



Address Validation - Street Level

XML Developer Guide

January 4, 2016



Important Information

UPS Developer Kit APIs

Your development of an application using the UPS Developer Kit APIs is governed by the UPS Technology Agreement or UPS Customer Technology Agreement you entered into with UPS. The following are key legal requirements from these agreements for the UPS Developer Kit APIs. For more information on all requirements for the UPS Developer Kit APIs, please refer to the UPS Technology Agreement or the Customer Technology Agreement.

Defined terms used but not defined in this document have the meaning set forth in the UPS Technology Agreement or the Corporate Technology Agreement.

Key Legal Requirements for UPS Developer APIs

Permitted Territories

This document can only be used in the countries of the Permitted Territory as defined in the UPS Technology Agreement or UPS Corporate Technology Agreement, as applicable.

Use

The application must not be designed to allow distribution of information received through the UPS Developer Kit APIs to third parties, other than to persons having a bona fide interest in such information (e.g., the shipper, receiver, or the third party payer, or to your service providers authorized by UPS).

Consent to Use of UPS Mark

- All screens or forms generated by your application including information received through the UPS Developer Kit APIs must include (1) the UPS Mark positioned in reasonable proximity to the Information and of an appropriate size to readily identify the source of the Information as UPS and (2) the following language at the bottom of every screen that displays the UPS Mark: "UPS, the UPS brand mark, and the Color Brown are trademarks of United Parcel Service of America, Inc. All Rights Reserved". Except as set forth in the preceding sentence, you have no right to use the UPS Mark without the prior written approval of UPS.
- You shall not use the UPS Mark in association with any third party trademarks in a manner that might suggest co-branding or otherwise create potential confusion as to source or sponsorship of the application, or ownership of the UPS Mark.
- The UPS Mark shall be used only as provided by UPS electronically or in hard copy form. The UPS Mark may not be altered in any manner, including proportions, colors, elements, etc., or animated, morphed or otherwise distorted in perspective or dimensional appearance.
- The UPS Mark may not be combined with any other symbols, including words, logos, icons, graphics, photos, slogans, numbers, or other design elements. A minimum amount of empty space must surround the UPS Mark separating it from any other object, such as type, photography, borders, edges, etc. The required area of empty space around the UPS Mark must be 1/3x, where x equals the height of the UPS Mark.

Copyright and Proprietary Notice

In your application and any POD Letters you prepare, you must include a prominent reproduction of UPS's copyright and proprietary notices in a form and format specified by UPS (See the [Copyright](#) section of this document).

Display of Information

The application must not display information concerning any other provider of shipping services or such other shipping services on any page, whether comprising one or more frames, displaying information your application receives from the UPS Developer Kit APIs. Your application must present all data within each field received through the UPS Developer Kit APIs without amendment, deletion, or modification of any type.

Notice

In all communications with UPS concerning this document, please refer to the document date located on the cover.

Copyright

© 2016 United Parcel Service of America, Inc. All Rights Reserved. Confidential and Proprietary

The use, disclosure, reproduction, modification, transfer, or transmittal of this work for any purpose in any form or by any means without the written permission of United Parcel Service is strictly prohibited.

Trademarks

Some of the UPS corporate applications use United States city, state, and postal code information obtained by United Parcel Service of America, Inc. under a non-exclusive license from the United States Postal Service.

Table of Contents

Chapter 1: Introduction	5
Release Features	6
January 2016	6
July 2015	6
Chapter 2: Understanding the Address Validation API	7
Business Processes and Rules	7
Address Validation Examples	8
Residential / Commercial Address Classification	9
Chapter 3: Customer Integration Environment (CIE)	10
Integration Testing	10
Production	10
Server Availability Check	10
Chapter 4: Address Validation API Elements and Details	11
Element Constraints	11
AddressValidationRequest Top Level Containers	12
AddressValidationRequest XPath Tables (XML)	12
Request	12
RegionalRequestIndicator	13
MaximumListSize	13
AddressKeyFormat	13
AddressValidationResponse Top Level Containers	15
AddressValidationResponse XPath Table	15
Response	15
ValidAddressIndicator	17
AmbiguousAddressIndicator	17
NoCandidatesIndicator	17
AddressClassification	17
AddressKeyFormat	18
Chapter 5: Address Validation Street Level XML Examples	20
AddressValidationRequest XML Example	20
AddressValidationResponse XML Example	20
Chapter 6: Address Validation Street Level FAQs	21
Appendices	23
Error Codes (XML)	24
Common Error Codes	24
Address Validation Street Level Error Codes (XML)	26
Address Validation Street Level API Supported Countries	27

Chapter 1: Introduction

In this document, you will find guidance and instructions for integrating the Address Validation Street Level XML API into your application, service, or system.

In this guide, you will find:

- New release features
- Address Validation Street Level XML API functionality and business rules
- XML API request and response schema
- FAQs for the Address Validation Street Level API
- Rating error response code descriptions
- Service code references

This guide applies to the following API:

- Address Validation Street Level XML API

This guide does not apply to the following APIs:

- Address Validation Street Level Web Service API
- Address Validation City, State, Zip API

Intended Audience

This guide is intended for developers who will be integrating the Address Validation Street Level XML API into their application, service, or system.

