Informatica®

Informatica® Address Verification
3.0

Developer Guide

Informatica Address Verification Developer Guide
3.0
May 2016

DISCLAIMER: Informatica LLC provides this documentation "as is" without warranty of any kind, either express or implied, including, but not limited to, the implied warranties of noninfringement, merchantability, or use for a particular purpose. Informatica LLC does not warrant that this software or documentation is error free. The information provided in this software or documentation may include technical inaccuracies or typographical errors. The information in this software and documentation is subject to change at any time without notice.

NOTICES

This Informatica product (the "Software") includes certain drivers (the "DataDirect Drivers") from DataDirect Technologies, an operating company of Progress Software Corporation ("DataDirect") which are subject to the following terms and conditions:

1. THE DATADIRECT DRIVERS ARE PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT.
2. IN NO EVENT WILL DATADIRECT OR ITS THIRD PARTY SUPPLIERS BE LIABLE TO THE END-USER CUSTOMER FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, CONSEQUENTIAL OR OTHER DAMAGES ARISING OUT OF THE USE OF THE ODBC DRIVERS, WHETHER OR NOT INFORMED OF THE POSSIBILITIES OF DAMAGES IN ADVANCE. THESE LIMITATIONS APPLY TO ALL CAUSES OF ACTION, INCLUDING, WITHOUT LIMITATION, BREACH OF CONTRACT, BREACH OF WARRANTY, NEGLIGENCE, STRICT LIABILITY, MISREPRESENTATION AND OTHER TORTS.

Publication Date: 2018-08-03

Table of Contents

Preface

Welcome to the Informatica Address Verification Developer Guide. This guide provides you information about Address Verification features and configuration settings. This guide also contains some examples that you can use to verify your settings.

Informatica Resources

Informatica Network

Informatica Network hosts Informatica Global Customer Support, the Informatica Knowledge Base, and other product resources. To access Informatica Network, visit https://network.informatica.com.

As a member, you can:

- Access all of your Informatica resources in one place.
- Search the Knowledge Base for product resources, including documentation, FAQs, and best practices.
- View product availability information.
- Review your support cases.
- Find your local Informatica User Group Network and collaborate with your peers.

As a member, you can:

- Access all of your Informatica resources in one place.
- Search the Knowledge Base for product resources, including documentation, FAQs, and best practices.
- View product availability information.
- Find your local Informatica User Group Network and collaborate with your peers.

Informatica Knowledge Base

Use the Informatica Knowledge Base to search Informatica Network for product resources such as documentation, how-to articles, best practices, and PAMs.

To access the Knowledge Base, visit https://kb.informatica.com. If you have questions, comments, or ideas about the Knowledge Base, contact the Informatica Knowledge Base team at KB_Feedback@informatica.com.

Informatica Documentation

To get the latest documentation for your product, browse the Informatica Knowledge Base at https://kb.informatica.com/_layouts/ProductDocumentation/Page/ProductDocumentSearch.aspx.

If you have questions, comments, or ideas about this documentation, contact the Informatica Documentation team through email at infa_documentation@informatica.com.

Informatica Product Availability Matrixes

Product Availability Matrixes (PAMs) indicate the versions of operating systems, databases, and other types of data sources and targets that a product release supports. If you are an Informatica Network member, you can access PAMs at https://network.informatica.com/community/informatica-network/product-availability-matrices.

Informatica Velocity

Informatica Velocity is a collection of tips and best practices developed by Informatica Professional Services. Developed from the real-world experience of hundreds of data management projects, Informatica Velocity represents the collective knowledge of our consultants who have worked with organizations from around the world to plan, develop, deploy, and maintain successful data management solutions.

If you are an Informatica Network member, you can access Informatica Velocity resources at http://velocity.informatica.com.

If you have questions, comments, or ideas about Informatica Velocity, contact Informatica Professional Services at ips@informatica.com.

Informatica Marketplace

The Informatica Marketplace is a forum where you can find solutions that augment, extend, or enhance your Informatica implementations. By leveraging any of the hundreds of solutions from Informatica developers and partners, you can improve your productivity and speed up time to implementation on your projects. You can access Informatica Marketplace at https://marketplace.informatica.com.

Informatica Global Customer Support

You can contact a Global Support Center by telephone or through Online Support on Informatica Network.

To find your local Informatica Global Customer Support telephone number, visit the Informatica website at the following link:
http://www.informatica.com/us/services-and-training/support-services/global-support-centers.

If you are an Informatica Network member, you can use Online Support at http://network.informatica.com.

CHAPTER 1

Introduction to Informatica Address Verification

This chapter includes the following topics:

Informatica Address Verification Overview

Informatica Address Verification is an online address validation solution that you can use to verify and validate international postal addresses in real time. Address Verification is a SOAP-based solution that runs on top of the latest Address Verification software and the most up-to-date reference address databases.

You can integrate Address Verification with CRM systems, e-commerce sites, or other cloud- or web-based solutions to verify and validate addresses. You can access Address Verification over Secure Sockets Layer (SSL) connections. To access Address Verification, you need an Address Verification Data Quality Center (DQC) user account with sufficient valid transactions.

A new version number in the Address Verification interface indicates a new Address Verification interface. For example, the current Informatica Address Verification URL contains the version number V3. To access the latest Address Verification features, ensure that your SOAP calls point to the latest Address Verification URL.

If you do not want to upgrade an Address Verification implementation, you can continue to access the previous versions until the time Address Verification supports the version. For more details about versions that Address Verification supports, contact an Address Verification sales representative.

Informatica Address Verification DQC Account

To validate addresses over Informatica Address Verification, you need a valid Informatica Address Verification DQC account with sufficient number of transactions.

To access the Informatica Address Verification DQC portal, you need an Informatica Passport account. If you do not have an Informatica Passport account, you can create one from https://dqc.addressdoctor.com/sso/account/register. After you create an Informatica Passport account, you can assign that account to one or more Informatica Address Verification DQC accounts.

If you do not have an Informatica Address Verification DQC account, you can use your Informatica Passport credentials to access the DQC portal, and create an account. You can access the Informatica Address Verification DQC account at https://dqc.addressdoctor.com/. Alternatively, you can access the Informatica Address Verification DQC portal from the Informatica Passport single sign-on dashboard at https://infapassport.okta.com/. When you create the first Informatica Address Verification DQC account for your organization, Informatica Address Verification assigns multiple free transactions to the new account.

For more information about Informatica Address Verification DQC accounts, see the *Informatica AddressDoctor Data Quality Center User Guide*.

Informatica Address Verification Transactions

To validate addresses over Informatica Address Verification, you must have sufficient valid transactions in your Informatica Address Verification DQC account. To authenticate an Informatica Address Verification address validation request, you must include the Informatica Address Verification DQC account ID and password in the request.

You can buy transactions online from the Informatica Address Verification DQC portal. You can choose the type of transactions that you need and buy sufficient number of transactions based on your requirements.

For example, you can use batch transactions to validate addresses in the batch and the certified modes. Similarly, you can use interactive transactions to validate addresses in the interactive and fast completion modes.

Key Capabilities of Informatica Address Verification

The following features significantly improve the Informatica Address Verification user experience:

One Interface

You can access all the process modes of Address Verification from a single interface. You can enter up to 1000 addresses in the batch mode. In the interactive and fast completion modes, Address Verification Cloud returns up to 100 suggestions.

Extensibility

Address Verification has a flexible interface that helps you access new Address Verification features without changing the WSDL. For example, the Version 4 Web Service defined enumerators for all parameters. Adding more options to the pervious web service meant modifications to the interface. Address Verification defines the Parameter names, but not the values, which are plain String values. In this way, you can add new values to existing parameters without changing the interface.

The current interface is the third iteration of the interface. The login URL contains identifies the interface version.

The following URL contains the version identifier `v3`:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:v1="http://validator5.AddressDoctor.com/Webservice5/v3">
```

Improved Performance

Address Verification can use the multithreading capabilities of Address Verification. Additionally, Address Verification also benefits from improved processing speed.

Default Method

The default call from Address Verification goes to the web method, `process`. Address Verification provides you with objects for parameters and addresses just as the Web Service 4 did. The objects have a new structure to support the capabilities of the Address Verification version 5 library.

Retrieving Detailed Information for Address Elements

You can retrieve the information in each `Address` element in the format that Informatica Address Verification (On-Premises) defines. To retrieve the information in the on-premises format, set the `AddressDetailed` element under `OutputOptions` to ON. The `AddressDetailed` element instructs Address Verification to return discrete elements in the response for each item within the `Address` element.

Informatica adds the `AddressDetailed` element in version 3 of the interface.

In the following example, the `AddressDetailed` element specifies that Address Verification returns all address information in discrete elements where possible:

```
<v3:OutputOptions>
     <v3:AddressDetailed>ON</v2:AddressDetailed>
</v3:OutputOptions>
```

The default value for `AddressDetailed` is OFF. When you set `AddressDetailed` to ON, Informatica Address Verification returns detailed address information if the information is available in the database.

The following example shows an address string within the `<Street>` element and the corresponding information that Address Verification can return within the `<AddressDetailed>` element:

```
<Street>
<String>3500 MAIN STATION DR SW</String>
</Street>
<AddressDetailed>
     <Street>
          <StreetDetailed>
               <Item>1</Item>
               <COMPLETE>MAIN STATION DR SW</COMPLETE>
               <COMPLETE_WITH_NUMBER>3500 MAIN STATION DR SW</COMPLETE_WITH_NUMBER>
               <NAME>MAIN STATION</NAME>
               <POST_DESCRIPTOR>DR</POST_DESCRIPTOR>
               <POST_DIRECTIONAL>SW</POST_DIRECTIONAL>
          </StreetDetailed>
     </Street>
</AddressDetailed>
```

Note: Before you use the `<AddressDetailed>` element, you must understand the operations of Informatica Address Verification (On-Premises). For more information about the `<AddressDetailed>` element, see the *Informatica Address Verification (On-Premises) Developer Guide*.

Product Features and Functionality

The Informatica Address Verification interface provides access to the functionality of the Informatica Address Verification software library. Each version of the interface provides access to the current software library. The interfaces differ in the elements and parameters that they contain.

The following sections describe representative features from different versions of the software library.

Note: Each description represents the feature behavior at the time that Informatica added the feature to the software library. A recent update might supersede the behavior of an older feature.

Single-Line Address Validation in China

You can configure Informatica Address Verification to return valid suggestions for a China address that you enter on a single line in fast completion mode. To enter an address on a single line, use the `AddressComplete` element. Enter the address in the Hanzi script.

When you enter a partial address, Address Verification returns one or more address suggestions for the address that you enter. When you enter a complete valid address, Address Verification returns the valid version of the address from the reference database.

Informatica added the feature in Address Verification (On-Premises) 5.9.0.

Multilanguage Support for Street Descriptors and Street Directionals in China

You can configure Informatica Address Verification to return the street descriptor and street directional information in a valid China address in a transliterated Latin script (Pinyin) or in English. Address Verification returns the other elements in the address in Pinyin.

Informatica added the feature in Address Verification (On-Premises) 5.9.0.

Support for Global Preferred Descriptors for Ireland Addresses

You can configure Informatica Address Verification to return the short or long forms of the following elements in the English language:

- Street descriptors
- Directional values

Use the GlobalPreferredDescriptor attribute to configure Address Verification to return the short or long form or to preserve the form that the input address uses. For example, you can configure Address Verification to abbreviate STREET to ST and NORTH to N.

You can also configure Address Verification to return the element values in the default format of the reference data.

Informatica Address Verification writes all street information to the street name field in a fielded Irish-language address.

Informatica added the feature in Address Verification (On-Premises) 5.9.0.

Rooftop Geocoding for Ireland Addresses

You can configure Informatica Address Verification to return high-precision rooftop geocoordinates for an address in Ireland.

To receive rooftop geocoordinates for Ireland addresses, set the following parameters:

- Set the `Type` attribute of the `Enrichment` parameter to GEOCODING.
- Set the `GeoCodingType` validation parameter to ROOFTOP.

Additionally, you must have GEOCODING transactions in your account.

Address Verification returns a geocoding status value of EGCB to indicate that the output address contains rooftop geocoordinates. If rooftop geocoordinates are not available, Address Verification does not return any geocoordinates for the address. When geocoordinates are not available, Address Verification returns the geocoding status code EGC0.

Informatica added the feature in Address Verification (On-Premises) 5.9.0.

Multilanguage Support for Street, Locality, and County Information in Ireland

You can configure Informatica Address Verification to read and write the street, locality, and county information for an Ireland address in the Irish language.

An Post, the Irish postal service, maintains the Irish-language information in addition to the English-language addresses. You can include Irish-language street, locality, and county information in an input address and retrieve the valid English-language version of the address. You can enter an English-language address to Informatica Address Verification and retrieve an address that includes the street, locality, and county information in the Irish language. Informatica Address Verification returns all other information in English.

Use the PreferredLanguage attribute to specify the output language.

Informatica added the feature in Address Verification (On-Premises) 5.9.0.

ISTAT Code Enrichment for Italy Addresses

You can configure Informatica Address Verification to add the ISTAT code to a valid Italy address. The ISTAT code contains characters that identify the province, municipality, and region to which the address belongs. The Italian National Institute of Statistics (ISTAT) maintains the ISTAT codes.

Address Verification returns the following status values to indicate the level of ISTAT code enrichment for each address:

- EIT0. ISTAT code data is not available for the input address.
- EIT1. The output address contains ISTAT code data.

To return the ISTAT codes for Italy addresses, set the `Type` attribute of the `Enrichment` parameter to SUPPLEMENTARY_IT.

Informatica added the feature in Address Verification (On-Premises) 5.9.0.

Geocoding Enrichment for Japan Addresses

You can configure Informatica Address Verification to return standard (interpolated) geocoordinates for addresses in Japan.

Address Verification can return geocoordinates at multiple levels of accuracy. When a valid address contains information to the Ban level, Address Verification returns house number-level geocoordinates. When a valid address contains information to the Chome level, Address Verification returns street-level geocoordinates. If an address does not contain Ban or Chome information, Address Verification returns locality-level geocoordinates.

Address Verification returns the following geocoding status values to indicate the geocode level for an address:

- EGC8. The geocoordinates represent the Ban.
- EGC7. The geocoordinates represent the Chome.
- EGC6. The geocoordinates represent the locality.

Informatica added the feature in Address Verification (On-Premises) 5.9.0.

Single-Line Validation of Japan Addresses

You can configure Informatica Address Verification to return valid suggestions for a Japan address that you enter on a single line in Fast Completion mode. You can retrieve suggestions for an address that you enter in the Kanji script or the Kana script. To enter an address on a single line, use the `AddressComplete` element.

When you enter a partial address, Address Verification returns one or more address suggestions for the address that you enter. When you enter a complete valid address, Address Verification returns the valid version of the address from the reference database.

Informatica added the feature in Address Verification (On-Premises) 5.9.0.

Lot Number Parsing in South Korea Addresses

Address Verification parses lot number and lot sub-number identifiers from a Hangul address to the fields that include the lot number and lot sub-number data.

For example, the following address element includes the 번지 or *bunji* lot number identifier:

```
<AddressComplete>서울시  영등포구  여의도동  20번지</AddressComplete>
```

Address Verification parses 20번지 to the house number field.

Informatica added the feature in Address Verification (On-Premises) 5.9.0.

Support for Revised Romanization Transliteration in South Korea Addresses

Address Verification can use the Revised Romanization system to transliterate an address between Hangul and Latin character sets. To specify a character set for output addresses from South Korea, use the PreferredScript attribute.

Informatica added the feature in Address Verification (On-Premises) 5.9.0.

Updates to Post Code Verification for South Korea

Address Verification adds a five-digit post code to a fully valid input address that does not include a post code. The five-digit post code represents the current post code format in use in South Korea.

Address Verification can add the five-digit post code to a fully valid lot-based address and a fully valid street-based address. To verify addresses in the older, lot-based format, set the MatchingExtendedArchive attribute to `true` or `1`.

Informatica added the feature in Address Verification (On-Premises) 5.9.0.

INE Code Enrichment for Spain Addresses

You can configure Informatica Address Verification to add the INE code to a valid Spain address. The INE code contains characters that identify the province, municipality, and street in the address. The National Institute of Statistics (INE) in Spain maintains the INE codes.

Address Verification can return the following status values to indicate the level of INE code enrichment for each address:

- EES0. INE code data is not available for the input address.
- EES1. The output address contains INE code data.

To return the INE codes for Spain addresses, set the `Type` attribute of the `Enrichment` parameter to SUPPLEMENTARY_ES.

Informatica added the feature in Address Verification (On-Premises) 5.9.0.

Support for Global Preferred Descriptors in United States Addresses

You can return the short form or the long form of the following elements in a United States address:

- Street descriptors
- Directional values
- Sub-building descriptors

For example, you can configure Address Verification to abbreviate STREET to ST and FLOOR to FL. Use the GlobalPreferredDescriptor attribute to configure Address Verification to return the short or long form or to preserve the form that the input address uses.

You can also configure Address Verification to return the element values in the default format of the reference data.

Informatica added the feature in Address Verification (On-Premises) 5.9.0.

Improved Parsing of Non-Standard United States Data

Address Verification parses non-standard mailbox data from the first line of the address to sub-building elements.

For example, the non-standard data might identify a courtroom or a college campus mailbox.

Informatica added the feature in Address Verification (On-Premises) 5.9.0.

Support for Eircodes in Ireland Addresses

Address Verification supports the eircode system in Ireland. An eircode is a seven-character code that uniquely identifies an Ireland address. The eircode system covers all residences, public buildings, and business premises and includes apartment addresses and addresses in rural townlands.

Informatica added the feature in Address Verification (On-Premises) 5.8.1.

Support for Hexaligne 3 Data in France Address Certification

Informatica Address Verification can use Hexaligne 3 data to verify France addresses in certified mode.

The Hexaligne 3 file contains additional data about the building that contains the address mailbox.

Informatica added the feature in Address Verification (On-Premises) 5.8.1.

Enhancements to Address Code LookUp Mode

Informatica Address Verification returns all available enrichment data with an output address from any country that supports a unique identifier value as a lookup key.

To retrieve address data in the address code lookup mode, set the `ProcessMode` parameter to `ADDRESSCODELOOKUP`.

Address Verification supports the `ADDRESSCODELOOKUP` process mode for the following countries and codes:

- Germany: DEU_AGS, DEU_LOCALITY_ID, DEU_STREET_ID
- South Africa.: ZAF_NADID
- Serbia: SRB_PAK
- South Korea: KOR_ADDRESS_ID
- United Kingdom: GBR_UDPRN, GBR_UPRN
- Japan: JPN_CHOUMEI_AZA_CODE, JPN_CHOUMEI_AZA_GAIKU_CODE

Informatica added the feature in Address Verification 5.8.0.

Rooftop Geocoding for United Kingdom Addresses

You can retrieve rooftop geocoordinates for a United Kingdom address.

To receive rooftop geocoordinates for United Kingdom addresses, set the following parameters:

- Set the `Type` attribute of the `Enrichment` parameter to GEOCODING.
- Set the `GeoCodingType` parameter in the `Process` element to ROOFTOP.

Informatica added the feature in Address Verification (On-Premises) 5.8.0.

Support for Individual House Numbers and Additional Enrichment of Germany Addresses

You can verify individual house numbers in Germany addresses. You can also retrieve the three-digit street code from the Frachtleitcode (or Freight Code) as an additional enrichment for verified Germany addresses. Street codes identify streets in Germany. Positions 6, 7, and 8 of the Franchtleitcode form the street code. A street code value of 994 indicates that the address points to a packstation. A packstation is a service delivery point of DHL Parcel Germany.

To retrieve the street code as an enrichment, set the `Type` attribute of the `Enrichment` parameter to `SUPPLEMENTARY_DE`. Address Verification returns the street code in the STREET_CODE field.

Informatica added the feature in Address Verification (On-Premises) 5.8.0.

Enrichment for South Korea Addresses

Informatica Address Verification introduces the following changes to South Korea address verification:

Address ID values as an Enrichment to South Korea Addresses

You can retrieve address ID values as an enrichment to valid South Korea addresses. The address ID value is a unique string that Address Verification assigns to South Korea addresses. Use the address ID value to obtain a South Korea address in the current, street-based format or in the older, land lot-based format. You can also use the address ID value to obtain the six-digit post code or the five-digit post code for the address.

To receive the address ID as an enrichment, set the `Type` attribute of the `Enrichment` parameter to `SUPPLEMENTARY_KR`.

Address Code Lookup Support for South Korea

When you use the address ID value to retrieve South Korea address information, configure Address Verification to run in address code lookup mode.

To enable address code lookup for a South Korea address, set the process mode to ADDRESSCODELOOKUP. Set the `CodeType` attribute of the `AddressCode` element to `KOR_ADDRESS_ID`.

Support for the Hangul Script in South Korea Addresses

The default script of the South Korea reference address database is Hangul. You can use Address Verification to verify South Korea addresses in the Hangul script and to receive the result output in Hangul script. Alternatively, you can configure the `PreferredScript` attribute of the `Validation` parameter to an appropriate value to receive the output in the ASCII script.

Support for New Postal Codes in South Korea

Address Verification supports the new, five-digit postal codes and the older, six-digit postal codes in South Korea. To verify the six-digit post codes, set the `MatchingExtendedArchive` attribute of the `Process` parameter to `true` or `1`. If you do not set `MatchingExtendedArchive` to `true` or `1`, Address Verification identifies the six-digit post codes as not valid.

Verification of Outdated South Korea Addresses

You can verify older, land lot-based addresses and addresses with older, six-digit post codes in South Korea. To verify the address and post code data, set the `MatchingExtendedArchive` attribute of the `Process` parameter to `true` or `1`. When you set `MatchingExtendedArchive` to ON, Address Verification returns an extended element result status of F if the address or the post code uses the older format. Address Verification does not return older address information if you set `MatchingExtendedArchive` to `false` or `0`.

Previously, Address Verification identified land lot-based addresses as not valid.

Informatica added the feature in Address Verification 5.8.0.

Support for Macau

You can use Informatica Address Verification to verify addresses from Macau. You can verify Macau addresses in the batch, interactive, or fast completion modes.

The reference databases for Macau contain address information up to the building level. Macau does not have a postal code system. The reference address databases for Macau are in the Portugese language and use the Latin script.

Informatica added the feature in Address Verification (On-Premises) 5.8.0.

Support for Non-Postal Addresses in New Zealand

You can use Informatica Address Verification to verify non-postal addresses, such as highways, parks, private bag, counter delivery, community mail box, and response bag, in New Zealand.

Address Verification verifies non-postal addresses in the same way that it verifies postal addresses. You can verify non-postal addresses in batch, interactive, and fast completion modes. New Zealand Post does not recognize the non-postal addresses or provides such addresses.

Informatica added the feature in Address Verification (On-Premises) 5.8.0.

Multilanguage Support for Province Information in Switzerland Addresses

Informatica Address Verification extends multilanguage support to province information in Switzerland addresses. Address Verification can return the province information in English, German, French, or Italian.

You can specify the language in the `PreferredLanguage` parameter. By default, Address Verification returns the information in the default language of the corresponding reference address database entry.

Informatica added the feature in Address Verification (On-Premises) 5.8.0.

Subbuilding Support for Spain

You can use Informatica Address Verification to return subbuilding information for addresses in Spain. Address Verification supports up to three subbuilding levels in Spain addresses.

Informatica added the feature in Address Verification (On-Premises) 5.8.0.

Support for House Number Verification in Taiwan Addresses

Informatica Address Verification verifies the house number information in Taiwan addresses.

Address Verification verifies the house number information when you verify addresses in Mandarin Traditional Chinese or in English. However, in the interactive and fast completion modes, you receive the best results for house numbers when you enter Taiwan addresses in Mandarin Traditional Chinese.

Informatica added the feature in Address Verification (On-Premises) 5.8.0.

Support for Timor-Leste

You can use Informatica Address Verification to verify addresses from Timor-Leste (East Timor). You can verify addresses from Timor-Leste in batch, interactive, and fast completion modes.

The reference address databases for Timor-Leste contain address information up to the street level. Timor-Leste does not have a postal code system. The reference address databases for Timor-Leste are in the Portuguese language and use the Latin script.

Informatica added the feature in Address Verification (On-Premises) 5.8.0.

Additional Enrichments for United Kingdom Addresses

You can retrieve delivery point type and organization key information as additional enrichments to verified United Kingdom addresses. The organization key is a unique eight-digit code that Royal Mail assigns to small organizations. The delivery point type is a single-character code that indicates whehter the address points to a residence (R), a small organization (O), or a large organization (L).

To receive the delivery point type and organization key, set the `Type` attribute of the `Enrichment` parameter to `SUPPLEMENTARY_GB`.

Informatica added the feature in Address Verification (On-Premises) 5.8.0.

Support for Unique Property Reference Numbers in Address Code LookUp Mode

You can use the Unique Property Reference Number (UPRN) codes to retrieve the corresponding United Kingdom addresses in the AddressCodeLookUp mode. In the AddressCodeLookUp mode, you can enter a UPRN and receive the corresponding address in the result output. UPRN is a numeric code that uniquely identifies a land or property unit in the United Kingdom.

To retrieve United Kingdom addresses using UPRN, set the `ProcessMode` parameter to `ADDRESSCODELOOKUP` and the `CodeType` tag of the `AddressCode` parameter to `GBR_UPRN`.

Informatica added the feature in Address Verification (On-Premises) 5.8.0.

NIS Code as Enrichment to Belgium Addresses

You can retrieve the NIS code as an enrichment to validated Belgium addresses. Informatica Address Verification returns a combination of the five-digit NIS code and a four-digit Neighbourhood ID along with the validated output for the Belgium addresses.

To receive NIS codes in the validated output for Belgium addresses, set the `Type` attribute of the `Enrichment` parameter to `SUPPLEMENTARY_BE`.

Informatica added the feature in AddressDoctor 5.7.0.

Ability to Remove Locality and Province Descriptors from China and Japan Addresses

You can remove the locality and province descriptors from China and Japan addresses if you set the following validation parameters:

- `PreferredLanguage` to ENGLISH.
- `PreferredScript` to ASCII_SIMPLIFIED, ASCII_EXTENDED, LATIN, LATIN_1, or LATIN_ALT.

For example, when you validate China addresses, Address Verification returns Chaoyang instead of Chaoyangqu and Beijing instead of Beijingshi.

When you validate Japan addresses, Address Verification returns Tokyo instead of Tokyo-to and Kyoto instead of Kyoto-fu.

Informatica added the feature in AddressDoctor 5.7.0.

Multilanguage Support for Locality Information in Switzerland Address Output

You can configure Address Verification to return the locality information in Switzerland address results consistently in French, German, or Italian. In previous releases, Address Verification returned all information in the Switzerland address results in the main language of the region to which the address belonged.

You can configure the `PreferredLanguage` parameter to one of the following values and receive the locality information in the corresponding language:

ALTERNATIVE_1

German

ALTERNATIVE_2

French

ALTERNATIVE_3

Italian

If you set `PreferredLanguage` to DATABASE, Address Verification returns the address in the main language of the region to which the address belongs. For example, German in Zurich and French in Geneva.

If you set `PreferredLanguage` to ENGLISH, Address Verification returns the English name of the locality if it is different from the name in the local language. Address Verification returns the other address elements in the main language of the region to which the address belongs. Note that English locality names are available only for Zurich addresses.

If you set `PreferredLanguage` to PRESERVE_INPUT, Address Verification returns the address in the same language in which you input the address. Note that Address Verification preserves the language only if the reference data is available in the same language that you used for the input. If the input address contains more than one language, Address Verification returns the output in the main language of the region.

Additionally, Address Verification returns the Street information for addresses in Biel/Bienne in the alternative language that you configured for the `PreferredLanguage` parameter.

Informatica added the feature in AddressDoctor 5.7.0.

Ability to Validate Bulgaria Addresses in Cyrillic Script

You can validate Bulgaria addresses in the Cyrillic script. By default, you receive the validated results in the Cyrillic script. To receive the results in the Latin script, set the `PreferredScript` parameter to ASCII_EXTENDED, ASCII_SIMPLIFIED, LATIN, LATIN_1, or LATIN_ALT.

The other enhancements to Bulgaria address validation include high-quality reference address data and improved parsing of locality information in Bulgaria address.

Informatica added the feature in AddressDoctor 5.7.0.

Ability to Validate Slovakia Addresses that Contain Street Name Abbreviations

You can validate Slovakia addresses with the following street name abbreviations:

• SNP. Slovak National Uprising	• ČSA. Czechoslovak Army
• čes. Czech	• arm. Army
• gen. General	• kpt. Captain
• mjr. Major	• nár. National
• národ. National	• nám. Square
• nábr. Waterfront	• Máj. The May
• pov. Uprising	• povst. Uprising

Note: Address Verification supports these abbreviations only in the input address and replaces these abbreviations with the postal authority-specified names in the validated output.

Informatica added the feature in AddressDoctor 5.7.0.

FIAS ID as Enrichment to Russia Addresses

You can retrieve the Federal Information Addressing System (FIAS) ID as an enrichment to the validated Russia addresses. FIAS ID is a 16-character alphanumeric string.

To receive FIAS ID values in the validated output for Russia addresses, set the `Type` attribute of the `Enrichment` parameter to `SUPPLEMENTARY_RU`.

Informatica added the feature in AddressDoctor 5.7.0.

Configurable Output Format for Element Descriptors

You can specify the output format for street, building, and subbuilding element descriptors in Australia and New Zealand addresses and the *Strasse* element in Germany addresses.

To specify the output format for the element descriptors, you can configure one of the following values for the `GlobalPreferredDescriptor` parameter:

DATABASE. Returns the element descriptor that the reference database specifies for the address. If the database does not specify a descriptor for the address, Informatica Address Verification copies the input value to the output address. DATABASE is the default value.

LONG. Returns the expanded form of the element descriptor in the input address.

SHORT. Returns the abbreviated form of the element descriptor in the input address.

PRESERVE_INPUT. Copies the element descriptor from the input address to the output address. If the input element descriptor is not an official name, Informatica Address Verification returns the corresponding value from the reference address database. For example, BD is not an official name for boulevard in Australia.

When you validate an Australia address where the input contains BD, Address Verification corrects the descriptor to BVD. BVD is the short form for boulevard in the Australia reference address database.

Informatica added the feature in AddressDoctor 5.6.0.

Related Topics:

- "GlobalPreferredDescriptor" on page 45

Single-Line Address Validation

You can use single-line address validation to validate addresses entered into the `AddressComplete` element as a single line and receive suggestions to complete the address. To use the single-line address validation, set the process mode to `FASTCOMPLETION`.

Informatica added the feature in AddressDoctor 5.5.0.

Related Topics:

- "Example: Single Line Address Validation" on page 81

Address Code Lookup Mode

You can use the `ADDRESSCODELOOKUP` process mode to enter a country-specific address code and retrieve the complete or partial address for the code.

Address Verification returns process status codes that denote the status of an `ADDRESSCODELOOKUP` request. A process status of A0 indicates that Address Verification cannot find any information for the code that you entered. A process status of A1 indicates that a partial or complete address is available for the code that you entered.

Note: Informatica Address Verification treats the `ADDRESSCODELOOKUP` transactions as batch transactions. You must have sufficient batch transactions in your account to use the `ADDRESSCODELOOKUP` process mode.

Informatica added the feature in AddressDoctor 5.4.1.

Related Topics:

- "Example: AddressCodeLookup" on page 87

Ability to Retrieve Address Codes for Outdated Japan Addresses

You can set the `MatchingExtendedArchive` parameter to retrieve the new address code for deprecated or outdated Japan addresses.

Informatica Address Verification validates the old address against the archived addresses in the reference database if you set `MatchingExtendedArchive` to true or 1 and the enrichment type to `SUPPLEMENTARY_JP`. Along with the validated output, Address Verification returns the old Choumei Aza code and the new Choumei Aza code as enrichment values. You can provide the new Choumei Aza code as input for the `ADDRESSCODELOOKUP` processing mode and receive the corresponding new address. If you set `MatchingExtendedArchive` to false or 0, the Address Verification might reject the input address or correct it to some other address.

For outdated addresses, Address Verification returns an Extended Element Result Status (EERS) value of F.

Informatica added the feature in AddressDoctor 5.4.1.

Related Topics:

- “MatchingExtendedArchive” on page 47

Point Address Geocoding

Informatica Address Verification supports the following point address geocoding types.

- Arrival Point (`ARRIVAL_POINT`). The geocoordinates are calculated for a point that is placed in the center of a street segment in front of the house. If the arrival point geocoordinates do not exist, Address Verification uses the Standard Geocode database as a fallback to interpolate the geo-coordinates. Arrival Point geocoding is the default option. For United Kingdom addresses, Informatica Address Verification returns the rooftop geocoordinates if you set the geocoding type to `ROOFTOP`. Rooftop geocoordinates are the measured coordinates for the center of the roof for the primary building on a parcel of land.
- Parcel Centroid (`PARCEL_CENTROID`). The geocoordinates are calculated for a point that is at the geographic center of the parcel of land. If the parcel centroid geocoordinates do not exist, Address Verification returns the status code EGC0 to indicate the geocoordinates are unavailable.

Informatica added the feature in AddressDoctor 5.4.0.

Support for Preserving Alias or Vanity Names for Localities and Streets

You can choose to retain the aliases or vanity names for localities and streets in the validated address output. You can set one of the following values for the `AliasLocality` and `AliasStreet` parameters to specify whether you want the vanity name or the official name in the output.

- **PRESERVE**. To retain the alias name or vanity name in the validated output.
- **OFFICIAL**. To receive the alias or the postal name as mandated by the postal regulations of the country in the validated output.
- **OFF**. To receive the postal name for the locality or street in the output.

The default value is OFFICIAL. If you are validating addresses in the certified mode, set these parameters to OFFICIAL.

Informatica added the feature in AddressDoctor 5.4.0.

Related Topics:

- “AliasStreet and AliasLocality” on page 39

CHAPTER 2

Informatica Address Verification Interface

This chapter includes the following topics:

Interface Overview

You can use SOAP 1.1 or SOAP 1.2 to communicate with Informatica Address Verification. You can access Address Verification over HTTP port 80 or HTTPS port 443. Connections over HTTPS port 443 support encrypted messages and are secure.

SOAP 1.1 and SOAP 1.2 differ in the XML namespaces these two versions use. Additionally, in SOAP 1.2, the action parameter in the `Content-Type` header replaces the `SOAPAction` HTTP header in SOAP 1.1.

Note that the SOAP interface for Address Verification is case-sensitive. Use of wrong cases for elements, parameters, or values could cause the request to fail. For elements and parameters, use exactly the same case that is given in the Request XML section of this document. For values other than boolean values use UPPERCASE. For boolean values, use either lowercase true or false or numeric values 1 or 0.

SOAP 1.1 Headers and Envelope

The following example shows a sample SOAP 1.1 request with the headers and the SOAP envelope XML.

