# **EDQ for Salesforce Commerce Cloud**

Version 19.1.0



# 1. Summary

The EDQ Commerce Cloud cartridge verify address and validates phone and email data in real time during registration, shipping, billing and address touchpoints; this cartridge supports both SiteGenesis (SGJC) and Storefront (SFRA) workflows.

## 2. Component Overview

#### 2.1. Functional Overview

The EDQ Commerce Cloud cartridge contains address verification with Pro Web - Address (Verification Engine) and Global Intuitive, as well Global Phone Validate and Email Validate functionality. The sections below describe each of these workflows in more detail.

#### 2.1.1. Email Validate

Email Validation will check your email at the moment you finish typing in the email field box and once you lose focus on the email field the Email Validate will start working letting you know if is a valid or an invalid email.

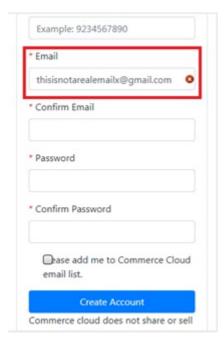
To configure your Email Validate, you need to access the Business Manager and go to EDQ Config (Select Site > Merchant Tools > Site Preferences > Custom Site Preferences Group > EDQ Config) and look for the email options.



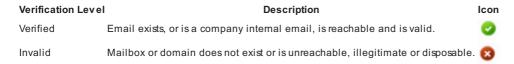
- The email enable option will enable/disable Email Validate in your store.
- The email validation option will enable/disable the page to prevent on going through.

The store touchpoints for Email Validate are:

- Registration form.
- · Billing/Payment form.



Email validate can result in two potential outcomes.



### 2.1.2. Global Phone Validate

Global Phone Validate will check your phone at the moment you finish typing in the phone field box and once you lose focus on the phone field, the Global Phone Validate will start working letting you know if is a valid or an invalid phone.

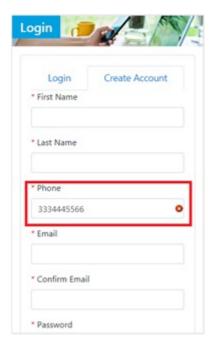
To configure your Global Phone Validate, you need to access the Business Manager and go to EDQ Config (Select Site > Merchant Tools > Site Preferences > Custom Site Preferences Group > EDQ Config) and look for the phone options.



- The phone enable option will enable/disable Global Phone Validate in your store.
- The phone validation option will enable/disable the page to prevent on going through.

The store touchpoints for Global Phone Validate are:

- Registration form.
- Add/Edit Address form.
- Billing/Payment form.



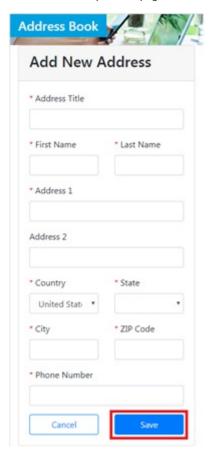
Global Phone Validate can result in two potential outcomes.

 Verification Level
 Description
 Icon

 Verified
 Phone matched to a high confidence and returned as a valid number.
 Invalid
 Could not match the number and returned as an invalid number.
 X

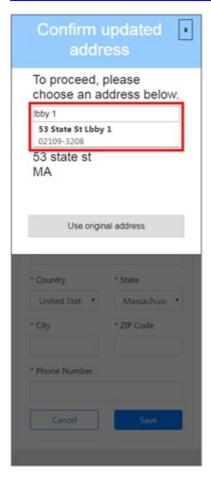
## 2.1.3. Pro Web - Address (Verification Engine)

Pro Web - Address (Verification Engine) will check your address and correct if possible; once you type your address in the respective address fields and press the page form submitting button.