Required Knowledge & Skills

This guide assumes you have a basic understanding of the following:

- Concepts and instructions in the *Introduction to the UPS Developer Kit* guide.
- XML syntax and structure
- Software development

How to Use this Guide

- If this is your first time working with the UPS Developer Kit, begin with the *Introduction to the UPS Developer Kit* guide. This guide serves as the foundation for all API-specific developer guides.
- If you are experienced with the UPS Developer Kit but have not yet worked with the Address Validation Street Level API, continue with [Chapter 2: Understanding the Address Validation API](#). There you will find an overview of the Rating functionality and business rules.
- [Chapter 3: Customer Integration Environment \(CIE\)](#) contains information on UPS integration environments.
- [Chapter 4: Address Validation API Elements and Details](#) contains an overview of the top-level containers and elements that comprise the Rating request and response.
- Request and response examples are located in [Chapter 5: Address Validation Street Level XML Examples](#).
- Reference tables, services codes, and error codes are located in the [Appendices](#).

Release Features

January 2016

No change.

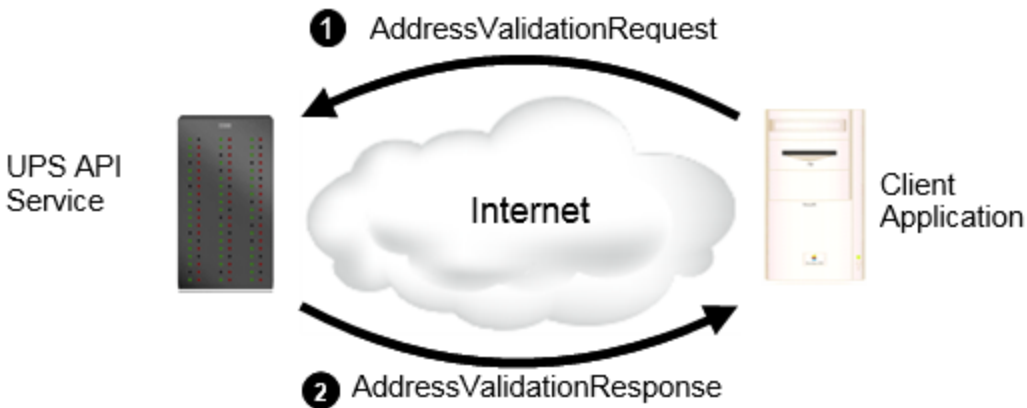
July 2015

No change.

Chapter 2: Understanding the Address Validation API

The Address Validation Street Level API can be used to check addresses against the United States Postal Service database of valid addresses in the U.S. and Puerto Rico. If an address is not valid according to this database, the API can optionally provide a list of valid addresses that correspond to the intended address. The API can also determine if an address is a commercial or residential address.

The figure below shows how client applications use address validation services. The process begins when the client sends an `AddressValidationRequest` to UPS API Services. UPS replies with an `AddressValidationResponse`.



Client applications can request either a general validation of a city, state, and zip code, or a validation of specific, street-level address. The `RegionalRequestIndicator` in the request determines which type of validation the application desires.

Client applications also indicate the maximum number of candidate addresses they wish to receive in the response. UPS returns candidate addresses only if the address that the client provides is not valid. Candidate addresses are valid addresses that might correspond to the requested address. If a client does not wish to receive candidate addresses, it can specify zero (0) for this value.

Business Processes and Rules


- UPS expects schema elements/tags as defined in the XPath and presented in the schema structure without spelling or structural deviations. Elements/tags that are not defined in the schema or do not conform to the schema structure will be ignored by UPS.
- Only users that plan to ship packages manifested, tendered, and delivered by UPS can use the API.
- Any customers/developers abusing or data mining the API will have their access revoked.
- To access API documentation the user must have or create a My UPS profile.
- To access testing and production sites the user must have an Access Key. To get an Access Key you must have a My UPS profile and an associated shipping account. For additional information, refer to the [UPS Developer Kit User Guide](#).

Address Validation

- The AVSL API does not perform Apt/Suite validation. In cases where there is an exact match (`ValidAddressIndicator` in Response) the API will return the Apt/Suite if it was provided in the Request.
- The AVSL API Response never returns the address provided in the Request. It is the responsibility of the customer to compare address provided in Request to address(es) returned in Response.

UPS Address Validation Notice

You must display the following notice, or such other language provided by UPS from time to time, in reasonable proximity to the Address Validation input and output information screens:

 NOTICE: UPS assumes no liability for the information provided by the address validation functionality. The address validation functionality does not support the identification or verification of occupants at an address.

Address Validation Examples

Validation of the Street Level Address Validation (SLAV) is not at the suite/apt# level, but if a perfect match contains a suite number or apt# the API will return a response that contains those elements.

