration of IBM Campaign and IBM Marketing Cloud is a web-based application tool that uses the basic integration scripts. This tool provides an
interface for easily managing and simplifying the upload of Campaign segmentation and contact
data to Engage databases and contact lists.

The enhanced integration provides a new web user interface through a Java™ EE application that
provides digital marketers the ability to rapidly run data uploads through uploading configurations that
are easy to create and edit. These configurations consist of a set of human readable settings that
are interpreted at run time for uploading contacts data from Campaign to Engage. Optionally, the
settings can be configured to request the Silverpop server to send emails to a specific recipient list.

Please refer to Silvia Bellucci (silviabellucci@it.ibm.com) or Alice Guidotti
(alice.guidotti@it.ibm.com) to request the enhanced integration package.

**Figure 1. Outline**

Developed in Python, Java, and JavaScript™, the provided solution enhances the base integration
of IBM Campaign and IBM Marketing Cloud. The enhanced integration uses the command line
scripts of the base integration and improves the user experience through an intuitive user interface
by using Web 2.0 technologies.

With the web-based tool of the enhanced integration, users can:

1. Create, view, update, duplicate, or delete uploading configurations.
2. Validate configurations against data in Silverpop Engage.
3. Create and view contact lists in Silverpop Engage.
4. Use automatically created triggers to easily upload Campaign data to Engage.

**Architecture**

The Configuration Manager for Enhanced Integration of IBM Campaign and IBM Silverpop Engage
is based on two custom software components: a Java EE web application that provides the new
web-based user interface to manage uploading configurations and a set of Python scripts that
provide the interface for uploading contact data to the Engage server.

The web application consists of a web archive module (WAR) that must be deployed onto the
Java EE server where IBM Campaign is installed. This environment is the local environment that
hosts the Campaign Analytics server. The enhanced integration manager works on installations of
IBM Campaign on IBM WebSphere® Application Server and Oracle WebLogic Server on UNIX®,
Linux®, and Windows systems.
The Python scripts implement the uploading engine and call the scripts of the base integration. These scripts are installed on the same local Campaign environment where you installed the base integrated solution.

**Figure 2. Architecture diagram**

![Architecture diagram](image)

**User's guide**

From the console, users can create, delete, or modify configurations that are used for uploading contacts from Campaign to Silverpop. Users can also create new contact lists without needing to access the Silverpop console online.

To simplify the user experience, the web console that is provided by this application can be included in the IBM Campaign interface by using a custom portlet or dashboard. In this way, all operations that are performed on the local side can be easily found from within the IBM Campaign UI. Users must log in by using their personal IBM Marketing Cloud account to view their Silverpop contact list from within IBM Campaign.

After the configurations are created, users can follow instructions similar to the ones described in the base integration guide to automatically upload data from Campaign to Engage. It is a set of procedures that involve flowcharts that contain Mail List processes and flowchart triggers. Basically, the user adds a **Mail List process** in a Campaign **flowchart** to assemble the output of upstream flowchart processes into a **tab-delimited file** that can be uploaded to Engage. The new enhanced integration creates the trigger and the tab-separated file that have only to be selected when the **Mail List** process for the contact upload is configured.

The following sections provide more details on the functionalities of the web application console.

**Managing uploading configurations**

1. **Log in to the web application**
By default, the web configuration tool can be accessed at the address of your Campaign server. For example, access by using a secure HTTPS connection:
https://<campaign-server>:<port>/SilverpopIntegrationConsole/login.xhtml

To access the web application console you must authenticate by using the credentials of any standard Silverpop user (the server needs access to internet).

You can use your personal Silverpop account. You are not required to know or use the credentials that are configured for running the scripts (contactUpload.bat) that are set when you installed the files of the base integration (that is, the Integration User).

2. Viewing contact lists in Silverpop
Figure 4. Contact Lists UI

You can directly view all the existing shared contact lists in Silverpop by browsing the folder structures in the Contact Lists tab without logging in to the Silverpop console.

3. Creating a new contact list in Silverpop
Figure 5. Creating a new contact list

You can also create a new contact list, without leaving the web configuration tool, by clicking *Create Contact List* on the Contact Lists tab.

Note. The tool does not provide all the contact list management functions of the standard Silverpop console. In particular:

- the contact lists are created in the root folder, and
- deletion of contact lists is not supported.

4. **Viewing existing uploading configurations**
On the Configurations tab, you can easily view the settings of all the available uploading configurations.

Configurations consist of a set of human readable settings that can be used at run time for uploading contacts data from Campaign to Engage. Optionally, the settings can be configured to request the Silverpop server to send emails to a specific recipient list.