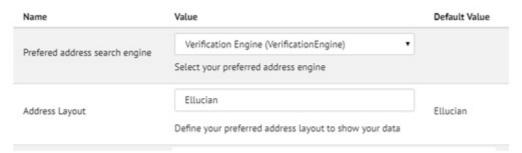


In case that your address needs to be corrected and does not match an address; Pro Web - Address (Verification Engine) will prompt a "confirm updated address" box to complete your address in real time; by typing down the address in the address input field, it will give you options with accurate addresses matching the address with the information provided by the user and finally by selecting one of the options it will autocomplete the addresses fields in the address form with correct and valid address and may continue with the store process.

For more information about Pro Web - Address (Verification Engine). (https://www.edq.com/documentation/apis/address-validate/soap/)



To configure Pro Web - Address (Verification Engine), you need to access the Business Manager and go to EDQ Config (Select Site > Merchant Tools > Site Preferences > Custom Site Preferences Group > EDQ Config) and look for the verification engine options.



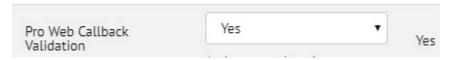
- The preferred address search engine option will let you choose between the address search engine options (choose verification engine to only use this selected engine).
- The address layout option lets you choose the structure in which your address will be returned.
- The Pro Web callback validation will let users specify a custom function to manage page transitions.
- The Pro Web Custom Transition Callback will let you specify a name of a function that will be called in order to define custom transition logic when using Pro Web Address (Verification Engine).

The store touchpoints for Pro Web - Address (Verification Engine) are:

- Add/Edit Address form.
- Billing/Payment form.

**Pro Web Custom Transition Callback** 

To enable the Pro Web Callback Transition go to the Business Manager and go to the EDQ Config (Select Site > Merchant Tools > Site Preferences > Custom Site Preferences Group > EDQ Config) and look for the Pro Web Callback Validation option and set the value to true.



After enabling the Pro Web Custom Transition Callback, set the value of the Transition Callback Name with the name of your custom function that will be executed prior to transitioning.



Since all logic is in the custom function you need to take in consideration the following points to create your custom function:

 Your custom function should receive a parameter; this is where we are going to send you a JSON with the data response from Pro Web Address (Verification Engine).
 e.g.

```
function myCustomFunction(myParameter) {
    ....
}
```

The response data from the variable that you're going to receive should look like this example (take note that the way that the data is shown depends on your layout configuration in the Business Manager).

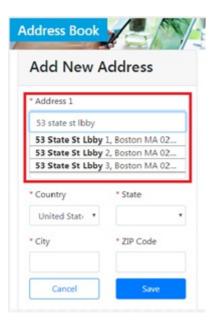
```
+4 code: "3208"
City name: "Boston"
Formatted Address 1: null
Formatted Address 2: "53 State St Lbby 2"
Formatted Address 3: "Boston MA 02109-3208"
PO Box: null
PO Box (Number): null
PO Box (Text): null
Primary number: "53"
Secondary number: "Lbby 2"
Secondary number (Number): "2"
Secondary number (Type): "Lbby"
State code: "MA"
State name: "Massachusetts"
Street: "State St"
Street (Descriptor): "St"
Street (Name): "State"
Street (Post-directional): null
Street (Pre-directional): null
ZIP Code: "02109"
```

Your custom function should return a boolean (true or false), depending on your function logic; if your function returns true the
page will continue with its normal workflow, if your custom function returns false the page won't make a transition to the next
one.

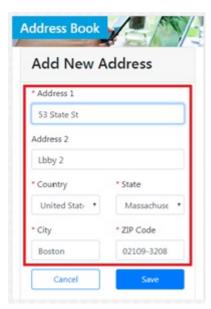
```
if (customLogic)
    return true;
else
    return false;
```

## 2.1.4. Global Intuitive

Global Intuitive will correct your address in real time while your typing down your address in the address field box.

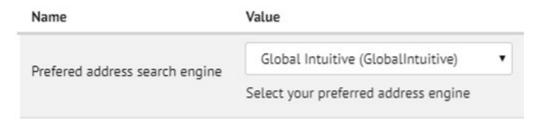


Global Intuitive will automatically fill the rest of the address fields in the form.