Company or Name:	LAKESIDE PAIN CENTER
AddressLine:	6010 LAKESIDE COMMONS DR.
AddressLine:	STE B
City:	Macon
State:	GA
Zip:	31210
Country:	US

If the following address is sent in a request, the system will check if it exists. In this case, this address does not exist but the API will be able to return an address that maybe similar to the one that was provided as a candidate.

Company or Name:	LAKESIDE PAIN CENTER
AddressLine:	6010 LAKESIDE
AddressLine:	
City:	Macon
State:	
Zip:	312
Country:	US


The AVSL API does not perform Apt/Suite validation. In cases where there is an exact match (ValidAddressIndicator in Response) the API will return the Apt/Suite if it was provided in the Request.

Company or Name:	LAKESIDE PAIN CENTER
AddressLine:	6010 LAKESIDE COMMONS DR.
AddressLine:	STE X
City:	Macon
State:	GA
Zip:	31210
Country:	US

Residential / Commercial Address Classification

The Address Validation Street Level API can determine whether a given address is a residential or commercial address. Address classification uses the same request/response exchange as address validation. In fact, classification and validation can be combined in a single request.

For address classification, UPS determines whether an address is a residence or a commercial location. For address classification requests, it is important that the user include as much information as possible or available about the address, for example, a contact name or "attention to" value. Such information is important for accurate results, as many locations include both commercial and residential entities (such as a deli on the ground floor of an apartment building).

 **NOTE:** Consignee name is heavily weighted when determining resi / comm classification. Contrary to popular assumptions, an "address" in and of itself is not resi or comm, but rather the aggregate of all address elements, including and especially consignee name, are used to determine the classification.

Chapter 3: Customer Integration Environment (CIE)

The Customer Integration Environment allows customers to test their application prior to launch. This environment is intended for integration testing of customer applications with the UPS servers.

Once your application has been thoroughly tested, you should redirect the application to the UPS Production Environment.



No stress testing should ever be performed by customers against any UPS systems.

To access testing and production sites the user must have an Access Key. You can request an Access Key after establishing a UPS profile and associating your shipping account.

System Availability

The Customer Integration Environment is available 24 hours a day, 7 days a week. Note, the system is occasionally down for server maintenance.

Integration Testing

Test your Address Validation Street Level application with valid and invalid address elements.



NOTE: In the Customer Integration Environment, Address Validation Street Level will only produce results for addresses in New York (NY) and California (CA).

It is recommended that you use addresses that are familiar to you, for example, your home or business address. This will ensure that your application has the ability to process success and error responses correctly.

For integration testing, direct your test Address Validation Street Level XML to:

<https://wwwcie.ups.com/ups.app/xml/XAV>

Production

Once testing is complete, direct your Address Validation Street Level XML to the production URL:

<https://onlinetools.ups.com/ups.app/xml/XAV>

Server Availability Check

All of the UPS services work using HTTPS POST. If the server is available, it will reply with the service name, remote user, server port, server name, and servlet path.


To see this in action, type the following URL in your web browser: <https://wwwcie.ups.com/ups.app/xml/XAV>

Service Name:	XAV
Remote User:	null
Server Port:	443
Server Name:	wwwcie.ups.com
Servlet Path:	/XAV

Chapter 4: Address Validation API Elements and Details

Element Constraints

The Request and Response XPath tables contain the full hierarchy of the API elements.

 UPS expects schema elements/tags as defined in the XPath and presented in the schema structure i.e., no spelling or structural deviations. Elements/tags that are not defined in the schema or do not conform to the schema structure will be ignored by UPS.

Element constraints are defined as follows:

Required

- Yes - indicates the element must present in the request or response
- Yes* - indicates the element must be present in the request if the parent container is present in the request
- No - indicates the element is optional and may be used if it applies to the request
- Cond – indicates the element is required under certain conditions

Type

- Container - holds a group of related elements.
- String - consists of alphanumeric characters, spaces, and decimals

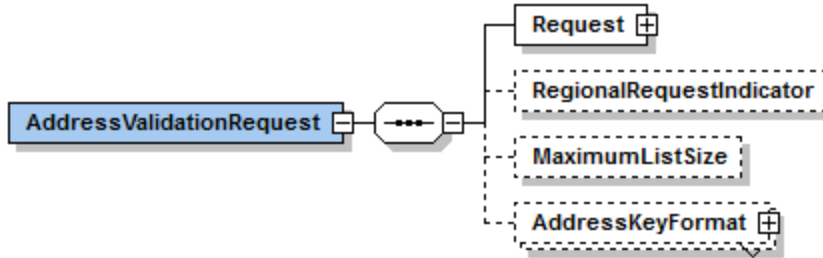
Length

- The maximum number of characters that can be present in a String. Length is not applicable (N/A) to Containers.

Max Allowed

- The maximum number of elements or containers that can be present in the parent container.

AddressValidationRequest Top Level Containers



AddressValidationRequest XPath Tables (XML)

Note: Not all containers/elements are reflected separately in the table of contents.