5. **Managing configurations: Create, update, delete**
You can create, update, duplicate, and delete configurations in an easy way. By clicking *Create Configuration*, a tool opens and guides you to select the database, contact lists, uploading options, and, optionally, the mailing template and recipient list.

6. **Validating existing configurations**
Figure 8. Validating existing configurations

By clicking **Validate Configuration**, you can validate the list of saved configurations to verify whether there are any inconsistencies with the data currently registered in Silverpop. Validation messages make it possible to easily detect anomalies without running the configurations. In this way, you are able to anticipate problems at run time and rapidly apply the necessary changes.

7. **Displaying the web console in custom portlets and dashboards inside IBM Campaign**
Figure 9. Displaying the dashboards within IBM Campaign

Figure 10. Managing portlets inside IBM Campaign
Optionally, you can easily configure the marketing platform portal (Campaign) to display the content of the web application tool next to the Engage console. For instance, the content can be displayed in a separate dashboard by using custom portlets. To access both applications, you need to authenticate twice (single sign-on is not supported).

Automatically uploading data from Campaign to Engage

Uploading contact data from Campaign to Engage, which optionally triggers Engage mailings after upload, requires similar preparations and flowchart configurations as described in the IBM Campaign and IBM Silverpop Engage Integration Guide (that is, the base integration). Refer to the integration guide to get more information.

In particular, in a Campaign flowchart, you can add a Mail List process to assemble the output of upstream flowchart processes into a tab-delimited file that can be uploaded to Engage. To support automatic data upload to Engage, the Mail List process works with a stored flowchart trigger.

The main difference consists in the introduction of the configurations that you create by using the web application tool. Using the web configuration tool, you do not need to find and specify the contact list, database, and mailing template IDs as parameters in the contactUpload script when you upload data from Campaign.

The web application tool also creates a stored trigger and an empty tab-separated file that can be easily identified from their names (the file names start with the configuration name). In this way, the Mail List process must be configured by selecting the appropriate tab-separated file and the appropriate trigger.
Figure 12. Automatic upload procedure

Figure 13. Mail process configuration
Installation instructions

1. Prerequisites

Before you can use the Enhanced Integration solution, you must complete the following activities:
1. **IBM Marketing Cloud:** Requires a Silverpop account to log in and establish access to an organization in Engage.

2. **IBM Campaign:** Install and configure IBM Campaign in your local computing environment (Campaign requires that you also install and configure IBM Marketing Platform). See the [IBM Campaign Installation Guide](#).

3. **IBM Campaign integration with IBM Marketing Cloud:** Install and configure the base integrated solution after you download the integration package by following the instructions in the Integration Guide. Version 1.2 of the base integration is a prerequisite.

To download the base integration package, go to the Campaign Silverpop Integration page on IBM developerWorks®.

2. **Install integration packages**
   The enhanced integration of Campaign with Engage depends on a collection of files and a WAR module that you install in the environment where Campaign is installed. The following packages must be used for completing the setup of the integration solution:
   - SilverpopIntegrationConsole.war: Java EE web archive application
   - SilverpopIntegrationExt.zip: Scripts and configuration files for Windows, UNIX, or Linux

All of the installation and configuration tasks are completed in the local Campaign environment where you installed the “base” integrated solution. This environment is on the local computer where the (WebSphere) Application Server hosting the Campaign Analytics server is installed.

3. **Install the integration scripts and configuration files**
   Follow this easy procedure on the Campaign Analytics server:
   1. Extract the compressed archive into a folder under
      ```
      ../Campaign/partition/partition1/
      ```
      The folder where you install the files (for example, campspext) is considered the `<CS_EXT_HOME>` directory.
   2. In `<CS_EXT_HOME>`, verify and set the properties defined in the `configExt.properties` file in the `conf` directory. As instructed by comments in the file, all the default values that are provided might be appropriate.
   3. In `<CS_EXT_HOME>`, configure `contactUploadExt.bat` on Windows or `contactUploadExt.sh` or UNIX/Linux in the `bin` directory. The only requirement is to set the `cs_home` variable to the path of the Campaign and Engage integration folder (base integration).

4. **Set up the Java property cs_ext_home**
   On the WebSphere Application Server:
   1. Log in to the administrative console of the WebSphere Application Server of the Campaign Analytics server (for example, `https://unicavmdemo:9060/ibm/console`).
   2. **Add the Java system property cs_ext_home.** Go to `Servers > Server Types > WebSphere application servers > server1`, then under Server Infrastructure go to Java and Process Management > Process definition > Java Virtual Machine > Custom properties (under Additional Properties). Select New... and set a property `cs_ext_home` specifying the absolute path to where the script files are installed on the Campaign Analytics server.
3. After you click OK, save the changes to the master configuration.