```
POST: /Webservice5/v3/AddressValidation.asmx
Host: 127.0.0.1
Content-Type: text/xml; charset=utf-8 Content-Length: length
SOAPAction: "AddressDoctor/Webservice5/v3/Process"
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
   <soap:Body>
```

```
     <Process xmlns=" http://validator5.AddressDoctor.com/Webservice5/v3">
     </Process>
   </soap:Body>
</soap:Envelope>
```

The following example shows the SOAP response to the preceding request.

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
   <soap:Body>
     <ProcessResponse xmlns="http://validator5.AddressDoctor.com/Webservice5/v3">
     </ProcessResponse>
   </soap:Body>
</soap:Envelope>
```

SOAP 1.2 Headers and Envelope

The following example shows a sample SOAP 1.2 request with headers and envelope. Note the different Content-Type used and xmlns: soap declaration.

```
POST /Webservice5/v3/AddressValidation.asmx HTTP/1.1
Host: 127.0.0.1
Content-Type: application/soap+xml; charset=utf-8 Content-Length: length
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://www.w3.org/2003/05/soap-
envelope">
   <soap:Body>
      <Process xmlns=" http://validator5.AddressDoctor.com/Webservice5/v3">
      </Process>
   </soap:Body>
</soap:Envelope>
```

The following example shows the SOAP 1.2 response to the preceding request.

```
HTTP/1.1 200 OK
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
   <soap:Body>
     <ProcessResponse xmlns=" http://validator5.AddressDoctor.com/Webservice5/v3">
     </ProcessResponse>
   </soap:Body>
</soap:Envelope>
```

Request XML

The following example shows the `<process>` tag and its elements for both the SOAP 1.1 and SOAP 1.2 requests.

```
    <Process xmlns="http://validator5.AddressDoctor.com/Webservice5/v3">
```

```
<login>string</login>
<password>string</password>
<parameters>
  <ProcessMode>string</ProcessMode>
  <ServiceParameters>
    <JobToken>string</JobToken>
    <CampaignId>string</CampaignId>
    <ReservedXml>string</ReservedXml>
    <UseTransactionPool>string</UseTransactionPool>
  </ServiceParameters>
  <ValidationParameters>
    <FormatType>string</FormatType>
    <FormatDelimiter>string</FormatDelimiter>
    <DefaultCountryISO3>string</DefaultCountryISO3>
    <ForceCountryISO3>string</ForceCountryISO3>
    <CountryType>string</CountryType>
    <CountryOfOriginISO3>string</CountryOfOriginISO3>
    <StreetWithNumber>boolean</StreetWithNumber>
    <FormatWithCountry>boolean</FormatWithCountry>
    <ElementAbbreviation>boolean</ElementAbbreviation>
    <PreferredScript>string</PreferredScript>
    <PreferredLanguage>string</PreferredLanguage>
    <AliasStreet>string</AliasStreet>
    <AliasLocality>string</AliasLocality>
    <GlobalCasing>string</GlobalCasing>
    <GlobalMaxLength>int</GlobalMaxLength>
    <GlobalPreferredDescriptor>string</GlobalPreferredDescriptor>
    <MatchingScope>string</MatchingScope>
    <MaxResultCount>int</MaxResultCount>
    <DualAddressPriority>string</DualAddressPriority>
    <StandardizeInvalidAddresses>boolean</StandardizeInvalidAddresses>
    <RangesToExpand>string</RangesToExpand>
    <FlexibleRangeExpansion>boolean</FlexibleRangeExpansion>
    <GeoCodingType>string</GeoCodingType>
    <MatchingAlternatives>string</MatchingAlternatives>
    <MatchingExtendedArchive>boolean</MatchingExtendedArchive>
    <DisableCertifiedModeISO3>string</DisableCertifiedModeISO3>
    <FormatMaxLines>int</FormatMaxLines>
    <Standardizations>
      <Element>string</Element>
      <Casing>string</Casing>
      <MaxLength>int</MaxLength>
      <MaxItemCount>int</MaxItemCount>
    </Standardizations>
    <Standardizations>
      <Element>string</Element>
      <Casing>string</Casing>
      <MaxLength>int</MaxLength>
      <MaxItemCount>int</MaxItemCount>
    </Standardizations>
    <AdditionalInformationSet>
      <Name>string</Name>
      <Value>string</Value>
    </AdditionalInformationSet>
    <AdditionalInformationSet>
      <Name>string</Name>
      <Value>string</Value>
    </AdditionalInformationSet>
    <OutputOptions>
      <RecordId>string</RecordId>
      <Organization>string</Organization>
      <Department>string</Department>
      <Contact>string</Contact>
      <Email>string</Email>
      <Building>string</Building>
      <SubBuilding>string</SubBuilding>
      <Street>string</Street>
      <HouseNumber>string</HouseNumber>
      <DeliveryService>string</DeliveryService>
      <Locality>string</Locality>
      <PreferredLocality>string</PreferredLocality>
      <PostalCode>string</PostalCode>
      <Province>string</Province>
      <Country>string</Country>
      <Residue>string</Residue>
      <RecipientLines>string</RecipientLines>
      <DeliveryAddressLines>string</DeliveryAddressLines>
      <CountrySpecificLocalityLine>string</CountrySpecificLocalityLine>
      <FormattedAddress>string</FormattedAddress>
      <AddressComplete>string</AddressComplete>
      <AddressDetailed>string</AddressDetailed>
    </OutputOptions>
  </ValidationParameters>
</parameters>
<addresses>
  <Address>
    <RecordId>string</RecordId>
    <Organization>
      <string>string</string>
      <string>string</string>
    </Organization>
    <Department>
      <string>string</string>
      <string>string</string>
    </Department>
    <Contact>
      <string>string</string>
      <string>string</string>
    </Contact>
    <Email>
      <string>string</string>
      <string>string</string>
    </Email>
    <Building>
      <string>string</string>
      <string>string</string>
    </Building>
    <SubBuilding>
      <string>string</string>
      <string>string</string>
    </SubBuilding>
    <Street>
      <string>string</string>
      <string>string</string>
    </Street>
    <HouseNumber>
      <string>string</string>
      <string>string</string>
    </HouseNumber>
    <DeliveryService>
      <string>string</string>
      <string>string</string>
    </DeliveryService>
    <Locality>
      <string>string</string>
      <string>string</string>
    </Locality>
    <PreferredLocality>
      <string>string</string>
      <string>string</string>
    </PreferredLocality>
    <PostalCode>
      <string>string</string>
      <string>string</string>
    </PostalCode>
    <Province>
      <string>string</string>
      <string>string</string>
    </Province>
    <Country>
      <string>string</string>
      <string>string</string>
    </Country>
    <Residue>
      <string>string</string>
      <string>string</string>
    </Residue>
    <RecipientLines>
      <string>string</string>
      <string>string</string>
    </RecipientLines>
    <DeliveryAddressLines>
      <string>string</string>
      <string>string</string>
    </DeliveryAddressLines>
    <CountrySpecificLocalityLine>
      <string>string</string>
      <string>string</string>
    </CountrySpecificLocalityLine>
    <FormattedAddress>
      <string>string</string>
      <string>string</string>
    </FormattedAddress>
    <AdditionalAddressInformation>
      <Name>string</Name>
      <Value>string</Value>
    </AdditionalAddressInformation>
    <AdditionalAddressInformation>
      <Name>string</Name>
      <Value>string</Value>
    </AdditionalAddressInformation>
    <AddressComplete>string</AddressComplete>
    <AddressCode>
      <CodeType>string</CodeType>
      <Value>string</Value>
    </AddressCode>
    <AddressDetailed>
      <Organization xsi:nil="true" />
      <Contact xsi:nil="true" />
      <Building xsi:nil="true" />
      <SubBuilding xsi:nil="true" />
      <Street xsi:nil="true" />
      <HouseNumber xsi:nil="true" />
      <DeliveryService xsi:nil="true" />
      <Locality xsi:nil="true" />
      <PostalCode xsi:nil="true" />
      <Province xsi:nil="true" />
      <Country xsi:nil="true" />
      <Residue xsi:nil="true" />
    </AddressDetailed>
  </Address>
  <Address>
    <RecordId>string</RecordId>
    <Organization>
      <string>string</string>
      <string>string</string>
    </Organization>
    <Department>
      <string>string</string>
      <string>string</string>
    </Department>
    <Contact>
      <string>string</string>
      <string>string</string>
    </Contact>
    <Email>
      <string>string</string>
      <string>string</string>
    </Email>
    <Building>
      <string>string</string>
      <string>string</string>
    </Building>
```

```
<SubBuilding>
  <string>string</string>
  <string>string</string>
</SubBuilding>
<Street>
  <string>string</string>
  <string>string</string>
</Street>
<HouseNumber>
  <string>string</string>
  <string>string</string>
</HouseNumber>
<DeliveryService>
  <string>string</string>
  <string>string</string>
</DeliveryService>
<Locality>
  <string>string</string>
  <string>string</string>
</Locality>
<PreferredLocality>
  <string>string</string>
  <string>string</string>
</PreferredLocality>
<PostalCode>
  <string>string</string>
  <string>string</string>
</PostalCode>
<Province>
  <string>string</string>
  <string>string</string>
</Province>
<Country>
  <string>string</string>
  <string>string</string>
</Country>
<Residue>
  <string>string</string>
  <string>string</string>
</Residue>
<RecipientLines>
  <string>string</string>
  <string>string</string>
</RecipientLines>
<DeliveryAddressLines>
  <string>string</string>
  <string>string</string>
</DeliveryAddressLines>
<CountrySpecificLocalityLine>
  <string>string</string>
  <string>string</string>
</CountrySpecificLocalityLine>
<FormattedAddress>
  <string>string</string>
  <string>string</string>
</FormattedAddress>
<AdditionalAddressInformation>
  <Name>string</Name>
  <Value>string</Value>
</AdditionalAddressInformation>
<AdditionalAddressInformation>
  <Name>string</Name>
  <Value>string</Value>
</AdditionalAddressInformation>
<AddressComplete>string</AddressComplete>
<AddressCode>
  <CodeType>string</CodeType>
  <Value>string</Value>
</AddressCode>
<AddressDetailed>
  <Organization xsi:nil="true" />
```

```
            <Contact xsi:nil="true" />
            <Building xsi:nil="true" />
            <SubBuilding xsi:nil="true" />
            <Street xsi:nil="true" />
            <HouseNumber xsi:nil="true" />
            <DeliveryService xsi:nil="true" />
            <Locality xsi:nil="true" />
            <PostalCode xsi:nil="true" />
            <Province xsi:nil="true" />
            <Country xsi:nil="true" />
            <Residue xsi:nil="true" />
          </AddressDetailed>
        </Address>
      </addresses>
      <enrichments>
        <Enrichment>
          <Type>string</Type>
          <PayloadXml>string</PayloadXml>
        </Enrichment>
        <Enrichment>
          <Type>string</Type>
          <PayloadXml>string</PayloadXml>
        </Enrichment>
      </enrichments>
    </Process>
  </soap:Body>
</soap:Envelope>
```

Notes About Elements in the Request XML

- There are mandatory elements and optional elements. You must include the mandatory elements in the request. However, you can choose not to include the optional elements.
- `<login>` and `<password>` are mandatory elements. If you do not include these elements, an authentication error occurs.
- `<ProcessMode>` is a mandatory element.
- `<ServiceParameters>` is optional. However, in certain process modes, some of the service parameters are mandatory (for example, JobToken is needed when you resubmit a request in the FastCompletion mode).
- `<ValidationParameters>` is optional. If you omit this, Informatica Address Verification uses the default values for the validation parameters.
- `<Standardizations>` is optional.
- `<AdditionalInformationSet>` is for Address Verification internal use and is not supported in customer requests.
- `<OutputOptions>` and its elements are optional.
- `<addresses>` is mandatory.
    - At least one `<Address>` element must be supplied.
    - Depending on the process mode specified, more `<Address>` elements can be supplied.
    - If any element of `<Address>` is omitted, that is considered as empty.
- `<enrichments>` is optional.

Response XML

The following example shows the `<ProcessResponse>` tag and its elements.

```
<ProcessResponse xmlns="http://validator5.AddressDoctor.com/Webservice5/v3">
      <ProcessResult>
        <StatusCode>int</StatusCode>
        <StatusMessage>string</StatusMessage>
        <JobToken>string</JobToken>
        <CampaignId>string</CampaignId>
        <ReservedXml>string</ReservedXml>
        <Results>
        <!-- There is one <Result> element for each <Address> element in the request -->
          <Result>
            <ProcessStatus>string</ProcessStatus>
            <CountryISO3>string</CountryISO3>
          <!-- There are between one and twenty <ResultData> elements, depending on
               the ProcessMode and the ProcessStatus -->
            <ResultDataSet xsi:nil="true" />
          <!-- There is one <Enrichment> element for each <Enrichment> element
               in the request -->
            <Enrichments xsi:nil="true" />
        </Result>
      </Results>
    </ProcessResult>
 </ProcessResponse>
```

CHAPTER 3

Informatica Address Verification Request Parameters

This chapter includes the following topics:

Request Parameters Overview

An Informatica Address Verification request contains many parameters including some optional parameters.

The required parameters in a request include login and password parameters, the process mode parameter, and the addresses parameter. Service parameters, validation parameters, and enrichment parameters are optional.

If a parameter requires a Boolean value, enter the value in lowercase letters or as a number. Enter `true` or `false` or 1 or 0.

Login and Password Parameters

Informatica Address Verification processes only those requests that contain valid values for the `login` and `password` parameters.

The `login` parameter contains the Informatica Address Verification Data Quality Center (DQC) account ID. The `password` parameter contains the password for the Informatica Address Verification DQC account ID provided for the `login` parameter.

You can view your Informatica Address Verification DQC account ID and password on the Informatica Address Verification DQC portal. To log in to the portal, use your Informatica Passport credentials. For more information about accessing the Informatica Address Verification DQC portal, see the *Informatica AddressDoctor Cloud Data Quality Center User Guide*.

If you leave the `login` and `password` parameters empty or enter incorrect values for these parameters, Informatica Address Verification returns an Authentication Failed error.

Example

The following example shows an Informatica Address Verification request. Note that the login and password values are masked for security reasons.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns="http://validator5.AddressDoctor.com/Webservice5/v3">
   <soapenv:Header/>
   <soapenv:Body>
        <Process>
            <login>*****</login>
            <password>**********</password>
            <parameters>
                <ProcessMode>INTERACTIVE</ProcessMode>
            </parameters>
            <addresses>
                <Address>
                    <FormattedAddress>
                        <string>U 8 995 BURKE ROAD</string>
                        <string>CAMBERWELL VIC 3124</string>
                        <string>AUSTRALIA</string>
                    </FormattedAddress>
                </Address>
            </addresses>
        </Process>
   </soapenv:Body>
</soapenv:Envelope>
```

ProcessMode Parameter

`ProcessMode` is a required parameter that you must specify in the request. You can specify one of the following values for the `ProcessMode` parameter.

AddressCodeLookup

To enter a country-specific address code and retrieve the complete or partial address for the code, set `ProcessMode` to `ADDRESSCODELOOKUP`. Informatica Address Verification considers the AddressCodeLookup transactions as batch transactions. You must have sufficient batch transactions in your account to use the AddressCodeLookup process mode.

You can use the AddressCodeLookup process mode to retrieve addresses from the following countries:

Country	Supported Codes
Germany	DEU_AGS, DEU_LOCALITY_ID, and DEU_STREET_ID
Japan	JPN_CHOUMEI_AZA_CODE and JPN_CHOUMEI_AZA_GAIKU_CODE
Serbia	SRB_PAK
South Africa	ZAF_NADID

Country	Supported Codes
South Korea	KOR_ADDRESS_ID
United Kingdom	GBR_UDPRN and GBR_UPRN

Batch

To cleanse large data files with minimal user intervention, set `ProcessMode` to `BATCH`. In batch mode, Address Verification corrects addresses but does not return address suggestions for the user to choose from. You can input up to 1000 addresses in a batch-mode request. For each of the addresses present in the request, Address Verification returns one result in the response. Each of the results contains one `ResultData` element.

Certified

To correct and validate addresses according to the postal certifications of Australia, Canada, France, New Zealand, or the United States, set `ProcessMode` to `CERTIFIED`. You can input up to 1000 addresses in a certified-mode request. If you also want Address Verification to return enrichments that are available in certified mode, specify the certification as the enrichment type in the request. If you enter addresses from countries that Address Verification does not support in certified mode, Address Verification processes the requests in batch mode.

The following table lists the postal certifications that Address Verification supports:

Country	Certification	Organization	Enrichment Type
Australia	AMAS	Australia Post	`AMAS`
Canada	SERP	Canada Post	`SERP`
France	SNA	La Poste	`SNA`
New Zealand	SendRight	New Zealand Post	`SEND_RIGHT`

Interactive

To receive address suggestions from which you can choose the best available match for the input address, set `ProcessMode` to `INTERACTIVE`. In the interactive mode you can choose the best possible match from the address suggestions that Address Verification returns.

FastCompletion

To receive address suggestions as you enter an address, set `ProcessMode` to `FASTCOMPLETION`. In the fast completion mode, you receive address suggestions with minimum input. For some countries such as the United Kingdom, you need to enter only the postal code and building number to receive a complete address.

Address Verification also supports the single-line address validation using the `AddressComplete` element in the fast completion mode. Single-line address validation support is available for the following countries:

- Australia
- Canada
- China (Hanzi)

- Germany
- Japan (Kanji and Katakana)
- New Zealand
- United Kingdom
- United States

For more information about single-line address validation in fast completion mode, see "Rules and Guidelines for Single-Line Address Validation" on page 36.

For interactive and fast completion modes, specify a single Address element in the request. The response contains one Result element, but the ResultDataSet might contain up to 100 ResultData elements with the suggestions. The following table lists the process modes and how many elements each array can contain.

ProcessMode	Addresses	Results	ResultDataSet
Batch or Certified	Up to 1000	Up to 1000	0 to 1
Interactive or FastCompletion	1	1	0 to 100

Syntax

```
<ProcessMode>ADDRESSCODELOOKUP | BATCH | CERTIFIED | FASTCOMPLETION | INTERACTIVE</
ProcessMode>
```

Example

The following example shows an Address Verification request for validating an address in the interactive mode.