For more information about Global Intuitive. (https://www.edq.com/documentation/apis/address-validate/global-intuitive/)

To configure Global Intuitive, you need to access the Business Manager and go to EDQ Config (Select Site > Merchant Tools > Site Preferences > Custom Site Preferences Group > EDQ Config) and look for the global intuitive option.



- The preferred address search engine option will let you choose between the address search engine options (choose verification engine to only use this selected engine).
- The Data Set Code option will let you set a code to use Data Sets in case the country ISO code requires it.
- The Activates or deactivates Data Set usage option will let you choose to activate or deactivate Data Set usage.
- The Data Set Code will let you set the Data Set that will be use for an especific country.
- The activates or deactivates Data Set usage will let you use the Data Set function; it should be false for countries that don't need Data Sets.

The store touchpoints for Global Intuitive are:

- · Add/Edit Address form.
- · Billing/Payment form.

## 2.2. Compatibility

EDQ cartridge was created with Commerce Cloud version 19.2 and may use the following touchpoints for its integration

SiteGenesis (SGJC):

- Register Form
- Address Form
- Shipping Form
- Billing Form

StoreFront (SFRA):

- Register Form
- Address Form
- Checkout Form

## 2.3. Privacy, Payment

EDQ cartridge does not have access to your payment data; it just works to verify and correct address data and verify phone and email data. The rest of the data is not checked by Experian.

# 3. Implementation Guide

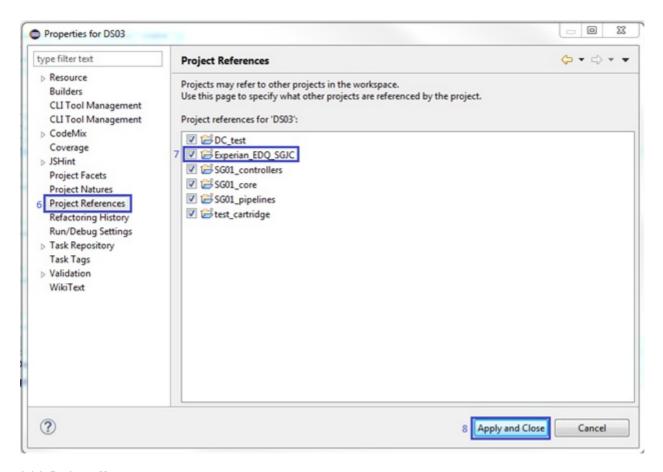
## 3.1. Setup

This integration process will require access to the following:

- Business Manager
- Commerce Cloud UX Studio

### 3.1.1. Upload a Cartridge Using Eclipse IDE

- 1. Open Eclipse IDE and click File > Import .
- 2. Select the type of project you want to add and click next.
- 3. Select the project you want to import and click "Finish" button.
- 4. Go to Project Explorer tab.
- 5. Select your digital server and go to its Properties.
- 6. On the properties window select the option "Project Reference".
- 7. Select the cartridge you want to add to your digital server.
- 8. Press "Apply and Close" button.



## 3.1.2. Business Manager

Register your cartridge.