Name	Constraint	Description
/AddressValidationRequest		
AddressValidationRequest	Required: Yes Type: Container Max Allowed: 1 Length: N/A	N/A

Request

Name	Constraint	Description
/AddressValidationRequest/Request		
Request	Required: Yes Type: Container Max Allowed: 1 Length: N/A	N/A
/AddressValidationRequest/Request/TransactionReference		
TransactionReference	Required: No Type: Container Max Allowed: 1 Length: N/A	TransactionReference identifies transactions between client and server.
/AddressValidationRequest/Request/TransactionReference/CustomerContext		
CustomerContext	Required: No Type: String Max Allowed: 1 Length: 1...512	The client uses CustomerContext to synchronize request/response pairs. The client establishes CustomerContext, which can contain any information you want, as long as it is valid XML; it is echoed back by the server.
/AddressValidationRequest/Request/RequestAction		
RequestAction	Required: Yes Type: String Max Allowed: 1 Length: 3	Indicates the action to be taken by the XML service. Must be 'XAV'.

RegionalRequestIndicator

Name	Constraint	Description
/AddressValidationRequest/RegionalRequestIndicator		
RegionalRequestIndicator	Required: No Type: String Max Allowed: 1 Length:	<p>If this indicator is present then either the region element, or any combination of Political Division 1, Political Division 2, PostcodePrimaryLow and the PostcodeExtendedLow fields will be recognized for validation in addition to the urbanization element.</p> <p>If this tag is present, US and PR street level address validation will not occur.</p> <p>The default is to provide street level address validation.</p> <p>Not valid with the address classification request option.</p>

MaximumListSize

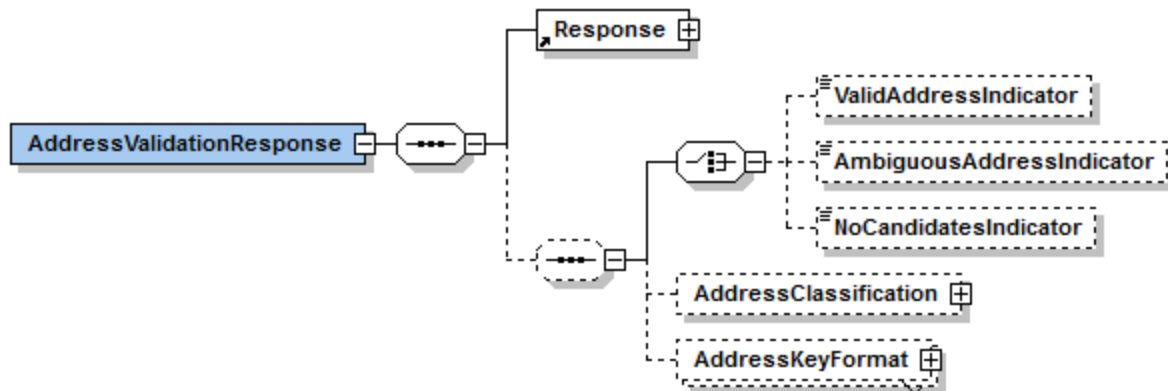
Name	Constraint	Description
/AddressValidationRequest/MaximumListSize		
MaximumListSize	Required: No Type: String Max Allowed: 1 Length: 1...3	<p>The maximum number of Candidates to return for this request.</p> <p>Valid values:</p> <ul style="list-style-type: none"> • 0-50 • Default value: 15.

AddressKeyFormat

Name	Constraint	Description
/AddressValidationRequest/AddressKeyFormat		
AddressKeyFormat	Required: Yes Type: Container Max Allowed: 1 Length:	<p>AddressKeyFormat container.</p> <p>The Key format is based on addressing standards jointly developed by the Postal Service and mailing industry.</p> <p>The information provided in the Address Key container will be returned in the same format.</p>
/AddressValidationRequest/AddressKeyFormat/ConsigneeName		
ConsigneeName	Required: No Type: String Max Allowed: 1 Length: 1...40	<p>Name of business, company or person.</p> <p>Ignored if user selects the RegionalRequestIndicator.</p>
/AddressValidationRequest/AddressKeyFormat/BuildingName		
BuildingName	Required: No Type: String Max Allowed: 1 Length: 1...40	<p>Name of building.</p> <p>Ignored if user selects the RegionalRequestIndicator.</p>
/AddressValidationRequest/AddressKeyFormat/AddressLine		
AddressLine	Required: No Type: String Max Allowed: 1 Length: 1...100	<p>Address line (street number, street name and street type) used for street level information.</p> <p>Applicable to US and PR only.</p> <p>Ignored if user selects the RegionalRequestIndicator.</p>