```
 <soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns="http://validator5.AddressDoctor.com/Webservice5/v3">
    <soapenv:Header/>
    <soapenv:Body>
        <Process>
            <login>*****</login>
            <password>**********</password>
            <parameters>
                <ProcessMode>INTERACTIVE</ProcessMode>
            </parameters>
            <addresses>
                <Address>
                    <FormattedAddress>
                        <string>U 8 995 BURKE ROAD</string>
                        <string>CAMBERWELL VIC 3124</string>
                        <string>AUSTRALIA</string>
                    </FormattedAddress>
                </Address>
            </addresses>
        </Process>
    </soapenv:Body>
</soapenv:Envelope>
```

Rules and Guidelines for Single-Line Address Validation

Consider the following rules and guidelines when you configure Address Verification for single-line validation in fast completion mode:

- Address Verification identifies the address elements in a single-line address input based on their position. You must enter the elements in the correct order even if you leave out some of the elements from the input. However, for optimum results, enter as many details as possible in the input address. When you enter an address in single line, ensure that you do not mix Delivery Address Line (DAL) elements and Country-Specific Last Line (CSLLN) elements.

The following table shows the address element order to use for the supported countries:

Country	Order of Address Elements
Australia	[Sub-building] [House Number] [Street] [Locality] [Province] [Post Code]
Canada	[Sub-building] [House Number] [Street] [Delivery Service] [Locality] [Province] [Post Code]
China	[Post Code] [Province] [Locality] [Street] [House Number] [Sub-building]
Germany	[Street] [House Number] [Post Code] [Locality] [Province]
Japan	[Post Code] [Province] [Main Locality] [Sub-locality] [Chome] Ban] [Go]
New Zealand	[Sub-building] [House Number] [Street] [Delivery Service] [Locality] [Post Code]
United Kingdom	[Sub-building] [House Number] [Street] [Main Locality] [Sub-locality] [Post Code]
United States	[Sub-building] [House Number] [Street] [Locality] [Province] [Post Code]

- Although delimiters are not required in single-line address input, Address Verification processes a comma or a semicolon in the input as an element separator. An input address with delimiters in a Latin script might fetch better suggestions than an address without delimiters. Informatica Address Verification does not support country, organization, building, or contact information in single-line address input.
- If the single-line address input contains only a numeric input, Address Verification considers it as the post code and returns suggestions accordingly. For countries where the house number appears on the left side of the street name or locality, if the single-line address input begins with a number that is followed by a string, Address Verification considers the number as a house number and the following string as the street name or locality. If Address Verification does not find a match for this combination, Address Verification tries to interpret the input as street name without house number or as a combination of post code and locality.
- When there is no perfect match for an input, Address Verification returns multiple suggestions to help you choose the most appropriate result.
- Address Verification returns an N6 process status code if singleiline address validation is not available for the country to which the input address belongs.

Service Parameters

You can optionally include one or more of the following service parameters in an Informatica Address Verification request.

JobToken

`JobToken` is an optional parameter that is applicable only for the interactive and fast completion mode requests. When an interactive or fast completion mode request fails to receive a **V** or **C** result, Informatica

Address Verification includes a job token in the response. A job token is an alphanumeric string that can contain up to 50 characters.

The following is an example of a job token:

```
<JobToken>b89be474-318c-4da3-87c2-f15b2e43c2cb</JobToken>
```

Include `JobToken` in the request when you resubmit an interactive or fast completion mode request that failed to receive a **V** or **C** result. When a request contains a valid job token, Address Verification recognizes the request as a resubmission of a previously uncompleted request. Address Verification does not bill requests that are resubmissions.

You can submit up to three interactive requests, including two resubmissions that contain a job token value, without additional cost. Similarly, in the fast completion mode, you can submit up to five requests, including four resubmissions that contain a job token value, without additional cost. If you use a job token value for more than the maximum permitted attempts, Address Verification bills such requests as new transactions.

CampaignId

`CampaignId` is an optional parameter for which you can specify an arbitrary string value of up to 20 characters. Informatica Address Verification returns this string in its response.

ReservedXml

`ReservedXml` is an optional parameter that is reserved for Informatica Address Verification internal use. You must exclude this parameter from the request or leave this parameter empty.

UseTransactionPool

`UseTransactionPool` is an optional parameter. You can include the `UseTransactionPool` parameter to specify the transaction pool that Informatica Address Verification must use for a call.

You can assign one of the following values for the `UseTransactionPool` parameter:

- **ANY**. Address Verification uses either production or test transaction pools based on the availability. Address Verification tries to use the production transaction pool first and uses the test transaction pool only if no production transactions are available. This is the default value.
- **PRODUCTION**. Address Verification uses only the production transaction pool.
- **TEST**. Address Verification uses only the test transaction pool.

Use the `UseTransactionPool` parameter if you have separate production and development environments. It is technically possible to use both kinds of transactions in either environment. Note that using test transactions for production purposes violates the Terms of Use. Using production (billable) transactions for testing reduces available transactions for the production environment.

Note: If you use the `UseTransactionPool` parameter during development, ensure that you change the value to production when you deploy the application to the production environment. We recommend making this externally configurable, for example through a configuration file or Registry setting. Alternatively, you can use compiler switches or such code to distinguish the production environment from the development environment.

Example: Service Parameters

The following example shows an Address Verification request that contains the service parameters setting:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns="http://validator5.AddressDoctor.com/Webservice5/v3">
    <soapenv:Header/>
    <soapenv:Body>
        <Process>
            <login>*****</login>
            <password>**********</password>
            <parameters>
                <ProcessMode>INTERACTIVE</ProcessMode>
                <ServiceParameters>
                    <JobToken>b89be474-318c-4da3-87c2-f15b2e43c2cb</JobToken>
                    <CampaignId>AddressDoctorTest</CampaignId>
                    <ReservedXml/>
                    <UseTransactionPool>PRODUCTION</UseTransactionPool>
                </ServiceParameters>
            </parameters>
            <addresses>
                <Address>
                    <FormattedAddress>
                        <string>U 8 995 BURKE ROAD</string>
                        <string>CAMBERWELL VIC 3124</string>
                        <string>AUSTRALIA</string>
                    </FormattedAddress>
                </Address>
            </addresses>
        </Process>
    </soapenv:Body>
</soapenv:Envelope>
```

Validation Parameters

Validation parameters are optional. If you do not specify validation parameters in the request, Informatica Address Verification uses the default value for parameters that have default values assigned to them.

AliasStreet and AliasLocality

You can configure the `AliasStreet` and `AliasLocality` parameters to retain the street or locality aliases or vanity names in the validated output. You can set PRESERVE, OFFICIAL, or OFF for `AliasStreet` and `AliasLocality`.

The following table describes the available options for `AliasStreet` and `AliasLocality` parameters.

	PRESERVE	OFFICIAL (Default)	OFF
AliasStreet	Retains the alias for the street in the output.	Returns either the street name alias or the postal name of the street as mandated by the postal regulations of the country.	Returns the postal name for the street in the output.
AliasLocality	Retains the alias for the locality in the output.	Returns either the vanity name or the postal name for the locality as mandated by the postal regulations of the country.	Returns the postal name for the locality in the output.

To validate addresses in the certified mode and generate output that conforms to the postal regulations of the country, set `AliasStreet` and `AliasLocality` to OFFICIAL.

Related Topics:

- "Support for Preserving Alias or Vanity Names for Localities and Streets" on page 24

CountryOfOriginISO3

CountryOfOriginISO3 is an optional parameter that you can include to specify the country from where a mailing originates.

Informatica Address Verification does not include the country name in FormattedAddress if the country in the input is the same as the value you set for CountryOfOriginISO3. Note that this setting works only if you set the FormatWithCountry parameter to **true** or **1**.

If the country of origin and the country in the input address do not match, Address Verification retains the input country information in the output. By default, the CountryOfOriginISO3 parameter is left empty. When this parameter is empty, Address Verification returns the country information in the FormattedAddress field of the output.

Example

```
<CountryOfOriginISO3>AUS</CountryOfOriginISO3>
```

CountryType

CountryType is an optional parameter that you can set to specify the ISO code or the language in which to retrieve the country value in the FormattedAddress field. The default value for the parameter is NAME_EN, which denotes the English name of the country.

The following table describes the supported values for the CountryType parameter:

CountryType	Description
ABBREVIATION	Abbreviation of the country name
ISO2	ISO two-character code
ISO3	ISO three-character code
ISO_NUMBER	ISO three-digit code
NAME_CN	Chinese
NAME_DA	Danish
NAME_DE	German
NAME_EN	English
NAME_KR	Croatian
NAME_NL	Dutch
NAME_PL	Polish
NAME_PT	Portuguese

NAME_RU	Russian
NAME_SA	Arabic
NAME_SE	Swedish
NAME_ES	Spanish
NAME_FI	Finnish
NAME_FR	French
NAME_GR	Greek
NAME_HU	Hungarian
NAME_IT	Italian
NAME_JP	Japanese

DefaultCountryISO3

`DefaultCountryISO3` is an optional parameter that you can include to specify the ISO 3 code of the country to which an address belongs to.

Include this parameter if most or all of the addresses that you enter belong to one country and do not contain the country information. By default, this parameter is left empty. If you leave this parameter empty and Informatica Address Verification fails to recognize the country from the input, Address Verification returns a `ProcessStatus` of **N1**.

Example

```
<DefaultCountryISO3>AUS</DefaultCountryISO3>
```

DisableCertifiedModeISO3

If you set the process mode to CERTIFIED, Informatica Address Verification processes addresses from countries with postal certifications in the certified mode.

You can exclude one or more countries from the certified mode by including the `DisableCertifiedModeISO3` parameter and the ISO 3 code for the countries.

Example

The following example shows the configuration to exclude Australia addresses from the certified mode validation:

```
<DisableCertifiedModeISO3>AUS</DisableCertifiedModeISO3>
```

DualAddressPriority

DualAddresses are addresses that contain both street addresses and delivery service or rural route addresses. Configure the `DualAddressPriority` parameter to specify which address type Informatica

Address Verification validates an input address against if the address contains both street addresses and delivery service or rural route addresses. `DualAddressPriority` is an optional parameter.

You can set the `DualAddressPriority` parameter to one of the following values:

- **DELIVERY_SERVICE**. To validate the delivery service part of the input address.
- **STREET**. To validates the street address part of the input address.
- **POSTAL_ADMIN**. To validate addresses based on the postal department's preference for the address type. This is the default value. When you validate addresses in the certified mode, set `DualAddressPriority` to POSTAL_ADMIN.

ElementAbbreviation

`ElementAbbreviation` is an optional parameter that you can include to specify how Informatica Address Verification handles street and locality names in the output.

If you set `ElementAbbreviation` to **true** or **1** in BATCH or INTERACTIVE mode, Informatica Address Verification abbreviates the locality name when the output is longer than the USPS-specified length.

If you set `ElementAbbreviation` to **true** or **1**, Informatica Address Verification abbreviates the output street name in Germany addresses to 22 characters. Note that Informatica Address Verification abbreviates the street name only if a short name is available in the reference database.

If you set `ElementAbbreviation` to **true** or **1**, Informatica Address Verification excludes the word CHOME from Japan addresses, and returns the CHOME number inside the number field. By default, Informatica Address Verification includes the word CHOME in the street field of the output along with the number of CHOME.

If you set `ElementAbbreviation` to **false** or **0**, Informatica Address Verification returns the street and locality names based on the input, field-length setting, and reference address database entries.

FlexibleRangeExpansion

`FlexibleRangeExpansion` is an optional parameter that you can use in conjunction with the `RangesToExpand` parameter. You can set `FlexibleRangeExpansion` to either **true** or **false**. The default value is **true**. Alternatively, you can use **1** and **0** instead of true and false, respectively.

If you set `FlexibleRangeExpansion` to **true**, Informatica Address Verification does not expand the ranges at the end of the list. Informatica Address Verification returns both individual numbers and number ranges so that the number of suggestions does not exceed the value that you set for `MaxResultCount`. Therefore, a suggestion list might contain both expanded and unexpanded ranges for house numbers and buildings, based on the values you specified for `MaxResultCount`, `RangesToExpand`, and `FlexibleRangeExpansion`.

ForceCountryISO3

`ForceCountryISO3` is an optional parameter that you can configure to override the country recognition capability of Informatica Address Verification.

If you configure `ForceCountryISO3`, Informatica Address Verification ignores the country information present in the data, and applies the country that you specified for this parameter. You can set this parameter to the ISO3 code of a country. By default, this parameter is left empty. If you leave this parameter empty, Informatica Address Verification identifies the country based on one of the following features:

The country recognition capabilities.

The value specified for the `DefaultCountryISO3` parameter.

Example

```
<ForceCountryISO3>AUS</ForceCountryISO3>
```

FormatDelimiter

FormatDelimiter is an optional parameter that you can configure to specify the element separator for address elements in the AddressComplete element. Informatica Address Verification uses this setting for parsing both input and output. Even though single-line address input does not require delimiters, an input with delimiters might fetch better suggestions.

You can specify one of the following delimiters for the parameter:

- **LF** (Line Feed)
- **SEMICOLON** (Default)
- **COMMA**
- **PIPE** (|)
- **TAB**

FormatMaxLines

You can include the FormatMaxLines element to specify the maximum number of lines that you want Informatica Address Verification to return in a multiline FormattedAddress output.

You can specify a value from 1 through 19. The default is 19.

FormatType

FormatType is an optional parameter that you can include in the request to specify what information goes in the FormattedAddress field. You can set one of the following values for the FormatType parameter:

- **ADDRESS_ONLY**. Specifies that the FormattedAddress field contains only DeliveryAddressLine (DAL) and CountrySpecificLocalityLine (CSLLN).
- **WITH_ORGANIZATION**. Specifies that the FormattedAddress field contains organization information, DAL, and CSLLN.
- **WITH_CONTACT**. Specifies that the FormattedAddress field contains contact, DAL, and CSLLN.
- **WITH_ORGANIZATION_CONTACT**. Specifies that the FormattedAddress field contains organization, contact, DAL, and CSLLN.
- **WITH_ORGANIZATION_DEPARTMENT**. Specifies that the FormattedAddress field contains organization, department, DAL, and CSLLN.
- **ALL**. Specifies that the FormattedAddress field contains organization, department, contact, DAL, and CSLLN. This is the default value.

You can set the FormatWithCountry or CountryOfOriginISO3 parameters to include the country name in the formatted address field.

FormatWithCountry

FormatWithCountry is an optional parameter that you can configure to specify whether the country name should be included in the FormattedAddress field.

If you set FormatWithCountry to **false** or **0**, Informatica Address Verification does not include the country name in the FormattedAddress field. If you set FormatWithCountry to **true** or **1**, and configure the

CountryOfOriginISO3 parameter, Informatica Address Verification returns the country name based on the value you set for the latter. Informatica Address Verification does not include the country information in the result if the address belongs to the country specified for CountryOfOriginISO3.

GeoCodingType

You can include the GeoCodingType element to specify the type of geocoordinates that you want Informatica Address Verification to include in the output. For the GeoCodingType setting to work, you must have valid geocoding transactions, and must set the Enrichment type to **GEOCODING**.

You can configure the following values for GeoCodingType:

- NONE. To receive the standard geocoordinates in the result output.
- ARRIVAL_POINT. To receive the arrival point geocoordinates in the result output. ARRIVAL_POINT is the default value.
- PARCEL_CENTROID. To receive the parcel centroid geocoordinates in the result output.
- ROOFTOP. To receive high-precision rooftop geocoordinates for Ireland and United Kingdom addresses.

Related Topics:

- “Enrichments Parameter” on page 56
- “Example: A V4 Address with GeoCoding” on page 77
- “Example: A United Kingdom Address with Building-Level Rooftop GeoCoding” on page 92

GlobalCasing

GlobalCasing is an optional element that you can include to specify the casing for all address elements. You can set one of the following values for GlobalCasing:

NATIVE

To return the result output in the same casing that the data in the reference address database uses.

UPPER

To return the result output in uppercase letters.

LOWER

To return the result output in lowercase letters.

MIXED

To return the result output in mixed case according to country-specific casing guidelines.

NOCHANGE

To return the result output in the same casing that the input address uses.

GLOBAL

To return the result output according to the value set for the GlobalCasing attribute.

GlobalMaxLength

`GlobalMaxLength` is an optional element that you can include to specify the maximum number of characters Informatica Address Verification returns in each line of address elements. Informatica Address Verification abbreviates lines that exceed this value.

The default value for `GlobalMaxLength` is **0**, which denotes that there is no restriction on the maximum number of characters.

GlobalPreferredDescriptor

You can configure the `GlobalPreferredDescriptor` element to specify the output format for descriptors in several countries. Use `GlobalPreferredDescriptor` to set English-language street, building, and subbuilding descriptors in Australia, New Zealand, and United States addresses. Use `GlobalPreferredDescriptor` to specify English-language street and building descriptors in Ireland addresses. The element also applies to the Strasse element descriptor in Germany addresses.

Set one of the following values for `GlobalPreferredDescriptor`:

- **DATABASE**. Returns the element descriptor that the reference database specifies for the address. If the database does not specify a descriptor for the address, Informatica Address Verification copies the input value to the output address. DATABASE is the default value.
- **LONG**. Returns the expanded form of the element descriptor in the input address.
- **SHORT**. Returns the abbreviated form of the element descriptor in the input address.
- **PRESERVE_INPUT**. Copies the element descriptor from the input address to the output address.

  If the input element descriptor is not an official synonym, Informatica Address Verification returns the corresponding value from the reference database. For example, when you validate an Australia address that contains BD for boulevard, Informatica Address Verification corrects the descriptor to BVD. BVD is the standard short term for boulevard in Australia.

The following SOAP request includes a `GlobalPreferredDescriptor` of PRESERVE_INPUT:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:v1="http://validator5.AddressDoctor.com/Webservice5/v3">
    <soapenv:Header/>
    <soapenv:Body>
        <Process xmlns="http://validator5.AddressDoctor.com/Webservice5/v3">
            <login>*******</login>
            <password>********</password>
            <parameters>
                <ProcessMode>INTERACTIVE</ProcessMode>
                <ValidationParameters>
                    <GlobalPreferredDescriptor>PRESERVE_INPUT
     </GlobalPreferredDescriptor>
                </ValidationParameters>
            </parameters>
            <addresses>
                <Address>
                    <FormattedAddress>
                        <string>OCEAN BD 2</string>
                        <string>SEACLIFF PARK 5049</string>
                        <string>SOUTH AUSTRALIA</string>
                        <string>AUS</string>
                    </FormattedAddress>
                </Address>
            </addresses>
        </Process>
    </soapenv:Body>
</soapenv:Envelope>
```

In the following response, Address Verification replaces BD with BVD, which is the standard short form for boulevard in Australia addresses:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/
XMLSchema">
   <soap:Body>
      <ProcessResponse xmlns="http://validator5.AddressDoctor.com/Webservice5/v3">
         <ProcessResult>
            <StatusCode>100</StatusCode>
            <StatusMessage>OK</StatusMessage>
            <Results>
               <Result>
                  <ProcessStatus>C4</ProcessStatus>
                  <CountryISO3>AUS</CountryISO3>
                  <ResultDataSet>
                     <ResultData>
                        <ResultNumber>1</ResultNumber>
                        <MailabilityScore>4</MailabilityScore>
                        <ResultPercentage>97.76</ResultPercentage>
                        <ElementInputStatus>60605040600000000060</ElementInputStatus>
                        <ElementResultStatus>F0F0E070F000000000E0</ElementResultStatus>
                        <ElementRelevance>10101010100000000010</ElementRelevance>
                        <ExtElementStatus>00000000000000000000</ExtElementStatus>
                        <AddressResolutionCode>00000000000000000000
                        </AddressResolutionCode>
                        <AddressType>S</AddressType>
                        <LanguageISO3>ENG</LanguageISO3>
                        <Address>
                           <Street>
                              <string>OCEAN BVD</string>
                           </Street>
                           <HouseNumber>
                              <string>2</string>
                           </HouseNumber>
                           <Locality>
                              <string>SEACLIFF PARK</string>
                           </Locality>
                           <PostalCode>
                              <string>5049</string>
                           </PostalCode>
                           <Province>
                              <string>SA</string>
                           </Province>
                           <Country>
                              <string>AUSTRALIA</string>
                           </Country>
                           <DeliveryAddressLines>
                              <string>2 OCEAN BVD</string>
                           </DeliveryAddressLines>
                           <CountrySpecificLocalityLine>
                              <string>SEACLIFF PARK  SA  5049</string>
                           </CountrySpecificLocalityLine>
                           <FormattedAddress>
                              <string>2 OCEAN BVD</string>
                              <string>SEACLIFF PARK  SA  5049</string>
                              <string>AUSTRALIA</string>
                           </FormattedAddress>
                           <AddressComplete>2 OCEAN BVD;SEACLIFF PARK SA 5049;
                            AUSTRALIA</AddressComplete>
                        </Address>
                     </ResultData>
                  </ResultDataSet>
               </Result>
            </Results>
         </ProcessResult>
      </ProcessResponse>
   </soap:Body>
</soap:Envelope>
```

Related Topics:

- “Configurable Output Format for Element Descriptors” on page 22

MatchingExtendedArchive

You can configure the MatchingExtendedArchive parameter to retrieve the Choumei Aza code for a non-current Japan addresses, to verify a South Korea address in the lot-based format, and to verify an older, six-digit post code in a South Korea address.