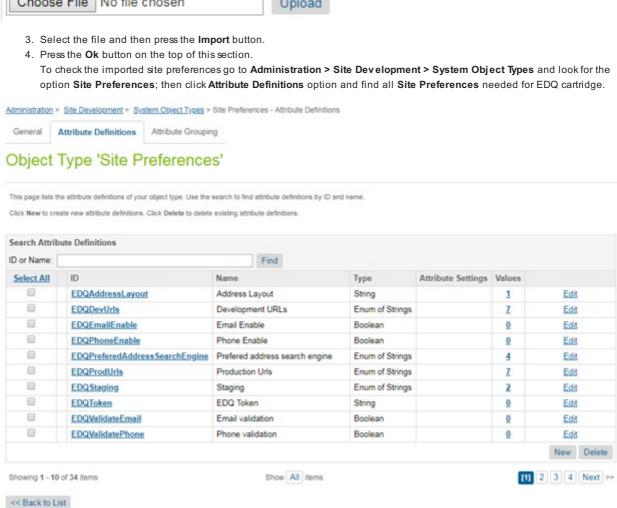
- 1. Log in to your Business Manager, then go to Administration > Sites > Manage Sites .
- 2. Look for your "site\_name" and select it.
- 3. Click on settings tab.
- 4. Add the cartridge name "int\_experian\_sgjc" (if you're using SiteGenesis or choose int\_experian\_sfra if you're using SFRA) into the "Cartridges" input-box (cartridges take preference from left to right).
- 5. Click "Apply" button.



#### 3.1.3. Site Preferences

- 1. Log into your Business Manager and go to Administration > Site Development > Site Import & Export .
- 2. On the Import section with the tag name Upload Archive check the Local option then select the file "SitePreferences.zip"; then press the Upload button.





## 3.2 Custom Code

### 3.2.1. Sitegenesis JavaScript Controller

### **EDQHeaders** integration

- Go to int\_experian\_sgjc cartridge and open the template located in templates > default > components > header > htmlhead\_UI.isml
- 2. The header line has been added at the end of the file "htmlhead\_UI.isml" file <isinclude template="EDQ/EDQHeaders" />

```
htmlhead ULisml SS
  1 <iscontent type="text/html" charset="UTF-8" compact="true"/>
2 <iscomment>THIS FILE ONLY CONTAINS DIFFERENCES BETWEEN SIMPLE AND RICH UI LIB INCLUDES</iscomment>
  3 <iscomment>
  4 The <!å€"BEGIN/END倦 comments are control statements for the build cartridge which can be found in xChange
  5 https://xchange.demandware.com/docs/DOC-5728 or checked out from SVN at
  6 https://svn2.hosted-projects.com/cs_europe/DWTechRepository/cartridges/build_cs
  7 If you are not using the build cartridge the comments can be safely removed.
  8 </iscomment>
 10 <iscomment>THIRD PARTY STYLESHEETS STYLING</iscomment>
 11 11 11 11 k href="${URLUtils.staticURL('/lib/jquery/ui/jquery-ui.min.css')}" type="text/css" rel="stylesheet" /> 12 12 12 12 13 

 14 <! -- [if lt IE 9]>
 15 <script src="${URLUtils.staticURL('/js/lib/html5.js')}"></script>
 16 <! [endif] -->
 18 <iscomment>Salesforce Commerce Cloud active data scripts</iscomment>
 19 <isactivedatahead/>
 21 <isinclude template="EDQ/EDQHeaders" />
```

#### **EDQUnicorn integration**

- Go to int\_experian\_sgjc cartridge and open the template located in templates > default > account > addressbook > addressdetails.isml
- 2. The <isinclude template="EDQ/EDQUnicom" /> line has been added before the buttons definition in "addressdetails.isml"

```
addressdetails.isml 🛭
25
             </span>
           </isif>
26
28
          <isscript>
29
             var currentCountry = require('~/cartridge/scripts/util/Countries').getCurrent(pdict);
          </isscript>
          <isdynamicform formobject="${pdict.CurrentForms.profile.address}" formdata="${currentCountry.dynamicForms.addressDetail</pre>
31
32
33
34 +
          <div class="form-row form-row-button";</pre>
             cisinclude template="EDQ/EDQUnicorn" />
             35
36
37
                ${Resource.msg('global.apply','locale',null)}
             </button>
             38
39
40
             </button>
             <isif condition="${pdict.Action != 'add'}">
                42
43
44
                </button>
45
46
             </isif>
47
          </div>
          <input type="hidden" name="${dw.web.CSRFProtection.getTokenName()}" value="${dw.web.CSRFProtection.generateToken()}"/>
48
        </fieldset>
58
     (/form)
```