On the WebLogic Application Server:
1. Locate the file `startWeLogic.cmd` on Windows systems or `startWeLogic.sh` on Linux systems, and open it for editing.
2. **Add the Java system property `CS_EXT_HOME` to the `JAVA_OPTIONS`.** Set the property `CS_EXT_HOME` specifying the absolute path to where the script files are installed on the Campaign Analytics server:
   ```
   Append -DCS_EXT_HOME=<your_cs_ext_home_fullpath> to the set JAVA_OPTIONS= command line.
   ```
3. Save the file.

5. **Install the Silverpop SSL certificate**

   On the WebSphere Application Server:
   1. If you are not already logged in, log in to the administrative console of the WebSphere Application Server of the Campaign Analytics server (for example, `https://unicavmdemo:9060/ibm/console`).
   2. Install the Silverpop certificate in the WebSphere truststore. Go to Security > SSL certificate and key management > key stores and certificate > [NodeDefaultTrustStore] > Signer certificates, select Retrieve from port, enter the following information, and select Retrieve signer information: `host1=api0.silverpop.com; port=443; alias=Silverpop`.
      Note. The host address that is given is the default Silverpop address. Enter the value for the pod that is assigned to your Silverpop organization. See your organization administrator for this value.
   3. Press OK.

   On the WebLogic Application Server:
   1. Manually add the Silverpop certificate to the keystore by downloading the Silverpop SSL certificate from `https://engage0.silverpop.com` by using a web browser.
      Note. The URL that is given is the default Silverpop address. Enter the value that is assigned to your Silverpop organization. See your organization administrator for this value.
   2. Add the certificate to the WebLogic keystore. For more information, see the WebLogic documentation.
   3. Disable the host name verification by locating the file `startWeLogic.cmd` on Windows systems or `startWeLogic.sh` on Linux systems, and open it for editing.
   4. Append `-Dweblogic.security.SSL.ignoreHostnameVerification=true` to the `JAVA_OPTIONS` in a similar way you did for the `CS_EXT_HOME` variable.
   5. Save the file.

6. **Deploy the web application on the application server**

   On the WebSphere Application Server:
   1. If you are not already logged in, log in to the WebSphere administrative console (for example, `https://unicavmdemo:9060/ibm/console`).
   2. **Deploy the web application.** Go to Applications > Application Types > WebSphere enterprise applications, select "Install", and proceed with the steps to install a web
application. Specify SilverpopIntegrationConsole.war as the WAR module to upload and install.

3. Since the Campaign Analytics server needs to be restarted for all the previous changes to take effect, restart the WebSphere Application Server now.

On the WebLogic Application Server:
1. If you are not already logged in, log in to the WebLogic administrative console (for example, http://unicavmdemo:7001/console/login).
2. **Deploy the web application.** Go to Deployments, choose Install, and proceed with the steps to install a web application. Specify SilverpopIntegrationConsole.war as the WAR module to upload and install.
3. Restart the WebLogic application server to ensure that the changes are propagated.

**Troubleshooting**

**Enable the WebSphere logging message.** Log in to the administrative console of the WebSphere Application Server of the Campaign Analytics server (for example, https://unicavmdemo:9060/ibm/console). Go to Troubleshooting > Logs and trace > server1 > Diagnostic trace service > Change log detail levels and specify the following log detail levels: *=info; SSL_*=all.

**Debugging the trigger behavior.** The trigger, which is automatically created by the web application tool, uses a new command line (contactUploadExt), which is simplified in terms of the command line used by the base integration. Running the contactUploadExt script manually can be useful during testing and troubleshooting. Copy the trigger definition to a command line on the Campaign Analytics server, and check its output.

**Debugging the trigger by using the command line.** The new command script, contactUploadExt, supports only two main and mandatory parameters (a mapping file is not supported).

**Manual about contactUploadExt**

```
2016-07-15 05:47:04,073 INFO  Running contactUploadExt...
```

**contactUploadExt.bat/sh**

Purpose: Use contactUploadExt to upload a tab-separated file that contains contact and segmentation data from tables in IBM Campaign to databases and contact lists in IBM Silverpop Engage. The command matches fields in Campaign to corresponding fields in Engage and uses configuration settings grouped and managed together through the web application of the Campaign Integrated Console for Silverpop.

Usage:
```
contactUploadExt -i <inputFile> -n <configDataName> [-c <configPropertiesFile> -s <spPropertiesFile> -j <jdbcPropertiesFile>]
```

- `-i | --inputFile` Tab-separated file from Campaign.
- `-n | --configDataName` Name for a saved configuration that specifies:
  - `--listID` Name for Engage database that receives the data.
  - `--contactListIDs` Names for one or more contact lists in Engage.
Using the Configuration Manager for Enhanced Integration of IBM Campaign and IBM Marketing Cloud

--importType
By default, the import adds and updates records (specify OPT_OUT to upload only opt-out records).