If you set MatchingExtendedArchive to true or 1, Informatica Address Verification validates the older address data against the archived addresses in the reference database. If the address is a non-current address, Address Verification returns the address with an extended element result status (EERS) value of F. An EERS value of F indicates that the output address is not the latest version of the address.

If you enable supplementary enrichment for Japan (SUPPLEMENTARY_JP), Address Verification can return the old and new Choumei Aza codes as enrichments in the address result. You can use the new Choumei Aza code as an input in the ADDRESSCODELOOKUP processing mode to retrieve the corresponding newer address.

If you enable supplementary enrichment for South Korea (SUPPLEMENTARY_KR), Address Verification returns an extended element result status of F if the address or the post code uses an older format.

You can verify an older, six-digit post code in a South Korea address when you set the MatchingExtendedArchive parameter to true or 1. If you do not set MatchingExtendedArchive to true or 1, Address Verification determines that six-digit post codes are not valid.

Address Verification does not return older address information if you set MatchingExtendedArchive to false or 0. If you set MatchingExtendedArchive to false or 0, Address Verification might reject the older input address or incorrectly return another address as the corrected output.

Related Topics:

- “Ability to Retrieve Address Codes for Outdated Japan Addresses” on page 23

MatchingScope and MatchingAlternatives

MatchingScope and MatchingAlternatives are optional parameters that you can include to specify the scope of address validation. You can configure these parameters to specify the level of address matching that you want Informatica Address Verification to do while validating an address.

The MatchingScope parameter defines the level of granularity to which Informatica Address Verification validates an address. The MatchingAlternatives parameter specifies whether to consider alternate names while validating an address.

You can set the MatchingScopeparameter to specify which address elements that you want Informatica Address Verification to validate. You can set the matching scope to locality, street, or delivery-point level. The default value of MatchingScope is ALL so that Informatica Address Verification validates all possible elements including locality, street, and delivery-point elements in the input address. You receive the best results if you set MatchingScope to ALL as Informatica Address Verification validates all possible elements. If you do not want Informatica Address Verification to consider some of the address elements for validation, you can set MatchingScope to a value other then ALL.

If the reference address database does not contain the specified level of information, Informatica Address Verification validates only those elements that the database contains.

You can configure Informatica Address Verification to validate address elements based on synonyms or alternate names that are available in the reference address database. Synonyms of an address element include alternate names of a place or a street. For example, vanity names in the U.S. addresses. The archival name of an address element refers to an outdated name of a place or street.

To match input address elements with synonyms or archival names that are available in the reference address database, set `MatchingAlternatives` to SYNONYM_ONLY, ARCHIVE_ONLY, or ALL. The default value of `MatchingAlternatives` is ALL. You can set `MatchingAlternatives` to NONE if you do not want Informatica Address Verification to consider synonyms or archival names when it validates an address.

You can set one of the following values for `MatchingScope`:

- **LOCALITY_LEVEL**. To ignore information other than the Locality and Postal Code information while validating the address.
- **STREET_LEVEL**. To validate up to the street information including the street name along with the LOCALITY_LEVEL elements. If you set `MatchingScope` to STREET_LEVEL, Informatica Address Verification ignores information such as house number and subbuilding.
- **DELIVERYPOINT_LEVEL**. To validate up to the delivery point information including house number and post box number and ignore further elements such as building, subbuilding, and organization.
- **ALL**. To validate all elements, including street, locality, and delivery point information, that Informatica Address Verification can validate.

You can include `MatchingAlternatives` to suppress the use of historical and synonym and exonym data for matching address elements. You can set one of the following values for `MatchingAlternatives`:

- **NONE**
- **SYNONYM_ONLY**
- **ARCHIVE_ONLY**
- **ALL**

Note that the actual scope of address validation is dependent on the corresponding country database.

MaxResultCount

`MaxResultCount` is an optional parameter that you can include to specify the maximum number of suggestions that you want to receive in interactive and fast completion modes.

The maximum allowed value for this parameter is 100. The default value is 20.

OutputOptions

`OutputOptions` is an optional parameter that you can include to specify which of the address elements you need in the output. If you do not include `OutputOptions` in the request, Informatica Address Verification returns all the available address elements except `PreferredLocality`, which is **OFF** by default.

You can use the `OutputOptions` parameter to filter out information that is not relevant for you.

Example

The following example shows the `OutputOptions` setting that filters out Contact and Organization information from the output.

```
<OutputOptions>
   <Contact>OFF</Contact>
   <Organization>OFF</Organization>
</OutputOptions>
```

The following example shows the `OutputOptions` setting to include `PreferredLocality` in the output.

```
<OutputOptions>
   <PreferredLocality>ON</PreferredLocality>
</OutputOptions>
```

Currently, Informatica Address Verification supports only **ON** and **OFF** for the `OutputOptions` settings.

PreferredLanguage

Informatica Address Verification can return address elements in multiple languages for addresses in Belgium, Canada, China, Ireland, and Switzerland. To determine the languages that Address Verification uses for the address elements, configure the `PreferredLanguage` parameter.

When the reference data includes elements in multiple languages, Address Verification can return address data in the following languages:

- The main spoken language of the region to which each address belongs. This is the default language for the address in the reference data.
- Any other language that Address Verification supports for an address. For example, the Belgium reference data contains address elements in Flemish, French, and German.

The reference data might contain data for a single address element or for a complete address in multiple languages. For example, Address Verification can return all address elements for Ireland in the English language and can return street names, locality names, and province names in the Irish language. Additionally, the reference data might specify different default languages for different parts of a country. For example, in the Switzerland reference data, the default language for addresses from Zurich is German, from Geneva is French, and from Ticino is Italian.

By default, Informatica Address Verification returns the addresses in the default language of the corresponding reference address database entry.

You can set one of the following values for the `PreferredLanguage` parameter:

- **DATABASE**. Returns all available address elements in the default language of the address in the reference data. This is the default value for `PreferredLanguage`.
- **ENGLISH**. Returns province and locality information in English if the reference data contains the information. For example, Address Verification can replace München with Munich and Bayern with Bavaria in Germany addresses.
- **ALTERNATIVE_1**, **ALTERNATIVE_2**, or **ALTERNATIVE_3**. Returns address elements in one of the alternative languages in the reference data. The alternative languages depend on the country to which the address belongs.
- **PRESERVE_INPUT**. Returns address elements in the same language as the input address. Address Verification preserves the language if the reference data contains the address in the input language. If the input address contains multiple languages, Address Verification returns the address in the default database language.

If you set `PreferredLanguage` to a language that is not available in the reference address database, the `Preferredlanguage` setting defaults to the DATABASE value. In such cases, Informatica Address Verification returns an extended element result status (EERS) value of E. The value indicates that the output uses the database language because the reference database does not contain the information in the language that you specified.

The language or format in which Informatica Address Verification returns the country name depends on the value you configure for the `CountryType` validation parameter.

If the input address contains an address element that the reference database does not contain, Informatica Address Verification copies the element to the result output. For such elements, Informatica Address Verification returns the information in the same language as that of the address input. An element input status value of 1 indicates that the reference address database does not contain the element. An element result status value of 1 indicates that Informatica Address Verification has copied the information from the input because the reference database does not contain the information.

Alternative Language Options for Multilanguage Countries

The following table lists the alternative language options for the multilanguage countries:

Country	ENGLISH	ALTERNATIVE_1	ALTERNATIVE_2	ALTERNATIVE_3
Belgium	Returns the province, locality, and street information in English if the reference address database contains the information in English. Returns the other address elements in the main language of the region to which the address belongs.	Flemish	French	German
Canada	Returns all address information in English.	English	French	Not available
China	Returns the street descriptor and street directional information in English. Returns all other address information in Pinyin.	Hanzi	Not available	Not available
Ireland	Returns all address elements in English.	English	Returns the street, locality, and county information in Irish. Returns all other address information in English.	Not available
Switzerland	Returns the locality and province information in English if the reference address database contains the information in English. Returns the other address elements in the main language of the region to which the address belongs. Address Verification returns the locality information in English for select localities such as Geneva and Zurich.	Returns the province and locality information in German. Returns the other address elements in the main language of the region to which the address belongs.	Returns the province and locality information in French. Returns the other address elements in the main language of the region to which the address belongs.	Returns the province and locality information in Italian. Returns the other address elements in the main language of the region to which the address belongs.

Note: When you set `PreferredLanguage` to ENGLISH for China addresses, Address Verification returns the street descriptor and street directional information in English and all other address information in Pinyin.

PreferredScript

The alphabet in which Informatica Address Verification returns the output varies from country to country. For most of the countries, the output is in Latin 1 or ASCII irrespective of the preferred language you set. `PreferredScript` is an optional parameter that you can include to specify the script in which Informatica Address Verification returns the result.

You can specify one of the following values for the `PreferredScript` parameter:

- **DATABASE**. To generate the address output in the same unicode script in which the reference address database stores the corresponding address. This is the default value.
- **POSTAL_ADMIN_PREF**. To generate the address output in the same unicode script that is the preferred script for the postal department of the country to which the address belongs. Typically, this is the same as the DATABASE script.
- **POSTAL_ADMIN_ALT**. To generate the address output in the same unicode script that is the postal department-approved alternative script in the country to which the address belongs. For example, if you set `PreferredScript` to POSTAL_ADMIN_ALT when you validate a Japan address, you receive the result in Kana characters instead of the Kanji script. Kanji is the POSTAL_ADMIN_PREF script for Japan.
- **ASCII_SIMPLIFIED**. To generate the address output in ASCII simplified script. Informatica Address Verification converts non-ASCII characters in the output to their basic ASCII equivalents. For example, ö is replaced with o in the output.
- **ASCII_EXTENDED**. To generate the address output in ASCII extended script that supports the expansion of special characters. Informatica Address Verification converts non-ASCII characters in the output to their general ASCII equivalents. For example, ö is replaced with oe in the output.
- **LATIN**. To generate the result in Latin script. If the input was in a different script, Informatica Address Verification transliterates such addresses to provide the output in Latin. Note that the Latin output might contain characters from any of the 15 different Latin character sets.
- **LATIN_ALT**. To generate the result in Latin alternate script that is used for transliteration. This settings works only for countries that support multiple Latin transliterations. For example, if you set `PreferredScript` to LATIN_ALT, Informatica Address Verification transliterates Russian addresses according to BGN rules instead of ISO rules.
- **LATIN_1**. To generate the result in Latin-1, Western European, characters.
- **PRESERVE_INPUT**. To generate the address output in the same script that you used for the address input. You can set this value only for addresses from Belarus, China, Greece, Japan, Kazakhstan, Macedonia, Russia, and Ukraine.

Note: If the input contains address elements that are not in the corresponding database, Informatica Address Verification copies such elements from the input to the output. For such elements, Informatica Address Verification uses the same script that you used for the input address irrespective of the value you set for `PreferredScript`.

If you set `PreferredScript` to PRESERVE_INPUT, Informatica Address Verification returns the output in the same script that was used for the input. If the input contains more than one script, Informatica Address Verification overrides the PRESERVE_INPUT setting and returns the address in the default script used in the reference address database.

For example: If you set `PreferredScript` to PRESERVE_INPUT and enter a Japan address that contains both Kanji and Latin elements, Informatica Address Verification returns all address elements in Kanji, which is the default script for Japan reference address database.

Countries and PreferredScript Options

The following table lists the preferred scripts that Address Verification can specify for different countries:

Country	DATABASE	POSTAL _ADMIN _PREF	POSTAL _ADMIN _ALT	LATIN	LATIN _ALT	LATIN _1
BLR	Belarusian Cyrillic	Belarusian Cyrillic	Belarusian Cyrillic	ASCII, transliterated to the Belarusian database standard	ASCII, transliterated to the Belarusian BGN standard	ASCII, transliterated to the Belarusian BGN standard
BGR	Bulgarian Cyrillic	Bulgarian Cyrillic	Bulgarian Cyrillic	ASCII, transliterated to the Bulgarian BGN standard	ASCII, transliterated to the Bulgarian BGN standard	ASCII, transliterated to the Bulgarian BGN standard
CHN	Hanzi	Hanzi	Hanzi	Latin, Mandarin transliteration	Latin, Cantonese transliteration	Latin-1
CZE	Latin-2	Latin-2	Latin-2	ASCII	Latin-2	Latin-1
GRC	Greek	Greek	Greek	Latin-1, transliterated to the ISO standard	ASCII, transliterated to the BGN standard	ASCII
HKG	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII
HUN	Latin-2	Latin-2	Latin-2	ASCII	Latin-2	Latin-1
ISR	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII
JPN	Kanji	Kanji	Kana	Latin-7	Latin-7	ASCII
KAZ	Cyrillic	Cyrillic	Cyrillic	Latin-2, transliterated to the ISO standard	ASCII, transliterated to the BGN standard	ASCII, transliterated to the BGN standard
KOR	Hangul	Hangul	Hangul	ASCII, transliterated using Revised Romanization	ASCII, transliterated using ISO/TR 11941 Romanization	ASCII, transliterated using Revised Romanization
LVA	Latin-7	Latin-7	Latin-7	ASCII	Latin-7	ASCII
MDA	Latin-2	Latin-2	Latin-2	ASCII	Latin-2	Latin-1
MKD	Cyrillic	Cyrillic	Cyrillic	ASCII	ASCII, transliterated to the Macedonian BGN standard	ASCII, transliterated to the Macedonian BGN standard
POL	Latin-2	Latin-2	Latin-2	ASCII	Latin-2	Latin-1

Country	DATABASE	POSTAL _ADMIN _PREF	POSTAL _ADMIN _ALT	LATIN	LATIN _ALT	LATIN _1
ROU	Latin-3	Latin-3	Latin-3	ASCII	Latin-3	Latin-1
RUS	Cyrillic	Cyrillic	Cyrillic	Latin-2, transliterated to the ISO standard	ASCII, transliterated to the BGN standard	ASCII, transliterated to the BGN standard
SVK	Latin-2	Latin-2	Latin-2	ASCII	Latin-2	Latin-1
TWN	Hanzi	Hanzi	Hanzi	Latin, Mandarin transliteration	Latin, Cantonese transliteration	Latin-1
UKR	Cyrillic	Cyrillic	Cyrillic	ASCII	ASCII, transliterated to the Ukrainian BGN standard	ASCII, transliterated to the Ukrainian BGN standard

Note: Address Verification can return addresses in ASCII-simplified and ASCII-extended characters for each country.

RangesToExpand

You can include the optional parameter`RangesToExpand` to specify whether Informatica Address Verification should expand the house number ranges for addresses from countries where individual house numbers exist.

You can set one of the following values of this parameter:

- **NONE**. Informatica Address Verification does not expand house number ranges. This is the default value.
- **ALL**. Informatica Address Verification expands the house number ranges for all addresses where individual house numbers exist
- **ONLY_WITH_VALID_ITEMS**. Informatica Address Verification expands only those ranges where all expandable items exist in the reference data.

For example, if you set `RangesToExpand` to **ONLY_WITH_VALID_ITEMS**:

If the database contains individual house numbers, Informatica Address Verification expands the house number range, and lists the individual house numbers in the suggestion list. For example, the United Kingdom addresses.

For countries that have only house number ranges in the reference database, Informatica Address Verification does not expand the ranges. In such cases, Informatica Address Verification returns the house number ranges in the output.

Standardizations

You can use the standardization elements to specify the `Casing`, `MaxLength`, and `MaxItemCount` values at the address element level, When you set the elements at the address element level, you override the corresponding global settings.

For the `Casing` element, the default value is **GLOBAL**. If you set `Casing` to **GLOBAL**, Informatica Address Verification applies the `GlobalCasing` value to the `Casing` element of the Standardizations list. You can also set the element to **NATIVE**, **UPPER**, **LOWER**, **MIXED**, or **NOCHANGE**. For more information about the values, see the *GlobalCasing* section.

The `MaxLength` element works the same way as the `GlobalMaxLength` parameter works. However, the `MaxLength` element under the Standardizations list affects only the specified address element.