Name	Constraint	Description
/AddressValidationRequest/AddressKeyFormat/Region		
Region	Required: No Type: String Max Allowed: 1 Length: 1...100	Single entry, containing the following: <ul style="list-style-type: none"> • PoliticalDivision2 • PoliticalDivision1 • PostcodePrimaryLow • PostcodeExtendedLow
/AddressValidationRequest/AddressKeyFormat/PoliticalDivision2		
PoliticalDivision2	Required: No Type: String Max Allowed: 1 Length: 1...30	City or town name.
/AddressValidationRequest/AddressKeyFormat/PoliticalDivision1		
PoliticalDivision1	Required: No Type: String Max Allowed: 1 Length: 1...30	State or Province/Territory name.
/AddressValidationRequest/AddressKeyFormat/PostcodePrimaryLow		
PostcodePrimaryLow	Required: No Type: String Max Allowed: 1 Length: 1...10	Postal Code.
/AddressValidationRequest/AddressKeyFormat/PostcodeExtendedLow		
PostcodeExtendedLow	Required: No Type: String Max Allowed: 1 Length: 1...10	4-digit Postal Code extension. For U.S. use only.
/AddressValidationRequest/AddressKeyFormat/Urbanization		
Urbanization	Required: No Type: String Max Allowed: 1 Length: 1...30	Political Division 3. Only Valid for Puerto Rico.
/AddressValidationRequest/AddressKeyFormat/CountryCode		
CountryCode	Required: Yes Type: String Max Allowed: 1 Length: 2	Country Code. For a list of valid values, refer to the Address Validation Street Level API Supported Countries table in the Appendix.

AddressValidationResponse Top Level Containers



AddressValidationResponse XPath Table

Note: Not all containers/elements are reflected separately in the table of contents.

Name	Constraint	Description
/AddressValidationResponse		
AddressValidationResponse	Required: Yes Type: Container Max Allowed: 1 Length: N/A	N/A

Response

Name	Constraint	Description
/AddressValidationResponse/Response		
Response	Required: Yes Type: Container Max Allowed: 1 Length: N/A	Response Container.
/AddressValidationResponse/Response/TransactionReference/CustomerContext		
CustomerContext	Required: Type: String Max Allowed: 1 Length: 1...512	Echoes back the Customercontext from the Request
/AddressValidationResponse/Response/TransactionReference/XpciVersion		
XpciVersion	Required: Type: String Max Allowed: 1 Length: 1...15	Represents the version of the schema used by the XAV Tool
/AddressValidationResponse/Response/ResponseStatusCode		
ResponseStatusCode	Required: Yes Type: String Max Allowed: 1 Length: 1	Identifies the success or failure of the transaction. <ul style="list-style-type: none"> • 1 = Success • 0 = Failure

Name	Constraint	Description
/AddressValidationResponse/Response/ResponseStatusDescription		
ResponseStatusDescription	Required: Type: String Max Allowed: 1 Length: 1...35	Describes Response Status Code. Returns text of 'Success' or 'Failure'.
/AddressValidationResponse/Response/Error		
Error	Required: Yes Type: Container Max Allowed: 1 Length: N/A	Error Container. The error is described with a code and description.
/AddressValidationResponse/Response/Error/ErrorSeverity		
ErrorSeverity	Required: Yes Type: String Max Allowed: 1 Length: 1...10	Valid values: 'Hard' - The provided Request data will always fail 'Transient' - A UPS application is suffering an intermittent failure. Try the transaction again later.
/AddressValidationResponse/Response/Error/ErrorCode		
ErrorCode	Required: Yes Type: String Max Allowed: 1 Length: 1...2	Warning code returned by the system.
/AddressValidationResponse/Response/Error/ErrorMessage		
ErrorMessage	Required: Type: String Max Allowed: 1 Length: 1...150	
/AddressValidationResponse/Response/Error/MinimumRetrySeconds		
MinimumRetrySeconds	Required: Type: String Max Allowed: 1 Length: 1...15	How long to wait before re-transmitting a Transaction that failed with a 'Transitive' error
/AddressValidationResponse/Response/Error/ErrorMessage		
ErrorMessage	Required: Yes Type: Container Max Allowed: 1 Length: 1...150	XPATH of the element causing the 'Hard' error
/AddressValidationResponse/Response/Error/ErrorMessage/ErrorMessageElementName		
ErrorMessageElementName	Required: Type: String Max Allowed: 1 Length: 1...30	Element whose value causes the 'Hard' error
/AddressValidationResponse/Response/Error/ErrorMessage/ErrorMessageAttributeName		
ErrorMessageAttributeName	Required: Type: String Max Allowed: 1 Length:	

Name	Constraint	Description
/AddressValidationResponse/Response/Error/ErrorDigest		
ErrorDigest	Required: Type: String Max Allowed: 1 Length: Unbounded	Details of the error

ValidAddressIndicator

Name	Constraint	Description
/AddressValidationResponse/ValidAddressIndicator		
ValidAddressIndicator	Required: Cond Empty Max Allowed: 1 Length: N/A	Indicates query found an exact match.

AmbiguousAddressIndicator

Name	Constraint	Description
/AddressValidationResponse/AmbiguousAddressIndicator		
AmbiguousAddressIndicator	Required: Cond Type: String Max Allowed: 1 Length: N/A	Indicates query could not find exact match. Candidate list follows.