- 3. Go to int\_experian\_sgjc cartridge and open the template located in templates > default > account > user > registration.isml
- 4. The <isinclude template="EDQ/EDQUnicorn" /> line has been added before the ending tag </isdecorate> in "registration.isml" file.

```
negistration.isml 🛭
                                          <div class="dialog-required"> <span class="required-indicator">&#8226; <em>${Resource.msg('global.required-indicator")
   85
   87
                                    <isif condition="${!pdict.CurrentSession.customer.externallyAuthenticated}">
   88
                                          <isinputfield formfield="${pdict.CurrentForms.profile.login.currentpassword}" type="password" dynamicname</pre>
   89
   90
                                          <isinputfield formfield="${pdict.CurrentForms.profile.login.newpassword}" type="password" dynamicname="tricisinputfield formfield="${pdict.CurrentForms.profile.login.newpasswordconfirm}" type="password" dynamicname="tricisinputfield formfield="${pdict.CurrentForms.profile.login.newpasswordconfirm}" type="password" dynamicname="tricisinputfield formfield="${pdict.CurrentForms.profile.login.newpasswordconfirm}"</pre>
   91
   92
   93
                                    <iselse/>
                                          <isinputfield formfield="${pdict.CurrentForms.profile.customer.email}" type="input" attributes="${attributes="${attributes="$}{attributes="$}}</pre>
   95
                                    </isif>
   96
   97
                                          <button type="submit" value="${Resource.msg('global.apply','locale',null)}" name="${pdict.CurrentForms.pre
${Resource.msg('global.apply','locale',null)}
   98
   99
  100
                                          </button>
                                    </div>
 101
 102
 103
                                    <input type="hidden" name="${dw.web.CSRFProtection.getTokenName()}" value="${dw.web.CSRFProtection.generateTol</pre>
 104
 105
                              </fieldset>
 186
                        </form>
 107
                  </isif>
 189
 110 +
                 Kisinclude template="EDQ/EDQUnicorn" />
  111 </isdecorate>
 112
```

- 5. Go to int\_experian\_sgjc cartridge and open the template located in templates > default > checkout > billing > billing.isml
- 6. The <isinclude template="EDQ/EDQUnicorn" /> line has been added before the ending tag </isdecorate> in "billing.isml" file.

```
billing.isml 🖾
199
       200
           payment methods
 201
        202
203
 204
       <iscomment>payment method area</iscomment>
       <isinclude template="checkout/billing/paymentmethods"/>
205
       <isbonusdiscountlineitem p_alert_text="${Resource.msg('billing.bonusproductalert','checkout',null)}" p_discount_line_item="${p.</pre>
297
288
 209
210
               <button class="button-fancy-large" type="submit" name="${pdict.CurrentForms.billing.save.htmlName}" value="${Resource.n</pre>
           </div>
211
 212
           <input type="hidden" name="${dw.web.CSRFProtection.getTokenName()}" value="${dw.web.CSRFProtection.generateToken()}"/>
213
214
215
       <isscript>
           importScript("util/ViewHelpers.ds");
216
           var addressForm = pdict.CurrentForms.billing.billingAddress.addressFields;
var countries = ViewHelpers.getCountriesAndRegions(addressForm);
 217
218
           var json = JSON.stringify(countries);
219
220
       </isscript>
       <script>window.Countries = <isprint value="${json}" encoding="off"/>;</script>
221
223 + cisinclude template="EDQ/EDQUnicorn" />
224
 225 </isdecorate>
226
```

- Go to int\_experian\_sgjc cartridge and open the template located in templates > default > checkout > shipping > singleshipping.isml
- 8. The <isinclude template="EDQ/EDQUnicorn" /> line has been added before the ending tag </isdecorate> in "singleshipping.isml" file.