`MaxItemCount` has different value ranges for various elements. The following table describes the `MaxLines`, `MaxLength`, and `Casing` attributes and supported values for different address elements:

Address Element	Default Value for GlobalCasing	Minimum Recommended value for MaxLength	Value Range Supported for MaxItemCount	Default Value for MaxItemCount
RecipientLine	NOCHANGE	25	1 through 6	6
DeliveryAddressLine	NOCHANGE	25	1 though 6	6
CountrySpecificLocalityLine	NOCHANGE	25	1 through 6	6
FormattedAddressLine	NOCHANGE	25	1 through 19	19
AddressComplete	GLOBAL	25	Not Applicable	Not Applicable
Country	GLOBAL	2	1 through 3	3
Locality	GLOBAL	20	1 through 6	6
PostalCode	GLOBAL	5	1 through 3	3
Province	GLOBAL	2	1 through 6	6
Street	GLOBAL	20	1 through 6	6
Number	GLOBAL	5	1 through 6	6
Building	GLOBAL	25	1 through 6	6
SubBuilding	GLOBAL	25	1 through 6	6
DeliveryService	GLOBAL	25	1 through 3	3
Organization	GLOBAL	25	1 through 3	3
Department	GLOBAL	25	1 through 3	3
Contact	GLOBAL	25	1 through 3	3
FirstName	GLOBAL	20	1 through 3	3
MiddleName	GLOBAL	20	1 through 3	3
LastName	GLOBAL	20	1 through 3	3
Title	GLOBAL	20	1 through 3	3

Address Element	Default Value for GlobalCasing	Minimum Recommended value for MaxLength	Value Range Supported for MaxItemCount	Default Value for MaxItemCount
Function	GLOBAL	20	1 through 3	3
Salutation	GLOBAL	20	1 through 3	3
Gender	GLOBAL	1	1 through 3	3
Residue	GLOBAL	25	1 through 6	6

StandardizeInvalidAddresses

You can set the optional parameter`StandardizeInvalidAddresses` to specify whether Informatica Address Verification should standardize addresses that are not valid (`ProcessStatus` = `Ix`) .

You can set this parameter to either **true** or **false**. Alternatively, you can set this parameter to **1** or **0** instead of true or false, respectively. If you set this parameter to true, Informatica Address Verification standardizes the following address elements even when the address is not valid:

- Street Types
- Pre and Post Directional
- Delivery Service Item
- Sub-building descriptors
- State/Province/Region (for example, California to CA)

Note that standardization of addresses can improve downstream business processes such as matching and de-duplication.

StreetWithNumber

You can set the optional parameter `StreetWithNumber` to specify whether house number should be included in the Street field.

If you set this parameter to **true** or **1**, Informatica Address Verification formats the house number to its correct position – before or after the street name – based on the local address conventions. By default, this parameter is set to **false** or **0**.

Addresses Parameter

`Addresses` is a required parameter that contains an array of Address objects.

You can use the `Addresses` parameter to enter addresses for validation. If you set `ProcessMode` to **INTERACTIVE** or **FASTCOMPLETION**, you can enter only one address. If you set `ProcessMode` to **BATCH** or **CERTIFIED**, you can enter up to 1000 addresses.

Note: All address elements are defined as arrays because address formats differ from country to country. In some countries, there are two or more street lines, such as street and dependent street, whereas some other countries have multiple entries for locality. The current Informatica Address Verification version supports up to six lines for each address element.

Enrichments Parameter

The `enrichments` parameter is an optional parameter that contains an array of `Enrichment` objects. Enrichments supplement the address record with useful information that is not part of the regular address.

Set the `Type` attribute of the `Enrichment` parameter to specify the type of enrichment that you want to receive for the address. For some enrichments, you need transactions of the corresponding type.

The following table describes the values that Informatica Address Verification supports for the `Type` attribute:

Enrichment Type	Description
AMAS	Returns any additional information that is available as part of the certified mode validation of Australia addresses.
CAMEO	Returns CAMEO consumer profiles from the Callcredit Information Group. To receive CAMEO consumer profiles as enrichments, you need transactions of type `CAMEO`.
GEOCODING	Returns geocoordinates for the validated address. To receive geocoordinates as address enrichments, you need transactions of type `GEOCODING`.
SEND_RIGHT	Returns any additional information that is available as part of the certified mode validation of New Zealand addresses.
SERP	Returns any additional information that is available as part of the certified mode validation of Canada addresses.
SNA	Returns any additional information that is available as part of the certified mode validation of France addresses.
SUPPLEMENTARY_AT	Returns country-specific enrichments for validated Austria addresses.
SUPPLEMENTARY_BE	Returns country-specific enrichments for validated Belgium addresses.
SUPPLEMENTARY_BR	Returns country-specific enrichments for validated Brazil addresses.
SUPPLEMENTARY_CH	Returns country-specific enrichments for validated Switzerland addresses.
SUPPLEMENTARY_DE	Returns country-specific enrichments for validated Germany addresses.
SUPPLEMENTARY_ES	Returns country-specific enrichments for validated Spain addresses.
SUPPLEMENTARY_FR	Returns country-specific enrichments for validated France addresses.
SUPPLEMENTARY_GB	Returns country-specific enrichments for validated United Kingdom addresses.
SUPPLEMENTARY_IT	Returns country-specific enrichments for validated Italy addresses.
SUPPLEMENTARY_JP	Returns country-specific enrichments for validated Japan addresses.

Enrichment Type	Description
SUPPLEMENTARY_KR	Returns country-specific enrichments for validated South Korea addresses.
SUPPLEMENTARY_PL	Returns country-specific enrichments for validated Poland addresses.
SUPPLEMENTARY_RS	Returns country-specific enrichments for validated Serbia addresses.
SUPPLEMENTARY_RU	Returns country-specific enrichments for validated Russia addresses.
SUPPLEMENTARY_US	Returns country-specific enrichments for validated United States addresses.
SUPPLEMENTARY_ZA	Returns country-specific enrichments for validated South Africa addresses.

Related Topics:

- "GeoCodingType" on page 44
- "Example: A V4 Address with GeoCoding" on page 77
- "Example: A United Kingdom Address with Building-Level Rooftop GeoCoding" on page 92

CHAPTER 4

Informatica Address Verification Response Elements

This chapter includes the following topics:

Response Elements Overview

The Response object contains general status information, results, and any other information that you have requested.

If the StatusCode indicates a successful transaction, Informatica Address Verification returns the results and any other information, such as enrichments, that you have requested.

If the StatusCode indicates an error, the StatusMessage contains a description of the error. In such cases, Informatica Address Verification leaves out the rest of the elements or returns the elements empty.

Status Codes and Messages

Status code is a number that indicates the status of a request, including any error condition that prevents Informatica Address Verification from processing a request.

The following table lists the status codes and status messages.

Status Code	Status Message
100	“OK” No error. The Results array contains the processed result output.
200	“Parameters must be supplied” The request does not contain the Parameters element.
201	"Invalid value for parameterEx Name"
202	“Invalid value for parameter ProcessMode value”
203	"Invalid value for parameter MatchingAlternatives value"
204	"Invalid value for parameter MatchingScope value"
205	"Invalid value for parameter FormatDelimiter value"
206	"Invalid value for parameter CountryType value"
207	"Invalid value for parameter PreferredScript value"
208	"Invalid value for parameter PreferredLanguage value"
209	"Invalid value for parameter GlobalCasing value"
210	"Invalid value for parameter FormatType value"
211	"Invalid value for parameter UseTransactionPool: value"
212	"Invalid value for parameter AliasStreet: value"
213	"Invalid value for parameter MaxResultCount: value"
214	"Invalid value for parameter DualAddressPriority: value"
215	"Invalid value for parameter RangesToExpand: value"
216	"Invalid value for parameter AliasLocality: value"
217	"DisableCertifiedModeISO3 must be one (or more) ISO3 code(s) (separated by |). Invalid value provided: value"
218	"Invalid value for parameter GlobalPreferredDescriptor: value"
219	"Invalid value for parameter FormatMaxLines (1-19): value"
220	"Invalid value for parameter GeoCodingType: value"

Status Code	Status Message
290	"Invalid Standardize Element value"
291	"Invalid Standardize Casing value"
299	"Invalid request - no values provided"
300	"Addresses must be supplied" The request does not contain the Addresses element.
301	"Addresses must contain at least one element"
302	"Addresses must contain no more than 1000 elements for ProcessMode value"
303	"Addresses must contain exactly one element for ProcessMode value"
400	"Enrichments not supported for ProcessMode value" Informatica Address Verification does not support enrichments in the process mode that you specified; for example, fast completion.
500	"Authentication failed" Either the login and password combination is not correct, or you do not have valid transaction for the process mode you specified.
501	"Insufficient transactions" You do not have sufficient transactions for the process mode that you specified.
502	Webservice not available
900	"Server error" An unexpected error has occurred and no details are available.
901	"The Address Verification service is temporarily not available"
902	"ProcessEX is no longer supported"

Process Status Codes

The ProcessStatus element indicates the result of an address validation process. ProcessStatus values contain an alphabet followed by a numeral.

The alphabet in the process status code indicates the general result or status category, whereas the digit denotes a quality grade. Higher numeric values indicate better results in a category. The following table describes the ProcessStatus codes.

Category	Process Status	Description
Verified	V4	The input data is correct. Informatica Address Verification checked all postally-relevant elements, and input matched perfectly.
	V3	The input data is correct, but Informatica Address Verification standardized some or all elements, or the input contains outdated names or exonyms.
	V2	The input data is correct, but Informatica Address Verification could not verify some elements because of incomplete reference data.
	V1	The input data is correct but user standardization has adversely affected the deliverability. For example, the postcode length is too short.
Corrected	C4	Informatica Address Verification has checked all postally-relevant elements.
	C3	Informatica Address Verification could not check some of the elements.
	C2	Delivery status is unclear because the reference data does not contain corresponding information.
	C1	Delivery status is unclear because user standardization has introduced errors.
Incorrect	I4	Informatica Address Verification could not correct the data completely, even though there is a single match with an address in the reference data. Note that the deliverability of an I4 address is very poor.
	I3	Informatica Address Verification could not correct the data completely because there are multiple matches in the reference data. Note that the deliverability of an I3 address is very poor.
	I2	Informatica Address Verification could not correct the data. Batch mode returns partial address suggestions. Note that the deliverability of an I2 address is very poor.
	I1	Informatica Address Verification could not correct the data completely. Batch mode does not have any suggestion to make. Note that the deliverability of an I1 address is very poor.
Fast Completion	Q3	Informatica Address Verification could retrieve multiple complete address suggestions for the input address.
	Q2	Informatica Address Verification could combine the input address elements and elements from the reference address data to generate a complete address suggestion.

Category	Process Status	Description
	Q1	Informatica Address Verification could not return any complete address suggestions. To generate a complete address suggestion, add data to the input address.
	Q0	The input address does not have sufficient information to receive suggestions.
Not Processed	N1	Informatica Address Verification could not recognize the country and hence could not validate the address.
	N2	Informatica Address Verification could not validate the address because the corresponding reference database was not available.
	N6	Informatica Address Verification could not validate the address because the single-line address verification is not supported for the destination country.
Web Service	W1	Not allowed - No transactions available for the specified country.
	W7	Input error - Engine has rejected Parameters or Input Data.
	W8	Timeout - Address was not processed.
	W9	Error - Address was not processed.

JobToken

Informatica Address Verification includes a JobToken in the response when the following conditions are true:

- The ProcessMode value is INTERACTIVE or FASTCOMPLETION.
- The ProcessStatus value is not Vx or Cx.
- The JobToken element in the request was omitted or empty.

If the first two conditions are not true, Informatica Address Verification does not include the JobToken element in the response.

If the first two conditions are true and you pass an invalid JobToken in the input request, Informatica Address Verification returns the invalid JobToken as output.

CampaignID

Informatica Address Verification returns the `CampaignId` that you included in the request.

ReservedXml

This element is reserved for Informatica Address Verification internal use.

Enrichments (Response)

Enrichments supplement the address with useful information that is not part of the actual address. Informatica Address Verification supports many enrichments that provide additional information about addresses. Most of the enrichments require you to have valid transactions for the requested enrichment type in addition to the address validation transactions.

If your account has valid enrichment transactions and you have included the Enrichment parameters in the request, Informatica Address Verification returns enrichment values in the response.

Informatica Address Verification considers postal certifications as enrichments and handles certification-related errors and status messages in the same way it handles other enrichment-related status or error messages.

Informatica Address Verification returns the enrichment information in the `PayloadXml` field. Typically, the `PayloadXml` field contains at least a `Status` element. There are several status values that are universal for all enrichments. Based on the enrichment type, Informatica Address Verification returns the status value either in plain text or as a 4-letter status code.

Enrichment Error Codes and Messages

Informatica Address Verification provides enrichment information in the `PayloadXml` field. Typically, the `PayloadXml` field contains at least a `Status` element. There are several status values that are universal for all enrichments. Based on the enrichment type, Informatica Address Verification returns the status values as either plain text or a 4-letter status code.

If an error occurs, Informatica Address Verification returns only the status message or code (`<Status>`Message or code`</Status>`). If the request was successful, Informatica Address Verification adds an XML node with the enrichment type as its root in the response XML. For example: `<GEOCODING><Status/> <LATITUDE/> <LONGITUDE/></GEOCODING>`.

Error Message and Code	Description
INVALID_ENRICHMENT_TYPE, WSEI	The Type element of the requested enrichment is not valid.
NOT_AVAILABLE_AT_THIS _LOCATION, WSEL	The requested enrichment is not available for the user location. Some of the enrichments are location-dependent. Such enrichments can only be performed by a server physically located in a certain country. Informatica Address Verification hosts several data centers around the world. If you get this error, you have requested a location-based enrichment from a server located outside of the country where the server on which the enrichment database is stored. Each data center has its own URL that enables you to send requests to a data center that has the location-based enrichment available. Contact Informatica Address Verification for more information and the URL of the data center you need to access.

Error Message and Code	Description
NOT_ALLOWED, WSEP	The account that is trying to access the enrichment information does not have the necessary permissions to do so. Contact your Informatica Address Verification sales representative to obtain necessary permissions.
NO_MORE_TRANSACTIONS, WSET	The account that is trying to access enrichment information does not have valid enrichment transactions. Please contact Informatica Address Verification Sales to purchase the necessary transactions.
NOT_AVAILABLE, EMCN, ECON, EMON, EGCN	The requested enrichment is not available for the country you specified. Some of the enrichments are available only for addresses from certain countries.
NOT_REQUESTED, WSEN	The enrichment was not requested because the address validation failed (that is, the `ProcessStatus` for address validation was not V or C). Informatica Address Verification does not send addresses that are not validated to external providers of enrichments.
SERVICE_DOWN, WSED	Web service of the external provider did not respond to the request or responded with an error flag.
TIMEOUT, WSED	The request was aborted to ensure a timely response to your request.

AMAS Status Codes and Output Fields

EAM0

AMAS output is not available for this address.

EAM1

AMAS output is provided – address is corrected or validated and DPID is delivered.

EAM2

AMAS output is not provided – no correction or validation possible. No DPID can be returned

Output Fields

ERRORCODE | RECORD_TYPE | DELIVERY_POINT_ID | LOT_NBR | POSTAL_DELIVERY_NBR | POSTAL_DELIVERY_NBR_PFX | POSTAL_DELIVERY_NBR_SFX | HOUSE_NBR_1 | HOUSE_NBR_SFX_1 | HOUSE_NBR_2 | HOUSE_NBR_SFX_2

SendRight Status Codes and Output Fields

ESR0

SendRight output is not available for this address.

ESR1

SendRight output is provided.

Output Fields

ADDRESS_TYPE | SOA_RECORD_IGNORED | DELIVERY_POINT_ID | DELIVERY_SERVICE_TYPE | DELIVERY_SERVICE_NUMBER | DELIVERY_SERVICE_LOCALITY | HOUSE_NUMBER | HOUSE_NUMBER_ALPHA | RD_NUMBER | HYGIENE | VALIDITY_CODE

SERP Status Codes and Output Fields

ESE0

SERP output is not available for the address.

ESE1

SERP output is provided for the address.

Output Fields

CATEGORY | EXCLUDED_FLAG

SNA Status Codes and Output Fields

ESN0

SNA output not available for the address.

ESN1

SNA output is provided for the address.

Output Fields

CATEGORY

Additional Country-Specific Information in Validated Output

Informatica Address Verification can provide additional information as enrichments to addresses from several countries. If you request country-specific information as an enrichment to an address set, Informatica Address Verification returns the information in the address response.

Address Verification returns status codes that describe the status of the enrichment data in the address response.

The status codes contain four characters in the following format:

- E. Indicates enrichment.
- The two-letter country code for the country.
- The number 1 or 0.

For example, the value EAT1 indicates that Address Verification returns enrichment data for Austria.

The following table shows the country-specific enrichment status and possible return values.

Enrichment	Status	Possible Return Values
SUPPLEMENTARY_AT	EATx	POSTAL_ADDRESS_CODE
SUPPLEMENTARY_BE	EBEx	NIS_CODE

Enrichment	Status	Possible Return Values
SUPPLEMENTARY_BR	EBRx	IBGE_CODE
SUPPLEMENTARY_CH	ECHx	POSTAL_CODE_EXTENSION
SUPPLEMENTARY_DE	EDEx	LOCALITY_ID | STREET_ID | AGS | STREET_CODE
SUPPLEMENTARY_ES	EESx	INE_CODE
SUPPLEMENTARY_FR	EFRx	INSEE_CODE | INSEE_9_CODE
SUPPLEMENTARY_GB	EGBx	DELIVERY_POINT_SUFFIX | UDPRN | UPRN | ADDRESS_KEY | DELIVERY_POINT_TYPE | ORGANIZATION_KEY
SUPPLEMENTARY_IT	EITx	ISTAT_CODE
SUPPLEMENTARY_JP	EJPx	CHOUMEI_AZA_CODE | NEW_CHOUMEI_AZA_CODE | GAIKU_CODE
SUPPLEMENTARY_KR	EKRx	ADDRESS_ID
SUPPLEMENTARY_PL	EPLx	GMINA_CODE | LOCALITY_TERYT_ID | STREET_TERYT_ID
SUPPLEMENTARY_RU	ERUx	FIAS_ID
SUPPLEMENTARY_SR	ERSx	POSTAL_ADDRESS_CODE
SUPPLEMENTARY_US	EUSx	COUNTY_FIPS_CODE | STATE_FIPS_CODE | MSA_ID | CBSA_ID |FINANCE_NUMBER| RECORD_TYPE | CMSA_ID | TIME_ZONE_CODE | TIME_ZONE_NAME | CENSUS_TRACT_NO | CENSUS_BLOCK_GROUP | CENSUS_BLOCK_NO
SUPPLEMENTARY_ZA	EZAx	NAD_ID

Consumer Segmentation (CAMEO) Information in Validated Output

Informatica Address Verification provides consumer segmentation (CAMEO) data as an enrichment to addresses. If you have specified CAMEO enrichment in the request, Informatica Address Verification returns the CAMEO information associated with the address in its response.