NoCandidatesIndicator

Name	Constraint	Description
/AddressValidationResponse/NoCandidatesIndicator		
NoCandidatesIndicator	Required: Cond Type: String Max Allowed: 1 Length: N/A	No Candidate found.

AddressClassification

Name	Constraint	Description
/AddressValidationResponse/AddressClassification		
AddressClassification	Required: No Type: Container Max Allowed: 1 Length: N/A	Container returning the classification of the input address, if requested
/AddressValidationResponse/AddressClassification/Code		
Code	Required: Yes* Type: String Max Allowed: 1 Length: 1	Contains the classification code of the input address. <ul style="list-style-type: none"> • 0 = Unknown • 1 = Commercial • 2 = Residential

Name	Constraint	Description
/AddressValidationResponse/AddressClassification/Description		
Description	Required: Yes* Type: String Max Allowed: 1 Length: 1...15	Contains the text description of the address classification code: <ul style="list-style-type: none"> • Unknown • Commercial • Residential

AddressKeyFormat

Name	Constraint	Description
/AddressValidationResponse/AddressKeyFormat		
AddressKeyFormat	Required: Cond Type: Container Max Allowed: 1 Length: N/A	AddressKeyFormat Container The Key format is based on addressing standards jointly developed by the Postal Service and mailing industry. The information provided in the Address Key container will be returned in the same format.
/AddressValidationResponse/AddressKeyFormat/AddressClassification		
AddressClassification	Required: No Type: Container Max Allowed: 1 Length: N/A	Container returning the classification of the address, if requested
/AddressValidationResponse/AddressKeyFormat/AddressClassification/Code		
Code	Required: Yes* Type: String Max Allowed: 1 Length: 1	Contains the classification code of the address. <ul style="list-style-type: none"> • 0 = Unknown • 1 = Commercial • 2 = Residential
/AddressValidationResponse/AddressKeyFormat/AddressClassification/Description		
Description	Required: Yes* Type: String Max Allowed: 1 Length: 1...15	Contains the text description of the address classification code. <ul style="list-style-type: none"> • Unknown • Commercial • Residential
/AddressValidationResponse/AddressKeyFormat/ConsigneeName		
ConsigneeName	Required: No Type: String Max Allowed: 1 Length: 1...40	Name of business, company or person. Not returned if user selects the RegionalRequestIndicator.
/AddressValidationResponse/AddressKeyFormat/BuildingName		
BuildingName	Required: Cond Type: String Max Allowed: 1 Length: 1...40	Name of building. Not returned if user selects the RegionalRequestIndicator.

Name	Constraint	Description
/AddressValidationResponse/AddressKeyFormat/AddressLine		
AddressLine	Required: Cond Type: String Max Allowed: 1 Length: 1...100	Address line (street number, street name and street type, and political division 1, political division 2 and postal code) used for street level information. Applicable to US and PR only. Not returned if user selects the RegionalRequestIndicator.
/AddressValidationResponse/AddressKeyFormat/Region		
Region	Required: No Type: String Max Allowed: 1 Length: 1...30	Single entry containing in this order Political Division 2, Political Division 1 and Post Code Primary Low and/or PostcodeExtendedLow.
/AddressValidationResponse/AddressKeyFormat/PoliticalDivision2		
PoliticalDivision2	Required: Cond Type: String Max Allowed: 1 Length: 1...30	City or Town.
/AddressValidationResponse/AddressKeyFormat/PoliticalDivision1		
PoliticalDivision1	Required: Cond Type: String Max Allowed: 1 Length: 1...30	State/Province. Returned if the location is within a State/Province/Territory. For International: returned if user enters valid Country Code, and City/postal code and it has a match. For Domestic addresses, the value must be a valid 2-character value (per US Mail standards). For International the full State or Province name will be returned.
/AddressValidationResponse/AddressKeyFormat/PostcodePrimaryLow		
PostcodePrimaryLow	Required: Cond Type: String Max Allowed: 1 Length: 1...10	Low-end Postal Code. Returned for countries with Postal Codes. May be alphanumeric
/AddressValidationResponse/AddressKeyFormat/PostcodeExtendedLow		
PostcodeExtendedLow	Required: No Type: String Max Allowed: 1 Length: 1...10	Low-end extended postal code in a range. Example in quotes: Postal Code 30076-'1234'. Only returned in candidate list. May be alphanumeric
/AddressValidationResponse/AddressKeyFormat/Urbanization		
Urbanization	Required: No Type: String Max Allowed: 1 Length: 1...10	Puerto Rico Political Division 3. Only Valid for Puerto Rico.
/AddressValidationResponse/AddressKeyFormat/CountryCode		
CountryCode	Required: Yes Type: String Max Allowed: 1 Length: 1...2	A country code. Required to be returned.