#### 3.2.2. Storefront Reference Architecture

#### **EDQHeaders** integration

- 1. Go to int\_experian\_sfra cartridge and open the template located in templates > default > common > htmlHead.isml.
- $2. \ \, \textit{The} \, \texttt{<} \textit{isinclude template="EDQ/EDQHeaders" />} \, \textit{line has been added at the end of "htmlHead.isml" file. } \, \text{ } \, \textit{line has been added at the end of "htmlHead.isml" file. } \, \text{ } \, \textit{line has been added at the end of "htmlHead.isml" file. } \, \text{ } \, \textit{line has been added at the end of "htmlHead.isml" file. } \, \text{ } \, \textit{line has been added at the end of "htmlHead.isml" file. } \, \text{ } \, \textit{line has been added at the end of "htmlHead.isml" file. } \, \text{ } \, \text{$

#### **EDQUnicorn integration**

- Go to int\_experian\_sfra cartridge and open the template located in templates > default > account > components > registerForm.isml
- 2. The "registerForm.isml" line has been added at the end of file <isinclude template="EDQ/EDQUnicorn" />
- 3. In the button section the attribute "name" with the label "save" to the submit button has been added.

```
negisterForm.isml
114
                 ${Resource.msg('button.createaccount.registration', 'registration', null)}
 115
             </button>
 116
             <div class="privacy-policy">
                 ${Resource.msg('description.privacypolicy.registration', 'registration', null)}
<a href="${'#'}">
 117
 118
 119
                     ${Resource.msg('link.privacypolicy.registration', 'registration', null)}
                 </a>
 120
            </div>
 121
        </form>
 122
 123
 124 + cisinclude template="EDQ/EDQUnicorn" />
125
```

- 4. Go to int\_experian\_sfra cartridge and open the template located in templates > default > account > addressForm.isml
- 5. In "addressForm.isml" file the <isinclude template="EDQ/EDQUnicorn" /> has been added after the submit button as shown below.
- 6. In the button section the attribute "name" with the label "save" has been added to the submit button.

```
addressForm.isml 🗮
157
         <!-- Cancel and Save Buttons -->
         158
 159
                 <a href="${URLUtils.url('Address-List')}" class="btn btn-block btn-outline-primary">${Resource.msg('button.cancel', 'account
             </div>
 161
                 cbutton type="submit" name="save" class="btn btn-save btn-block btn-primary">${Resource.msg('button.save', 'account',null)}-
kisinclude template="EDQ/EDQUnicorn" />
163 +
 165
             </div>
 166
         </div>
 167
168 </form>
```

- 7. Go to int\_experian\_sfra cartridge and open the template located in templates > default > checkout > checkout.isml
- 8. The <isinclude template="EDQ/EDQUnicorn" /> line has been added after the checkout workflow buttons in "checkout.isml" file.

```
n checkout.isml 33
                  <!-- Checkout Workflow Buttons -->
                  <div class="row";
                      <div class="col-12 next-step-button">
                         <div>
                            </button>
  82
83
84
85
86
87
88
89
90
91
                            <button class="btn btn-primary btn-block submit-payment" type="submit" name="submit" value="submit-payment">
                                                                        eckout',
                             <button class="btn btn-primary btn-block place-order" data-action="${URLUtils.url('CheckoutServices-PlaceOrder')}"</pre>
                                    type="submit" name="submit" value="place-order">${Resource.msg('button.place.order', 'checkout', null)}
                             </button>
                         </div>
                     </div>
  92
                  </div
                  <isinclude template="EDQ/EDQUnicorn" />
              c/div
```

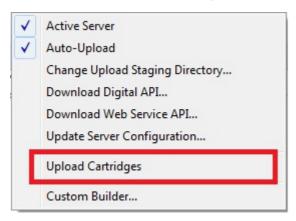
9. In the <!-- Step 1: Shipping --> section the attribute id with the label editShipping has been added to the html span tag.