If you request CAMEO data in interactive mode, Informatica Address Verification does not provide any CAMEO data for the suggestions that are returned. However, in its response, Informatica Address Verification includes a `JobToken` that you can use to resubmit the address that you have selected from the suggestions. When you resubmit the address along with the `JobToken`, Informatica Address Verification returns the CAMEO data for the selected address.

When you request CAMEO enrichment, the `PayloadXML` field in the response contains `<CATEGORY>`, `<GROUP>`, `<INTERNATIONAL>`, and `< MVID>` elements apart from the `<Status>` element. Informatica Address Verification also provides additional information for these elements in the corresponding description elements such as `<CATEGORY_DESCRIPTION>`, `<GROUP_DESCRIPTION>`, and `<INTERNATIONAL_DESCRIPTION>`.

The following table lists the CAMEO status codes:

Status	Description
ECON	CAMEO Consumer Segmentation codes are not available for the country
ECO0	CAMEO Consumer Segmentation codes are not available for the address
ECO1	CAMEO Consumer Segmentation codes have been appended to the address

GeoCoding Data in Validated Output

You can obtain the geocoordinates for an address as an enrichment in the validated address output.

If you request geocoding in the interactive mode, Informatica Address Verification does not provide any geocoordinates for the suggestions that are returned. However, in its response, Informatica Address Verification includes a `JobToken` that you can use to resubmit the address that you have selected from among the suggestions. When you resubmit the address along with the `JobToken`, Informatica Address Verification returns the geocoordinates for the selected address,

When you request geocoordinates as an enrichment, the `PayloadXml` in the response contains `<LATITUDE>`, `<LONGITUDE>`, and `<LAT_LONG_UNIT>` elements in addition to the `<Status>` element. The `LAT_LONG_UNIT` denotes the projection used; for example, WGS84 for World Geodetic System 1984.

The following table describes the possible status values for the GEOCODING enrichment.

Status	Description
EGC0	Geocoordinates could not be appended to this address
EGC4	Geocoordinates are accurate to partial postal code level
EGC5	Geocoordinates are accurate to postal code level
EGC6	Geocoordinates are accurate to locality level
EGC7	Geocoordinates are accurate to street level
EGC8	Geocoordinates are accurate to house number level (interpolated)
EGC9	High-precision arrival point geocoordinates (point addressing)
EGCA	High-precision parcel centroid geocoordinates
EGCB	High-precision rooftop geocoordinates

Results

For successful transactions, Informatica Address Verification returns the Results object, which is an array of multiple elements such as the Process Status code, validated address, and CountryISO3. Informatica Address Verification returns one result object for each Address object in the addresses array of the request.

ResultDataSet

The `ResultDataSet` array contains various `ResultData` objects. Based on the `ProcessMode`, `ResultDataSet` can contain corrections for the input address or multiple address suggestions. If it is a response to a batch mode or certified mode, the `ResultDataSet` array might contain up to 1000 validated addresses.

ResultData

A ResultData object represents one suggestion or a corrected result, depending on the `ProcessMode` and `ProcessStatus`. It also might contain the following information about the processing and the corrected address or suggestions in the form of an address object.

MailabilityScore

Indicates the possibility of successful delivery to the address on the following 6-point scale:

5. Completely confident of delivery.

4. Almost certain.

3. A good chance of delivery.

2. A fair chance of delivery.

1. Very low chance of delivery.

0. No chance of delivery.

ResultPercentage

Indicates how similar is the corrected result to the input address. Values close to 100% imply high similarity. The `ResultPercentage` value enables you to filter out extensively corrected addresses, especially when the data quality requirements are very high. You can also use the `ResultPercentage` value to determine which of the interactive mode results show the least deviation from the input.

Element Status and Relevance values

`ElementStatus` values give a detailed explanation of the outcome of a validation operation. `ElementStatus` values are relevant only when a validation operation has been performed. However, Informatica Address Verification returns some values for `ElementInputStatus` even after a parsing operation. `ElementInputStatus` and `ElementResultStatus` might contain status information for up to 20 address elements.

`ElementInputStatus` provides per-element status on the matching of input elements with the reference data. `ElementResultStatus` provides per-element information about any corrections or modifications done to the input elements while generating the validated output. The following list shows element positions from left to right:

1. PostalCode level 0
2. PostalCode level 1 (for example, ZIP+4 - Plus 4 addition)
3. Locality level 0
4. Locality level 1 (for example, Urbanization, Dependent Locality)
5. Province level 0
6. Province level 1 (for example, Sub Province)
7. Street level 0
8. Street level 1 (for example, Dependent street)
9. HouseNumber level 0
10. HouseNumber level 1
11. Delivery service level 0 (for example, PO Box, GPO, Packstation, Private Bags)

12. Delivery service level 1
13. Building level 0
14. Building level 1
15. SubBuilding level 0
16. SubBuilding level 1
17. Organization level 0
18. Organization level 1
19. Country level 0 (Mother country)
20. Country level 1 (for example, Territory)

Note: Level 0 in this list pertains to the Item 1 status information, whereas level 1 summarizes the status information for Items 2-6.

ElementInputStatus

Each character in the `ElementInputStatus` describes one address element as explained in the preceding bullet. The possible values are:

0. The input address contains no data at this position.

1. The data at this position cannot be found in the reference data.

2. The position cannot be checked because reference data is missing.

3. The data is incorrect. The reference database suggests that the Number or DeliveryService value is outside the range expected by the reference data. In batch and certified modes, the input data at this position is passed uncorrected as output. In suggestion list modes, Informatica Address Verification can provide alternatives.

4. The data at this position matches the reference data, but with errors.

5. The data at this position matches the reference data, but the data element was corrected or standardized.

6. The data at this position matches the reference data without any error.

ElementResultStatus

Each character in the `ElementResultStatus` maps to one address element as explained earlier in this topic. The possible values for the first 18 positions (address elements without country) are:

0. The output address contains no data at this position.

1. The data at this position cannot be found in the reference data. The input data is copied to the output data.

2. Data at this position is not checked but is standardized.

3. Data at this position is checked but does not match the expected reference data. The reference data suggests that the number data is not in the valid range. The input data is copied to the output. The status value applies in batch mode only.

4. Data at this position is copied from the input because the corresponding reference data is not available.

5. Data at this position is validated but not changed because multiple matches exist in the reference data. The status value applies in batch mode only.

6. Data validation deleted the input value at this position.

7. Data at this position is validated but contained a spelling error. Validation corrected the error by copying the value from the reference data.

8. Data at this position is validated and updated by adding a value from the reference data. It can also mean that the reference database contains additional data for the input element. For example, validation can add a building or sub-building number if a perfect match is found for the street name or building name.

9. Data at this position is validated but not changed, and the delivery status is not clear. For example, the DPV value is wrong.

C. Data at this position is validated and verified, but the name data is out of date. Validation changed the name data.

D. Data at this position is validated and verified but changed from an exonym to an official name.

E. Data at this position is validated and verified. However, data validation standardized the character case or the language. Address validation can change the language if the value fully matches a language alternative. For example, address validation can change "Brussels" to "Bruxelles" in a Belgian address.

F. Data at this position is validated, verified, and not changed, due to a perfect match with reference data.

Positions 19 and 20 in the output string relate to country data. The following values are possible for positions 19 and 20.

0. The output address contains no data at this position.

1. The country is not recognized.

4. The country is recognized from the DefaultCountryISO3 setting.

5. The country is not recognized because multiple matches are available.

6. The country is recognized from a script.

7. The country is recognized from the address format.

8. The country is recognized from major town data.

9. The country is recognized from province data.

C. The country is recognized from territory data.

D. The country is recognized from the country name, but the name contains errors.

E. The country is recognized from the country name without errors.

F. The country is recognized from the ForceCountryISO3 setting.

ElementRelevance

In addition to the element status values described in the preceding bullets, Informatica Address Verification provides you information about the relevance of various address elements with regard to the postal guidelines of the region. The possible values for each address element are 1 for relevant and 0 for not relevant. For any given address, all address elements with a value of 1 must be present for an output address to be deemed valid by the local postal authority. `ElementRelevance` may vary from address to address for countries with different address types, for example, rural verus metropolitan addressing. Furthermore, `AddressElements` that have actually been validated against reference data with `ElementResultStatus` 7 and higher may override the default `ElementRelevance` value defined for that `AddressElement`.

Note that that `ElementRelevance` is meaningful only for addresses with the `ProcessStatus` value of C or V in batch and I3 and I4 for INTERACTIVE requests.

ExtElementStatus

The Extended Element Result Status or EERS code is a twenty-character output field for all valid or corrected addresses. The EERS indicates that additional information might be available in the reference database for the given address. The following EERS codes are available:

1. Data available for the element in the database, but not used for validation.

2. Element unchecked, but changed because of wrong syntax/format.

3. Numeric in element correct, but element changed because of wrong syntax/descriptor.

4. Element correct or unchecked, but moved because of wrong format.

5. Alternative available in database – for example, language, preferred locality name, alias name.

6. Unvalidated parts inside element like additional information.

7. Level change like moving HNO1 to HNO2 or swapping Locality2 with Locality1.

8. Type change for fielded input only; for example, moving SubBuilding to Building Level 2.

9. General Postal Authority Rule.

A. Dominant match for dual address processing.

B. Relevance is only a country-wide default and cannot be trusted.

C. Fast Completion Overflow.

D. Numeric for range expansion (interpolated).

E. Language not available for the country, database language returned.

F. Output address is outdated.

For all other scenarios, Informatica Address Verification returns an EERS value of zero.

AddressResolutionCode

The Address Resolution Code or ARC is a twenty-character output field for all invalid records. The ARC explains why an address is rejected. The following ARC codes are available:

2. Missing element in address.

3. Numeric provided inside element is outside permissible range – for example, wrong numeric inside street name or house number; 100 Main St when house numbers range from 400-800.

4. Multiple inputs for the element.

5. Input element ambiguous / multiple matches.

6. Element contradicts other elements. For example, the postal code information and locality information do not match.

7. 3 strike rule/too many corrections in combination of several elements.

8. General Postal Authority Rule.

For all other scenarios, Informatica Address Verification returns an ARC value of zero.

LanguageISO3

The LanguageISO3 field contains the ISO 639 3-letter code of the language in which Informatica Address Verification returns the result. For example, `DEU` for German.

For transliterated output, Informatica Address Verification specifies the original language in the output. For example, `JPN` in case of romanized Japanese output.

If the language information is not available, Informatica Address Verification returns a value of `XXX` in the LanguageISO3 code field.

Address

The `Address` object contains the address elements of a result. If there were multiple items, such as Locality and Dependent Locality, they are stored in a sequence of the same element.

```
<Locality>
    <string>Locality</string>
```

```
        <string>DependentLocality</string>
    </Locality>
```

In VB.NET this is represented as an Array. Note that Informatica Address Verification does not include the empty elements in the XML response.

The `Address` object also contains a `Residue` element that contains information that was part of the input but is not required for mail delivery in the country to which the address belongs. The `Residue` element can contain multiple strings of data that is superfluous to the validated output. You can set the `MaxItemCount` value for the `Residue` element to specify the maximum number of strings to include in the `Residue` element.

Address Types

You can use the AddressType information in the result output to identify the type of mail box an address points to.

For United States addresses, Informatica Address Verification returns the address type values that the United States Postal Service specifies. The United States Postal Service includes a Record Type value in the reference data for domestic addresses.

New Zealand Posts provides a rural address identifier. Canada Post provides a large volume receiver identifier.

If the reference data does not contain a formal address type designator, Address Verification uses different data elements to assign address types to addresses. Address Verification assign address type values to addresses from countries that do not define address types based on criteria that Address Verification defined.

Address Verification employs a range of criteria to decide the address type for the following countries:

- Australia
- Canada
- France
- New Zealand

Address Verification has a separate set of criteria for identifying the address type when verifying the United States addresses in the Fast Completion mode.

Address Types from the United States Postal Service

Informatica Address Verification returns the United States Postal Service address type for a United States address when you perform verification in Batch, Certified, or Interactive mode.

The United States Postal Service provides the following address types for United States addresses:

- F. The address identifies an organization.
- G. The address is a general delivery address. In a general delivery address, the postal code and the recipient data identify the address.
- H. The address identifies a high-rise building. The address contains subbuilding elements such as apartment or suite.
- P. The address identifies a Post Office Box or a delivery service.
- R. The address is a rural route/highway contract address.
- S. The address identifies a street.
- U. Unidentified. The address is not valid, and Address Verification does not assign an address type.

Address Type Indicators for the United States Addresses in Fast Completion Mode

When verifiying addresses in the fast completion mode, Informatica Address Verification returns the following address type values:

- B. The address identifies a building.
- F. The address identifies an organization.
- L. The address post code identifies the organization as a large volume receiver. The reference data adds or verifies the organization name.
  Address Verification can determine that the address is a large volume receiver in one of the following ways:
  - The address post code identifies the organization as a large volume receiver.
  - The reference data does not contain street or building information.
- P. The address identifies a Post Office Box or a delivery service.
- S. The address identifies a street. S is the default address type. If Address Verification cannot determine the address type from the address data, it returns the default value.
- U. Unidentified. The address is not valid, and Address Verification does not assign an address type.

If an address meets the criteria for more than one address type, Address Verification assigns the first applicable address type from the following list:

L, F, P, B, S

Address Type Indicators for Australia Addresses

Informatica Address Verification returns the following address type values for Australia addresses:

- B. The address identifies a building.
- F. The address identifies an organization.
- L. The address post code identifies the organization as a large volume receiver. The reference data adds or validates the organization name.
  Address Verification can determine that the address is a large volume receiver in one of the following ways:
  - The address post code identifies the organization as a large volume receiver.
  - The reference data does not contain street or building information.
- P. The address identifies a Post Office Box or a delivery service.
- S. The address identifies a street. S is the default address type. If Address Verification cannot determine the address type from the address data, it returns the default value.
- U. Unidentified. The address is not valid, and Address Verification does not assign an address type.

If an address meets the criteria for more than one address type, Address Verification assigns the first applicable address type from the following list:

L, F, P, B, S

Address Type Indicators for Canada Addresses

Informatica Address Verification returns the following address types for Canada addresses:

- B. The address identifies a building.

- F. The address identifies an organization. In Canada addresses, the type F addresses are a subset of the type L addresses. Therefore, the address type F also indicates a large volume receiver.
- G. The address is a general delivery address. In a general delivery address, the postal code and the recipient data identify the address. Address Verification uses the delivery record in the reference data to identify the address type.
- L. The address post code identifies the organization as a large volume receiver. The address might or might not contain an organization name.
- P. The address identifies a Post Office Box or a delivery service.
- R. The address identifies a rural route. Address Verification uses the delivery record in the reference data to identify the address type.
- S. The address identifies a street. S is the default address type. If Address Verification cannot determine the address type from the address data, it returns the default value.
- U. Unidentified. The address is not valid, and Address Verification does not assign an address type.

If an address meets the criteria for more than one address type, Address Verification assigns the first applicable address type from the following list:

F, L, P, B, R, S, G.

Address Type Indicators for France Addresses

Informatica Address Verification returns the following address type values for France addresses:

- B. The address identifies a building.
- F. The address identifies an organization. The address does not include a CEDEX post code.
- G. The address is a general delivery address. The reference data does not contain a match for the street information. However, the reference data contains a match for the CEDEX postal code in the address.
- L. The postal code identifies the organization as a large volume receiver. The address might or might not contain an organization name. The reference data uses the CEDEX postal code to add or validate the organization name.
- P. The address identifies a Post Office Box or a delivery service.
- S. The address identifies a street address. S is the default address type. If Address Verification returns S as the default value if Address Verification cannot determine the address type from the address data.
- U. Unidentified. The address is not valid, and Address Verification does not assign an address type.

If an address meets the criteria for more than one address type, Address Verification assigns the first applicable address type from the following list:

L, F, P, B, S, G.

Address Type Indicators for New Zealand Addresses

Informatica Address Verification returns the following address type values for New Zealand addresses:

- B. The address identifies a building.
- F. The address identifies an organization.
- L. The address post code identifies the organization as a large volume receiver. The reference data adds or validates the organization name.

Address Verification can determine that the address is a large volume receiver in one of the following ways:

- The address post code identifies the organization as a large volume receiver.
- The reference data does not contain street or building information.

- P. The address identifies a Post Office Box or a delivery service.
- R. The address identifies a rural route. Address Verification uses the delivery record in the reference data to identify the address type.
- S. The address identifies a street. S is the default address type. If Address Verification cannot determine the address type from the address data, it returns the default value.
- U. Unidentified. The address is not valid, and Address Verification does not assign an address type.

If an address meets the criteria for more than one address type, Address Verification assigns the first applicable address type from the following list:

L, F, P, B, R, S

Address Type Indicators for Addresses from the Rest of the World

Informatica Address Verification returns the following address type values for addresses from countries other than Australia, Canada, France, New Zealand, and the United States:

- B. The address identifies a building.
- F. The address identifies an organization.
- L. The address post code identifies the organization as a large volume receiver. The reference data adds or validates the organization name.
  Address Verification can determine that the address is a large volume receiver in one of the following ways:
  - The address post code identifies the organization as a large volume receiver.
  - The reference data does not contain street or building information.
- P. The address identifies a Post Office Box or a delivery service.
- S. The address identifies a street. S is the default address type. If Address Verification cannot determine the address type from the address data, it returns the default value.
- U. Unidentified. The address is not valid, and Address Verification does not assign an address type.

If an address meets the criteria for more than one address type, Address Verification assigns the first applicable address type from the following list:

L, F, P, B, S

Note: Address Verification can return information relevant to the address type on other output elements. Consult the Process Status, Element Input Status, and Element Result Status values.

Address Resolution Code Example

When you validate a Canada address without sub-building information in certified mode, Informatica Address Verification returns an ARC value of 2 to indicate that the sub-building information is missing.

Input Address

```
1555 Finch Avenue E
North York ON M2J4X9
```

Output Address

```
1555 Finch Avenue E
North York ON M2J 4X9
```

Address Verification returns a process status value of I3 for the address, indicating that the address is rejected.