Chapter 5: Address Validation Street Level XML Examples

AddressValidationRequest XML Example

```
<?xml version="1.0" ?>
  <AccessRequest xml:lang='en-US'>
    <AccessLicenseNumber>YOURACCESSLICENSENUMBER</AccessLicenseNumber>
    <UserId>YOURUSERID</UserId>
    <Password>YOURPASSWORD</Password>
  </AccessRequest>
<?xml version="1.0" ?>
<AddressValidationRequest xml:lang='en-US'>
  <Request>
    <TransactionReference>
      <CustomerContext>Your Customer Context</CustomerContext>
      <XpciVersion>1.0</XpciVersion>
    </TransactionReference>
    <RequestAction>XAV</RequestAction>
    <RequestOption>1</RequestOption>
  </Request>
  <AddressKeyFormat>
    <AddressLine>12380 MORRIS RD</AddressLine>
    <Region>ALPHARETTA GA 30005-4177</Region>
    <PoliticalDivision2>ALPHARETTA</PoliticalDivision2>
    <PoliticalDivision1>GA</PoliticalDivision1>
    <PostcodePrimaryLow>30005</PostcodePrimaryLow>
    <PostcodeExtendedLow>4177</PostcodeExtendedLow>
    <CountryCode>US</CountryCode>
  </AddressKeyFormat>
</AddressValidationRequest>
```

AddressValidationResponse XML Example

```
<?xml version="1.0" ?>
<AddressValidationResponse>
  <Response>
    <TransactionReference>
      <CustomerContext>Your Customer Context</CustomerContext>
      <XpciVersion>1.0</XpciVersion>
    </TransactionReference>
    <ResponseStatusCode>1</ResponseStatusCode>
    <ResponseStatusDescription>Success</ResponseStatusDescription>
  </Response>
  <ValidAddressIndicator/>
  <AddressKeyFormat>
    <AddressLine>12380 MORRIS RD</AddressLine>
    <Region>ALPHARETTA GA 30005-4177</Region>
    <PoliticalDivision2>ALPHARETTA</PoliticalDivision2>
    <PoliticalDivision1>GA</PoliticalDivision1>
    <PostcodePrimaryLow>30005</PostcodePrimaryLow>
    <PostcodeExtendedLow>4177</PostcodeExtendedLow>
    <CountryCode>US</CountryCode>
  </AddressKeyFormat>
</AddressValidationResponse>
```

Chapter 6: Address Validation Street Level FAQs

Category	Question	Answer
General	What countries' addresses can be validated by the Address Validation - Street Level API?	The Address Validation - Street Level API allows the validation of street level address in the US and Puerto Rico only. Note: AVSL supports classification for US & Canadian addresses only.
General	Does the SLAV API classify addresses?	The Address Validation - Street Level API classifies addresses in both US and Canada. The API classifies addresses as Residential or Commercial in keeping with UPS standards.
General	Why do we get back a candidate list of addresses even when the response has a Valid Address Indicator?	The API returns a candidate list even when there is a Valid Address Indicator because the parameters entered have more than one valid match. If the address that was entered as part of the request is returned as part of the candidate list then the address is valid and should be used.
General	How is street level AV completed? Does it use the USPS to validate an address?	Currently the Address Validation Street Level API's database is updated at monthly intervals with new address information from USPS. Generally, the database update will occur around the 15th of the month. The actual date changes from month to month dependent upon the amount of testing a given data set might require issues that are found, and other factors that can contribute to the swiftness of data validation. The database updates should synchronize the information with the USPS.
General	How much time will it take to program/implement the Address Validation - Street Level API?	The programming/implementation of the Address Validation - Street Level APIs may vary and is strictly dependent on the skill level of the developer. An implementation of Address Validation Street Level may take as little as a week for a very skilled developer to as long as months for a less skilled developer.
Address Validation frequency of updates	How frequently are the databases updated for Address Validation Street Level?	Address Validation Street Level API 1. Classification = weekly 2. Validation = monthly
Batch Upload	Does either the Address Validation or Address Validation Street Level APIs offer the ability to batch upload?	No.
CASS Certified	Is the Address Validation Street Level API CASS certified or the data we receive CASS certified?	UPS does not provide any API or application with customer-facing CASS-certified address validation. CASS certification is required for discounts on USPS services, but not required for any UPS services or discounts. UPS address validation is not CASS-certified. It is also free, as we are not competing with vendors that provide CASS-certified Address

Category	Question	Answer
		<p>Validation software. The Address Validation API is provided to allow customers who do not already use CASS-certified Address Validation software, our Address Validation Street Level API can be used to help clean-up addresses for UPS shipments.</p> <p>UPS delivers to addresses that are not in the USPS database (some examples are addresses that are warehouses that do not accept mail and areas where the USPS only provides PO Box delivery) - so an invalid address may still be deliverable.</p> <p>In addition, a valid address may be the wrong address, and still require an address correction. NOTE: The source of our data for address validation is the USPS that we subscribe to and refresh monthly</p>
Resi/Comm Indicator	Which APIs provide address classification?	The Address Validation Street Level API provides address classification only for US and CA
Resi/Comm Database	What process does UPS use to apply residential/commercial designations?	UPS business logic is used to determine the results
Suite/Apt #	Does Address Validation Street Level API provide a candidate list for addresses that have suite/apt information?	The API does not return candidate lists for suite or apartment number ranges
Valid/Invalid Address	How do I know whether the address I submitted is correct?	<p>If the address is entered incorrectly, the API will return an AmbiguousAddressIndicator and a list of candidate addresses.</p> <p>If the address is entered correctly, the API will return a ValidAddressIndicator and the correct address.</p>

