BUDDY MEDIA DATA ADAPTOR – DEVELOPER GUIDE

What is a data adaptor?

Data Adaptors, commonly known as Plugins, allow you to reformat your lead generation data prior to sending it to your end-point of choice. You'll be able to send the data in a pre-defined XML or JSON format to your own CRM or email tools.

How is the data reformatted in your platform?

You'll provide an XSL template that we'll apply to your lead generation data. After reformatting the data will use HTTP POST to send data to your desired destination.

When I send data to my system I'll need a token, how do I do this?

You can insert a token into your XSL template so that it will send with each POST message.

What does Form Builder data look like before it's been transformed?

```
POST /path/path HTTP/1.1
Host: someURL.com

oid=VALUE_OF_ID&amp;LeadSource=VALUE_OF_SOURCE&amp;facebook_user_id=VALUE_OF_FACEBOOK_ID
```

What is an example of XSL to transform Form Builder data?

example 1: Salesforce Web to Lead

```
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
<xsl:output method="text/html" omit-xml-declaration="yes" />
<xsl:template match="/">POST /servlet/servlet.WebToLead?encoding=UTF-8 HTTP/1.1
Host: https://www.salesforce.com<![CDATA[

]]>oid=<xsl:value-of select="root/org_id"/>&amp;
LeadSource=<xsl:value-of select="root/lead_source"/>&amp;
custom_field=<xsl:value-of select="root/form_builder_field_label"/>
</xsl:template>
</xsl:stylesheet>
```

example 2: YesMail with a token - Token replaced with asterisks for example

```
<xsl:stylesheet version="1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
  <xsl:output method="text/html" omit-xml-declaration="yes" />
```

```
   <xsl:template match="/">POST /enterprise/composite/subscribeAndSend HTTP/1.1
Host: https://services.yesmail.com
Content-Type: application/json
Authorization: Basic ***************** <![CDATA[

]]]]>{"subscriber":
    {
       "subscriptionState" : "SUBSCRIBED",
        "name":"allowResubscribe","value":"true",
         "division" : {"value" : "Transactional" },
             "attributes" :
              {
                 "attributes" :
                  [
                    {"name" : "email", "value" :"<xsl:value-of select="root/email"/>"},
                    {"name" : "firstName","value" : "<xsl:value-of select="root/first_name"/>"},
                    {"name" : "lastName","value":"<xsl:value-of select="root/last_name"/>"},
                    {"name":"postalCode","value":"<xsl:value-of select="root/postal_code"/>"},
                    {"name":"prefersMobile","value":"-1"},
                    {"name":"emailFormat","value":"HTML"}
                  ]
               }
         },
  "subscriberMessage":{"masterId":"<xsl:value-of select="root/master_ID"/>"}
}
</xsl:template>
</xsl:stylesheet>
```

Example 3: Simple POST without Token

```
<xsl:stylesheet version="1.0"
 xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
 <xsl:output method="text/html" omit-xml-declaration="yes" />
  <xsl:template match="/">POST /iphoneservices/registeruser.ashx? HTTP/1.1
Host: https://www.test.com
Content-Type: application/x-www-form-urlencoded <![CDATA[

]] ]]> facebook_page_name=<xsl:value-of select="root/page_name"/>&amp;first_name=<xsl:value-of
select="root/first_name"/>&amp;last_name=<xsl:value-of
select="root/last_name"/>&amp;email=<xsl:value-of
select="root/email"/>&amp;mobile_phone=<xsl:value-of select="root/phone"/>
</xsl:template>
</xsl:stylesheet>
```

What fields are the values I want to send versus the ones from the form?

The first variable represents the key in the key:value pair (JSON) or the attribute (XML).

Using the JSON example below, let’s break down the components.

```
<xsl:stylesheet version="1.0"
 xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
 <xsl:output method="text/html" omit-xml-declaration="yes" />
  <xsl:template match="/">POST /enterprise/composite/subscribeAndSend HTTP/1.1
Host: https://services.yesmail.com
Content-Type: application/json
Authorization: Basic ****************** <![CDATA[

]]]]>{"subscriber":
    {
      "subscriptionState" : "SUBSCRIBED",
       "name":"allowResubscribe","value":"true",
        "division" : {"value" : "Transactional" },
           "attributes" :
            {
              "attributes" :
               [
                {"name" : "email", "value" :"<xsl:value-of select="root/email"/>"},
                {"name" : "firstName","value" : "<xsl:value-of select="root/first_name"/>"},
                {"name" : "lastName","value":"<xsl:value-of select="root/last_name"/>"},
                {"name":"postalCode","value":"<xsl:value-of select="root/postal_code"/>"},
                {"name":"prefersMobile","value":"-1"},
                {"name":"emailFormat","value":"HTML"}
               ]
            }
        },
 "subscriberMessage":{"masterId":"<xsl:value-of select="root/master_ID"/>"}
}
</xsl:template>
</xsl:stylesheet>
```

breaking it down:

1. Define as XSL & Provide a Header:

```
<xsl:stylesheet version="1.0"
 xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
  <xsl:output method="text/html" omit-xml-declaration="yes" />
    <xsl:template match="/">POST /enterprise/composite/subscribeAndSend HTTP/1.1
Host: https://services.yesmail.com
Content-Type: application/json
Authorization: Basic ******************
```

2. Provide a new line for the payload:

```
<![CDATA[
]]]]>
```

3. Define a key:

```
 "name" : "email"
```

4. Pull in the field from form builder:

```
<xsl:value-of select="root/email"/>
```

Note: What you define within value-of does not have to match the actual field's variable name. This mapping will be finalized by the user on the form.

When should I set a field as configurable?
Set a field as configurable when this data can change based on where the plugin is used. For example, you may want a field that defines if the data's source is Desktop or Mobile. Set your 'source' field to configurable so that Marketers can create an instance for desktop forms that will set 'source' to 'desktop', and a second instance where 'source' will be sent as 'mobile'

Can my plugin be reused on two separate accounts that I want to send form data to?
By defining selected variables as configurable *(e.g., credentials or a campaign ID)* when creating your plugin, you'll allow marketers to create multiple "instances" of your plugin with different configurations.

Can an adaptors be setup with multiple credentials or meta-data?
Yes, configurable fields can be defined within an adaptor. This allows the client to create an "instance" of the adaptor with their own configuration.

For example, an adaptor for MailChimp could be created for the channel with configurable fields for user credentials and a campaignID. Each client could then create their own instance that connected to their MailChimp account along with a specific campaign ID.

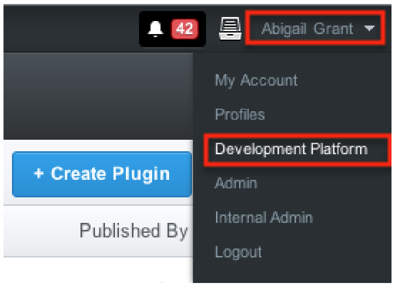
What happens if an adaptor fails?
Developers are responsible for collecting errors with transmission of data to their end-point. If, for example, form data fails to store in MailChimp because a user's credentials are no longer correct, the adaptor's developer should log this error against his adaptor.

However, all responses from the 3rd party end point will be recorded for the developer in our development platform. The develop will be able to view the payload in failed responses.

If an adaptor fails, is the data available in Buddy Media? Can I resend failed data to the adaptor?
Yes, data sent to an adaptor is always stored at Buddy Media. Since transmission is real-time data that has failed cannot be resent to the adaptor.

Where and how can I create an adaptor?

To get access to the developer platform send a request to our Product Support Team. Once enabled you can access the platform from under your name, and follow the Data Adaptor guide on pages 1- 3. https://na1.salesforce.com/sfc/#version?selectedDocumentId=06930000003Gl7g