10. In the <!-- Step 2: Payment and Billing--> section the attribute id with the label editPayment has been added to the html span tag.

```
checkoutisms  

cl-- Step 2: Payment and Billing-->
clsinclude template="checkout/billing/billing" />
clsinclude template="checkout/billing/billing" />
cdiv class="card payment-summary">
cdiv class="card-header clearfix">
checkout', null)}</hd>
checkout', null)}</hd>
checkout', null)}</hd>
checkout', null)}</hd>
checkout', null)}</hd>
checkout', null)}</hd>
class="edit-button pull-right">${Resource.msg('action.edit.step', 'checkout', null)}</pr>
checkout', null)}
checkout', null')
checkout',
```

- If your cloud server has the Auto-Upload option active, save all files and all new cartridges are going to be uploaded to your sandbox instance.
- If you want to upload manually from Eclipse
- 1. Click the cloud server icon
- 2. Choose the option Upload Cartridges.

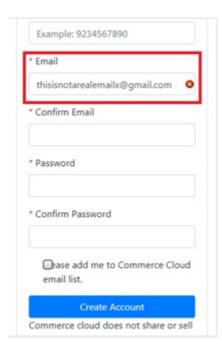


# 3.3. Testing

For both (Sitegenesis and Storefront Reference Architecture) stores the testing touchpoints are the same.

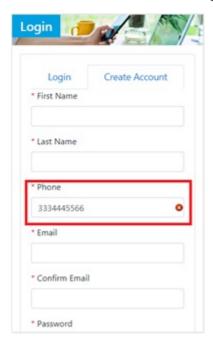
#### 3.3.1. Email Validate

In the create account page type your email in the email input field and the Email Validate will verify your email once you lose focus on the email field, the Email Validate will start working letting you know if is a valid or an invalid email.



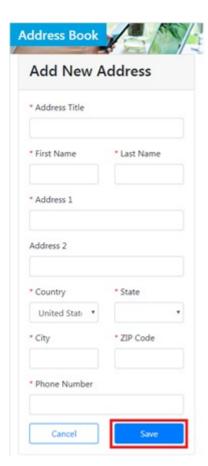
#### 3.3.2. Global Phone Validate

In the create account page type your phone and Global Phone Validate will check your phone once you lose focus on the phone field, the Global Phone Validate will start working letting you know if is a valid or an invalid phone.



## 3.3.3. Pro Web - Address (Verification Engine)

Once your logged into your store go to you profile and add new address; type down an address in the address input fields and Pro Web - Address (Verification Engine) will check and correct your address if possible, once you press the page form submitting button; Pro Web - Address (Verification Engine) will check your address and will give you the correct address for it.

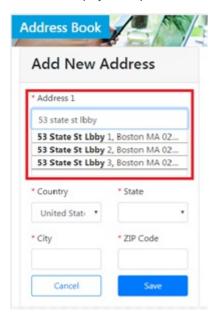


In case that your address needs to be corrected and does not match an address; Pro Web - Address (Verification Engine) will prompt a "confirm updated address" box to complete your address in real time; by typing the address in the address input field, it will give you options with accurate addresses matching the address with the information provided by the user and finally by selecting one of the options it will autocomplete the addresses fields in the address form with correct and valid address and may continue with the store process.

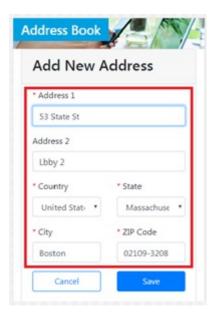


### 3.3.4. Global Intuitive

Once you're logged into your store go to profile and add new address; start typing an address in the address input field and Global Intuitive will display some options to match the address as you type it; select the correct address from the list.



And automatically global intuitive will fill the rest of the address fields in the form.



## 4. Operations, Maintenance

## 4.1. Availability

By default, each service has a timeout configuration in its code:

- Pro Web Address (Verification Engine) = 3.5 seconds
- Global Phone Validate = 3.5 seconds
- Email Validate = 15 seconds
   If you don't get an answer in that time; you can try making the request again.