Appendices

Error Codes (XML)

When the UPS system is unable to respond to a request, be it from a malformed request, an illegal or invalid value, or other issues, the API generates an error response. For examples and additional information on error responses, refer to *Chapter 3: Error Responses*, in the *Introduction to the UPS Developer Kit* guide.

To discover errors, check the `ResponseStatusCode` element. A “1” normally indicates a successful response, whereas a “0” indicates a Transient or Hard error. The `PrimaryErrorCode` element contains the error code and description.

- Successful responses may or may not include warnings.
 - Without warnings - Indicates the request has been processed as anticipated.
 - With warnings - Indicates the request has been processed with potentially unanticipated results. The warning contains information in the response that should be passed to the end user.
- The severity of an error may be transient or hard.
 - Transient error - Indicates an error that is temporary in nature. Such errors may be caused by response timeouts due to high server loads or scheduled maintenance in progress. The request may be issued successfully at a later time.
 - Hard error - Indicates the request has a problem that the system is not able to resolve. These errors are critical and prevent requests from processing.



Do not resubmit requests with Hard Errors until the problems causing the error have been resolved.

For examples and additional information on error responses, refer to *Chapter 3: Error Responses*, in the *Introduction to the UPS Developer Kit* guide.

Common Error Codes

The following error codes can apply to all Web Services and APIs.

Code	Severity	Description
10001	Hard	The XML document is not well formed
10002	Hard	The XML document is well formed but the document is not valid
10003	Hard	The XML document is either empty or null
10006	Hard	Although the document is well formed and valid, the element content contains values which do not conform to the rules and constraints contained in this specification
10013	Hard	The message is too large to be processed by the Application
20001	Transient	General process failure
20002	Hard	The specified service name, {0}, and version number, {1}, combination is invalid
20003	Hard	Please check the server environment for the proper J2EE ws apis
20006	Hard	Invalid request action
20007	Hard	Missing Required field, {0}
20008	Hard	The field, {0}, contains invalid data, {1}
20012	Hard	The Client Information exceeds its Maximum Limit of {0}
250000	Hard	No XML declaration in the XML document
250001	Hard	Invalid Access License for the tool. Please re-license.
250002	Hard	Invalid UserId/Password
250003	Hard	Invalid Access License number
250004	Hard	Incorrect UserId or Password
250005	Hard	No Access and Authentication Credentials provided

Error Codes (XML)

Code	Severity	Description
250006	Hard	The maximum number of user access attempts was exceeded
250007	Hard	The UserId is currently locked out; please try again in 24 hours.
250009	Hard	License Number not found in the UPS database
250019	Hard	Invalid Field Value
250050	Transient	License system not available

Address Validation Street Level Error Codes (XML)

Error Code	Severity	Description	Condition
260050	Hard	Invalid Request Action	The request action is invalid.
264001	Transient	AV Service is not available	Adapter parsing error, business process calling error, backend service is unavailable, etc.
264002	Hard	Country Code is invalid or missing.	The country code is not US or PR.
264003	Hard	The Maximum allowable Candidate List size has been exceeded within the User Request.	The maximum candidate list size requested from the user has been exceeded.
264004	Hard	The maximum validation query time has been exceeded due to poor address data.	Request has timed out. Usually due to insufficient or poor address data from client.
264005	Hard	Address classification is not valid for a regional request.	The customer submits a request for address classification with a regional address format.
264006	Hard	Invalid candidate list size.	The maximum candidate list size given by the customer is not a numeric value between 0 and 2147483647.
264007	Hard	Address classification is not allowed for the country requested.	The request contains address classification with a country code that is not supported for classification.
264008	Hard	Country code and address format combination is not allowed.	Country code and address format combination is not allowed.
264027	Hard	Additional address fields are needed to perform the requested operation.	The Country Code is valid but the other fields are blank. Additional fields need to be provided in order to validate or classify the address.
264030	Hard	The state is not supported in the Customer Integration Environment.	

Address Validation Street Level API Supported Countries

UPS country code abbreviations generally follow the recommendations of the International Standards Organization (ISO), which publishes a list of country abbreviations in ISO Standard 3166.



NOTE: Not all UPS services are available in every country. Refer to the [UPS Rate and Service Guide](#) at [UPS.com](#) for more information on UPS services.

Country Name	Country Code	Residential / Commercial Classification	Street Level Validation
United States	US	X	X
Puerto Rico	PR		X
Canada	CA	X	