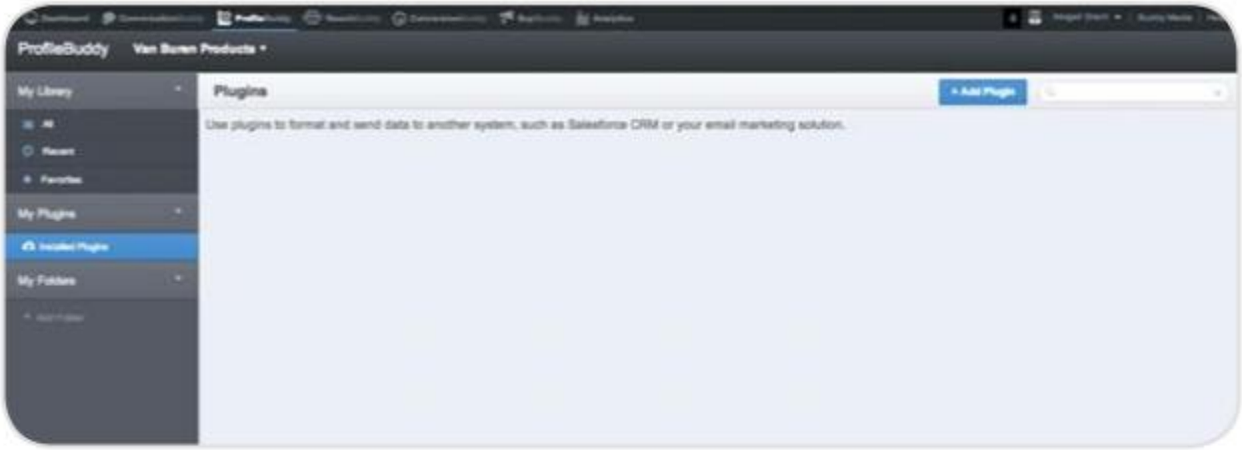
MARKETER'S GUIDE

I'm done now what?

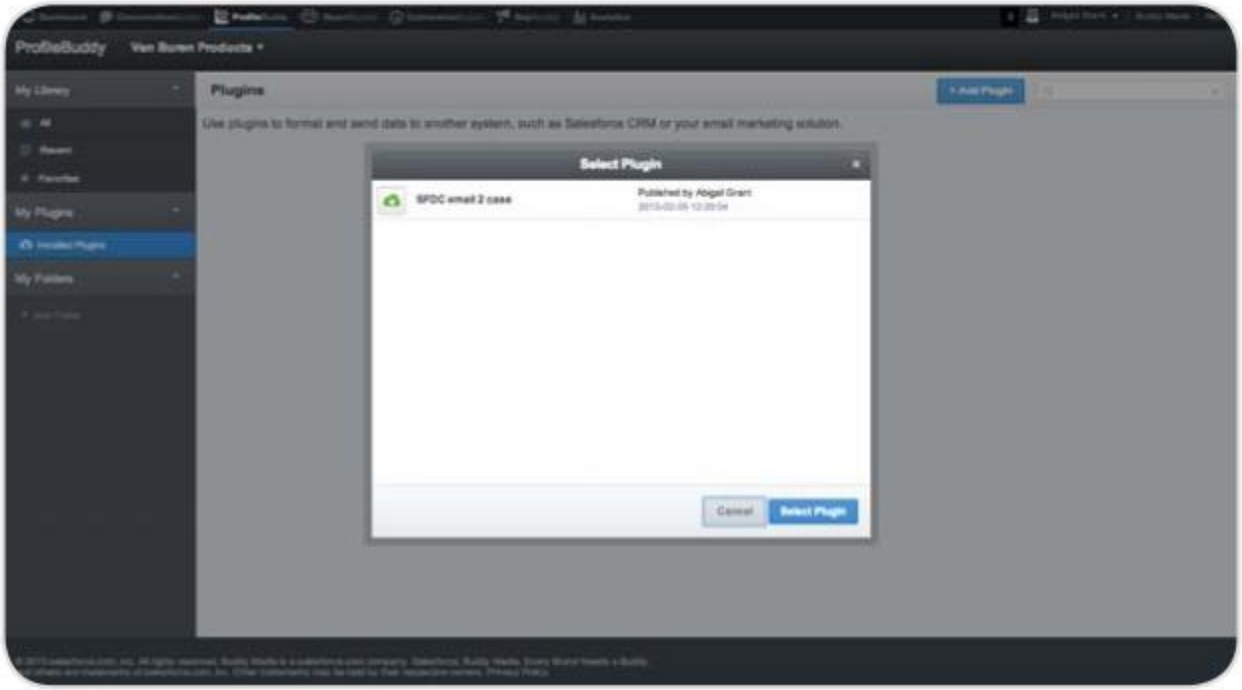
It's time to hand it over to the marketing team. They will want to install your plugin and associated it with their lead generation form.

How do I install?

From the ProfileBuddy or ReachBuddy main page you'll see a tab named Installed Plugins. Click the '+ Add Plugin' button to select & install your plugin.



Use plugins to format and send data to another system, such as Salesforce CRM or your email marketing solution.



Use plugins to format and send data to another system, such as Salesforce CRM or your email marketing solution.

How do I associate an adaptor plugin with a form builder sapplet?
All form builder sapplets will have a new section for plugins. Within the sapplet select the appropriate plugin instance.
Once you've selected the right plugin, you'll see all fields being captured by the plugin. Use the dropdown fields to associate the plugin's fields with fields you've already created in form builder.

How do I stop using a plugin?
In the plugins tab simply select "none" and save to stop sending form builder data through the adaptor.