### 4.2. Support

For any support contact email are

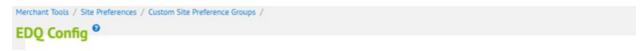
QAS Technical Support, Us.support.qas@experian.com

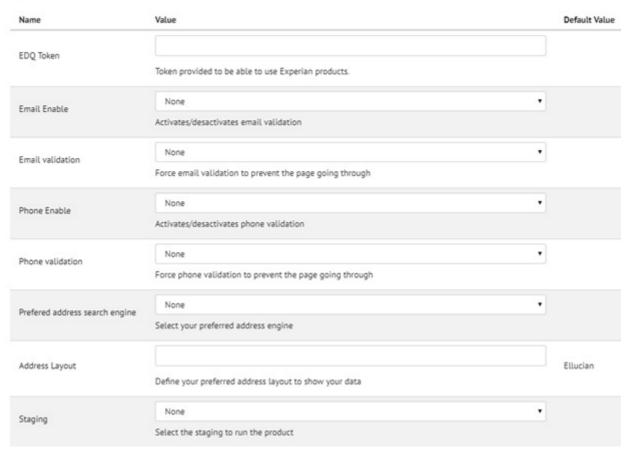
### 5. User Guide

# 5.1. Business Manager

Once the EDQ SitePreferences are uploaded in the business manager; choose your site and go to **Merchant Tools > Site Preferences > Custom Site Preferences Group** and choose EDQ Config option.

- 1. EDQ Token (String): credentials provided to use the service.
- Set to the value supplied to you by your Experian Partner
- 2. Email Enable (Boolean): set to enable/disable email validation.
- 3. Email Validation (Boolean): set enable/disable to force email validation to prevent the page going through.
- 4. Phone Enable (Boolean): set to enable/disable phone validation.
- 5. Phone Validation (Boolean): set enable/disable to force phone validation to prevent the page going through.
- 6. Preferred address search engine (Enum of String): Select one option between:
  - 1. Both
  - 2. Global Intuitive
  - 3. Verification Engine
  - 4. None
- 7. Address layout (String): Address layout format for the data.
- 8. Staging (Enum of String): Select one option between:
  - 1. Production
  - 2. Development
- 9. Data Set Code (String): Code to use Data Sets in case the country ISO code requires it.
- 10. Activates or deactivates Data Set usage (Boolean): Activates or deactivates Data Set usage (only use for Non USA).
- 11. Pro Web Custom Transition Callback (String): Name for the custom customer function to be triggered.
- 12. Pro Web Callback validation (Boolean): Activates or deactivates page transition from the callback.





### 5.3. Storefront Functionality

For each storefront touchpoint

# SiteGenesis (SGJC)

SGJC	Email Validate	Global Phone Validate	Pro Web - Address (Verification Engine)	Global Intuitive
Register Form	<b>V</b>	×	×	×
Address Form	×	<b>✓</b>	<i>V</i>	<b>✓</b>
Shipping Form	*	<b>✓</b>	<i>V</i>	<b>✓</b>
Billing Form	•	V	<b>V</b>	<b>/</b>

# StoreFront (SFRA)

SFRA	Email Validate	Global Phone Validate	Pro Web - Address (Verification Engine)	Global Intuitive
Register Form	<b>✓</b>	<b>✓</b>	×	×
Address Form	×	<b>✓</b>	<b>✓</b>	<b>~</b>
Checkout Form	<b>V</b>	<b>V</b>	<b>V</b>	~

### 6. Known Issues

• Timeout connection issues can be experience due to a slow internet connection. If you don't get an answer in the default time; you can try making the request again.

## 7. Failover and recovery

• If the unicom/pegasus libraries connection fails the store will continue with it's normal workflow; this issue may be caused due a non valid token or slow internet connection.