--mailingTemplateID
Send mailing after upload. Specify template name.

--recipientListID
Contact list name for list of mailing recipients.

--removeContactData
Remove existing data from contact list, replace with data from input file.

The following options are passed to and processed by contactUpload.bat/sh:

- -c | --configPropertiesFile
Absolute or relative path to config.properties.

- -s | --spPropertiesFile
Absolute or relative path to sp.properties.

- -j | --jdbcPropertiesFile
Absolute or relative path to jdbc.properties.

Using an appropriate configuration you can:
- Upload to an Engage database
- Upload to a single contact list
- Upload to multiple contact lists
- Upload to contact list(s) and trigger a mailing to a single contact list

Examples:

Upload to Engage and optionally trigger mailing using data in a configuration
contactUploadExt.bat/sh -i <inputFile> -n <configDataName>
<CS_EXT_HOME>\bin\contactUploadExt.bat -i C:\IBM\Campaign\partitions\partition1\CSint\Uploads\FC_out.tsv
-n "Premium teams"

Upload to an Engage, renamed properties file in a new location
contactUploadExt.bat/sh -i <inputFile> -n <configDataName> -s <spPropertiesFile>
<CS_EXT_HOME>\bin\contactUploadExt.bat -i C:\IBM\Campaign\partitions\partition1\CSint\Uploads\FC_out.tsv
-n "All employees" -s C:\IBM\Campaign\partitions\partition1\CSint\CS_admin\SPaccessProps.properties

NOTE: A mapping file (.xml) defines the mapping between fields in the input file and fields in the Engage database. The file specifies also the database ID, and optionally, the contact list ID and import type (OPT_OUT). For this reason, if you want to specify a custom mapping file (-m) not use this script but use the standard script <CS_HOME>\bin\contactUpload.bat.

contactUploadExt mandatory parameters

contactUploadExt -i <inputFile> -n <configDataName>

- -i | --inputFile
Tab-separated file from Campaign.

- -n | --configDataName
Name for a saved configuration

<table>
<thead>
<tr>
<th>options</th>
<th>contactUpload (Enhanced Integration)</th>
<th>contactUploadExt (Enhanced Integration)</th>
</tr>
</thead>
<tbody>
<tr>
<td>-i</td>
<td>(Required) Path to the tab-separated file that contains the list of values to be uploaded to Engage.</td>
<td>Required option (same meaning as in contactUpload).</td>
</tr>
<tr>
<td>-n</td>
<td>Not supported.</td>
<td>The name used to identify a configuration (persisted in the JSON file).</td>
</tr>
<tr>
<td>-l</td>
<td>(Required) ID of the Engage database that receives the uploaded values. Specify the ID as defined in the Engage interface.</td>
<td>You specify the database ID in a configuration.</td>
</tr>
<tr>
<td>-m</td>
<td>Absolute or relative path to a file (.xml) that defines the mapping between fields in the input file and fields in the Engage database.</td>
<td>Not supported.</td>
</tr>
<tr>
<td>-t</td>
<td>If no value is specified, the system adds and updates records in the specified Engage database with data from the input file (-i). All input values are considered Opt-In contacts. If you specify (-t OPT_OUT), the uploaded records that match existing records are added as Opt-Out contacts.</td>
<td>You specify the import type in a configuration.</td>
</tr>
</tbody>
</table>
### Conclusion

In conclusion, the Configuration Manager for Enhanced Integration of IBM Campaign and IBM Marketing Cloud is a useful graphical tool that supports automatic uploading of contact data from IBM Campaign to IBM Marketing Cloud.

Because the Marketing Platform portal of IBM Campaign can be configured with custom portlets and dashboards, the new feature can be provided from within the portal itself, thus providing an improved user experience.
Related topics

- Visit the IBM Campaign Version-independent Integration with IBM Engage Integration Guide overview page.
- Read the IBM Campaign Version-independent Integration with IBM Engage Integration Guide.
- For more information about the Campaign Silverpop Integration, see the IBM ExperienceOne Solution Engineering community on developerWorks.
- For more information about Campaign, see the IBM Campaign documentation on the IBM Knowledge Center.
- Explore Cloud Computing on developerWorks.
- Direct Integration with WebSphere Commerce provides information on the direct integration model of OMS and WebSphere Commerce.

© Copyright IBM Corporation 2016
Trademarks
(www.ibm.com/developerworks/ibm/trademarks/)