The ARC value in the following table indicates the reason for the rejection:

	EIS	**ERS**	**ARC**	**Relevance**	**Explanation**
Sub-building	0	0	2	0	Sub-building missing

EERS Example

In Hong Kong (HKG), it is possible that one or more buildings are available for a given street and house number combination. In this example, the address does not contain a building in the input, but several buildings exist in the reference data for the given street and house number combination.

Input Address

```
1 Harbour Rd.
Wan Chai
Hong Kong Island
```

Output Address

```
1 Harbour Rd.
Wan Chai
Hong Kong Island
```

Process status code is V4, which indicates that the addresses are a perfect match.

	EIS	ERS	EERS	Relevance	Explanation
Building	0	0	1	0	Several building records available for the given Street/HNO

Several building records exist for this address. To avoid false positives, Informatica Address Verification does not appended any building information to the output. The EERS code indicates that building information is available in the reference database, but was not used for validation.

CHAPTER 5

Examples

This chapter includes the following topics:

Example: A V4 Address with GeoCoding

The following example shows a Germany address that Informatica Address Verification validates with a V4 status. The example also shows the following settings:

- `Enrichment` type is set to GEOCODING, and `GeoCodingType` to NONE.
- `GlobalPreferredDescriptor` is set to LONG.
- `PreferredLanguage` is set to ENGLISH.
- `PreferredScript` is set to LATIN.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:v1="http://validator5.AddressDoctor.com/Webservice5/v3">
    <soapenv:Header/>
    <soapenv:Body>
        <Process xmlns="http://validator5.AddressDoctor.com/Webservice5/v3">
            <login>*****</login>
            <password>******</password>
            <parameters>
                <ProcessMode>INTERACTIVE</ProcessMode>
                <ValidationParameters>
                    <PreferredLanguage>ENGLISH</PreferredLanguage>
                    <PreferredScript>LATIN</PreferredScript>
                    <GlobalPreferredDescriptor>LONG</GlobalPreferredDescriptor>
                    <GeoCodingType>NONE</GeoCodingType>
                </ValidationParameters>
            </parameters>
            <addresses>
                <Address>
                    <FormattedAddress>
                        <string>Röntgenstr. 9</string>
                        <string>67133 Maxdorf</string>
                        <string>DEU</string>
                    </FormattedAddress>
                </Address>
            </addresses>
            <enrichments>
                <Enrichment>
                    <Type>GEOCODING</Type>
                </Enrichment>
            </enrichments>
        </Process>
    </soapenv:Body>
</soapenv:Envelope>
```

Informatica Address Verification returns the following response:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/
XMLSchema">
   <soap:Body>
      <ProcessResponse xmlns="http://validator5.AddressDoctor.com/Webservice5/v3">
         <ProcessResult>
            <StatusCode>100</StatusCode>
            <StatusMessage>OK</StatusMessage>
            <Results>
               <Result>
                  <ProcessStatus>V4</ProcessStatus>
                  <CountryISO3>DEU</CountryISO3>
                  <ResultDataSet>
                     <ResultData>
                        <ResultNumber>1</ResultNumber>
                        <MailabilityScore>5</MailabilityScore>
                        <ResultPercentage>100.00</ResultPercentage>
                        <ElementInputStatus>60600060600000000060</ElementInputStatus>
                        <ElementResultStatus>F0F080F0F000000000E0</ElementResultStatus>
                        <ElementRelevance>10100010100000000010</ElementRelevance>
                        <ExtElementStatus>00E000E0000000000000</ExtElementStatus>
                        <AddressResolutionCode>00000000000000000000
                        </AddressResolutionCode>
                        <AddressType>S</AddressType>
                        <LanguageISO3>ENG</LanguageISO3>
                        <Address>
                           <Street>
                              <string>Roentgenstrasse</string>
                           </Street>
                           <HouseNumber>
                              <string>9</string>
                           </HouseNumber>
                           <Locality>
                              <string>Maxdorf</string>
                           </Locality>
                           <PostalCode>
                              <string>67133</string>
                           </PostalCode>
                           <Province>
                              <string>Rhineland-Palatinate</string>
                           </Province>
                           <Country>
                              <string>GERMANY</string>
                           </Country>
                           <DeliveryAddressLines>
                              <string>Roentgenstrasse 9</string>
                           </DeliveryAddressLines>
                           <CountrySpecificLocalityLine>
                              <string>67133 Maxdorf</string>
                           </CountrySpecificLocalityLine>
                           <FormattedAddress>
```

```
                              <string>Roentgenstrasse 9</string>
                              <string>67133 Maxdorf</string>
                              <string>GERMANY</string>
                           </FormattedAddress>
                           <AddressComplete>Roentgenstrasse 9;67133 Maxdorf;GERMANY
                           </AddressComplete>
                        </Address>
                     </ResultData>
                  </ResultDataSet>
                  <Enrichments>
                     <Enrichment>
                        <Type>GEOCODING</Type>
                        <PayloadXml><![CDATA[<GEOCODING><Status>EGC8</
Status><COMPLETE>49.490461, 8.290109 WGS84</COMPLETE><LATITUDE>49.490461</
LATITUDE><LONGITUDE>8.290109</LONGITUDE><LAT_LONG_UNIT>WGS84</LAT_LONG_UNIT></
GEOCODING>]]></PayloadXml>
                     </Enrichment>
                  </Enrichments>
               </Result>
            </Results>
         </ProcessResult>
      </ProcessResponse>
   </soap:Body>
</soap:Envelope>
```

Note the following points in the response:

- The output is in English using LATIN script.
- Str in the input has been expanded to its longer form: strasse.
  Röntgenstr has been converted to Roentgenstrasse.
- The output contains interpolated geocoordinates as specified by the `GeoCodingType` **NONE**.
- Address elements in the `AddressComplete` field are separated by semicolon, which is the default format delimiter.

Related Topics:

- “GeoCodingType” on page 44
- “Enrichments Parameter” on page 56
- “Example: A United Kingdom Address with Building-Level Rooftop GeoCoding” on page 92

Example: An Austria Address with Country-Specific Enrichment

The following example shows an Informatica Address Verification request to validate an Austria address and retrieving the PAC information for the address.

The following validation parameters are set in this request:

- `FormatDelimiter` is set to COMMA.
- `PreferredScript` is set to LATIN.
- `PreferredLanguage` is set to ENGLISH.
- `MaxResultCount` is set to 100.
- Under `Standardizations` for LOCALITY, `Casing` is set to UPPER.

- Type of `Enrichment` is set to SUPPLEMENTARY_AT.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:v1="http://validator5.AddressDoctor.com/Webservice5/v3">
    <soapenv:Header/>
    <soapenv:Body>
        <Process xmlns="http://validator5.AddressDoctor.com/Webservice5/v3">
            <login>********</login>
            <password>*******</password>
            <parameters>
                <ProcessMode>INTERACTIVE</ProcessMode>
                <ValidationParameters>
                    <FormatDelimiter>COMMA</FormatDelimiter>
                    <PreferredScript>LATIN</PreferredScript>
                    <PreferredLanguage>ENGLISH</PreferredLanguage>
                    <MaxResultCount>100</MaxResultCount>
                    <Standardizations>
                        <Element>LOCALITY</Element>
                            <Casing>UPPER</Casing>
                            <MaxLength>50</MaxLength>
                            <MaxItemCount>2</MaxItemCount>
                    </Standardizations>
                </ValidationParameters>
            </parameters>
            <addresses>
                <Address>
                    <FormattedAddress>
                        <string>Grafenweg 3</string>
                        <string>6714 Nüziders</string>
                        <string>AUT</string>
                    </FormattedAddress>
                </Address>
            </addresses>
            <enrichments>
                <Enrichment>
                    <Type>SUPPLEMENTARY_AT</Type>
                </Enrichment>
            </enrichments>
        </Process>
    </soapenv:Body>
</soapenv:Envelope>
```

Note the following points in the response:

- The elements in `AddressComplete` are separated by commas.
- The Locality element is returned in all capital letters because of the element standardization setting in the request.
- The output is in English language using the Latin script.
- The PAC code for the address.

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/
XMLSchema">
   <soap:Body>
      <ProcessResponse xmlns="http://validator5.AddressDoctor.com/Webservice5/v3">
         <ProcessResult>
            <StatusCode>100</StatusCode>
            <StatusMessage>OK</StatusMessage>
            <Results>
               <Result>
                  <ProcessStatus>V4</ProcessStatus>
                  <CountryISO3>AUT</CountryISO3>
                  <ResultDataSet>
                     <ResultData>
                        <ResultNumber>1</ResultNumber>
                        <MailabilityScore>5</MailabilityScore>
                        <ResultPercentage>100.00</ResultPercentage>
                        <ElementInputStatus>60600060600000000060</ElementInputStatus>
                        <ElementResultStatus>F0F080F0F000000000E0</ElementResultStatus>
                        <ElementRelevance>10100010100000000010</ElementRelevance>
                        <ExtElementStatus>00E0E0E0000000000000</ExtElementStatus>
                        <AddressResolutionCode>00000000000000000000
                        </AddressResolutionCode>
                        <AddressType>S</AddressType>
                        <LanguageISO3>DEU</LanguageISO3>
                        <Address>
                           <Street>
                              <string>Grafenweg</string>
                           </Street>
                           <HouseNumber>
                              <string>3</string>
                           </HouseNumber>
                           <Locality>
                              <string>NUEZIDERS</string>
                           </Locality>
                           <PostalCode>
                              <string>6714</string>
                           </PostalCode>
                           <Province>
                              <string>Vorarlberg</string>
                           </Province>
                           <Country>
                              <string>AUSTRIA</string>
                           </Country>
                           <DeliveryAddressLines>
                              <string>Grafenweg 3</string>
                           </DeliveryAddressLines>
                           <CountrySpecificLocalityLine>
                              <string>6714 Nueziders</string>
                           </CountrySpecificLocalityLine>
                           <FormattedAddress>
                              <string>Grafenweg 3</string>
                              <string>6714 Nueziders</string>
                              <string>AUSTRIA</string>
                           </FormattedAddress>
                           <AddressComplete>Grafenweg 3,6714 Nueziders,AUSTRIA
                           </AddressComplete>
                        </Address>
                     </ResultData>
                  </ResultDataSet>
                  <Enrichments>
                     <Enrichment>
                        <Type>SUPPLEMENTARY_AT</Type>
                        <PayloadXml><![CDATA[<SUPPLEMENTARY_AT><Status>EAT1
                        </Status><POSTAL_ADDRESS_CODE>100501606
                        </POSTAL_ADDRESS_CODE></SUPPLEMENTARY_AT>]]></PayloadXml>
                     </Enrichment>
                  </Enrichments>
               </Result>
            </Results>
         </ProcessResult>
      </ProcessResponse>
   </soap:Body>
</soap:Envelope>
```

Example: Single Line Address Validation

The following example shows an Australia addresses validated using the single line address validation feature. Note the following settings in the input:

- `ProcessMode` is set to FASTCOMPLETION.
- Address is entered in the `AddressComplete` field.

- `MaxResultCount` is set to 5.
- No `FormatDelimiter` is specified.
- `GlobalPreferredDescriptor` is set to SHORT.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:v1="http://validator5.AddressDoctor.com/Webservice5/v3">
    <soapenv:Header/>
    <soapenv:Body>
        <Process xmlns="http://validator5.AddressDoctor.com/Webservice5/v3">
            <login>******</login>
            <password>*******</password>
            <parameters>
                <ProcessMode>FASTCOMPLETION</ProcessMode>
                <ValidationParameters>
                    <MaxResultCount>5</MaxResultCount>
                    <GlobalPreferredDescriptor>SHORT</GlobalPreferredDescriptor>
                </ValidationParameters>
            </parameters>
            <addresses>
                <Address>
                    <Country>
                        <string>AUS</string>
                    </Country>
                    <AddressComplete>255 George Street 2000 Sydney</AddressComplete>
                </Address>
            </addresses>
        </Process>
    </soapenv:Body>
</soapenv:Envelope>
```

In the response, note the following:

- Only five suggestions are returned because of the `MaxResultCount` setting.
- Address elements in the `AddressComplete` field of the output are separated by semicolon, which is the default `FormatDelimiter`.
- Element descriptors use the short form. For example, ST for Street.

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/
XMLSchema">
   <soap:Body>
      <ProcessResponse xmlns="http://validator5.AddressDoctor.com/Webservice5/v3">
         <ProcessResult>
            <StatusCode>100</StatusCode>
            <StatusMessage>OK</StatusMessage>
            <JobToken>4ab1d710-a854-4d88-a321-202014996462</JobToken>
            <Results>
               <Result>
                  <ProcessStatus>Q3</ProcessStatus>
                  <CountryISO3>AUS</CountryISO3>
                  <ResultDataSet>
                     <ResultData>
                        <ResultNumber>1</ResultNumber>
                        <MailabilityScore>2</MailabilityScore>
                        <ResultPercentage>100.00</ResultPercentage>
                        <ElementInputStatus>60600050600000000060</ElementInputStatus>
                        <ElementResultStatus>F0F080E0F000800000E0</ElementResultStatus>
                        <ElementRelevance>10101010100000000010</ElementRelevance>
                        <ExtElementStatus>00000000000000100000</ExtElementStatus>
                        <AddressResolutionCode>00000000000000000000
                        </AddressResolutionCode>
                        <AddressType>S</AddressType>
                        <LanguageISO3>ENG</LanguageISO3>
                        <Address>
                           <Building>
                              <string>NATIONAL AUSTRALIA BANK HOUSE</string>
                           </Building>
                           <Street>
                              <string>GEORGE ST</string>
                           </Street>
                           <HouseNumber>
                              <string>255</string>
                           </HouseNumber>
                           <Locality>
                              <string>SYDNEY</string>
                           </Locality>
                           <PostalCode>
                              <string>2000</string>
                           </PostalCode>
                           <Province>
                              <string>NSW</string>
                           </Province>
                           <Country>
                              <string>AUSTRALIA</string>
                           </Country>
                           <DeliveryAddressLines>
                              <string>NATIONAL AUSTRALIA BANK HOUSE</string>
                              <string>255 GEORGE ST</string>
                           </DeliveryAddressLines>
                           <CountrySpecificLocalityLine>
                              <string>SYDNEY  NSW  2000</string>
                           </CountrySpecificLocalityLine>
                           <FormattedAddress>
                              <string>NATIONAL AUSTRALIA BANK HOUSE</string>
                              <string>255 GEORGE ST</string>
                              <string>SYDNEY  NSW  2000</string>
                              <string>AUSTRALIA</string>
                           </FormattedAddress>
                           <AddressComplete>NATIONAL AUSTRALIA BANK HOUSE;255
                            GEORGE ST; SYDNEY  NSW  2000;AUSTRALIA
                            </AddressComplete>
                        </Address>
                     </ResultData>
                     <ResultData>
                        <ResultNumber>2</ResultNumber>
                        <MailabilityScore>2</MailabilityScore>
                        <ResultPercentage>100.00</ResultPercentage>
                        <ElementInputStatus>60600050600000000060</ElementInputStatus>
                        <ElementResultStatus>F0F080E0F000800800E0</ElementResultStatus>
                        <ElementRelevance>10101010100000010010</ElementRelevance>
                        <ExtElementStatus>00000000000000100000</ExtElementStatus>
                        <AddressResolutionCode>00000000000000000000
                        </AddressResolutionCode>
                        <AddressType>S</AddressType>
                        <LanguageISO3>ENG</LanguageISO3>
                        <Address>
                           <Building>
                              <string>NATIONAL AUSTRALIA BANK HOUSE</string>
                           </Building>
                           <SubBuilding>
                              <string/>
                              <string>G</string>
                           </SubBuilding>
                           <Street>
                              <string>GEORGE ST</string>
                           </Street>
                           <HouseNumber>
                              <string>255</string>
                           </HouseNumber>
                           <Locality>
                              <string>SYDNEY</string>
                           </Locality>
                           <PostalCode>
                              <string>2000</string>
                           </PostalCode>
                           <Province>
                              <string>NSW</string>
                           </Province>
                           <Country>
                              <string>AUSTRALIA</string>
                           </Country>
                           <DeliveryAddressLines>
                              <string>NATIONAL AUSTRALIA BANK HOUSE</string>
                              <string>G 255 GEORGE ST</string>
                           </DeliveryAddressLines>
                           <CountrySpecificLocalityLine>
                              <string>SYDNEY  NSW  2000</string>
                           </CountrySpecificLocalityLine>
                           <FormattedAddress>
                              <string>NATIONAL AUSTRALIA BANK HOUSE</string>
                              <string>G 255 GEORGE ST</string>
                              <string>SYDNEY  NSW  2000</string>
                              <string>AUSTRALIA</string>
                           </FormattedAddress>
                           <AddressComplete>NATIONAL AUSTRALIA BANK HOUSE;
                            G 255 GEORGE ST;SYDNEY  NSW  2000;AUSTRALIA
                           </AddressComplete>
                        </Address>
                     </ResultData>
                     <ResultData>
                        <ResultNumber>3</ResultNumber>
                        <MailabilityScore>2</MailabilityScore>
                        <ResultPercentage>100.00</ResultPercentage>
                        <ElementInputStatus>60600050600000000060</ElementInputStatus>
                        <ElementResultStatus>F0F080E0F000800800E0</ElementResultStatus>
                        <ElementRelevance>10101010100000010010</ElementRelevance>
                        <ExtElementStatus>000000000000001C0000</ExtElementStatus>
                        <AddressResolutionCode>00000000000000000000
                        </AddressResolutionCode>
                        <AddressType>S</AddressType>
                        <LanguageISO3>ENG</LanguageISO3>
                        <Address>
                           <Building>
                              <string>NATIONAL AUSTRALIA BANK HOUSE</string>
                           </Building>
                           <SubBuilding>
                              <string/>
                              <string>L 1-30</string>
                           </SubBuilding>
                           <Street>
                              <string>GEORGE ST</string>
                           </Street>
                           <HouseNumber>
                              <string>255</string>
                           </HouseNumber>
                           <Locality>
                              <string>SYDNEY</string>
                           </Locality>
                           <PostalCode>
                              <string>2000</string>
                           </PostalCode>
                           <Province>
                              <string>NSW</string>
                           </Province>
                           <Country>
                              <string>AUSTRALIA</string>
                           </Country>
                           <DeliveryAddressLines>
                              <string>NATIONAL AUSTRALIA BANK HOUSE</string>
                              <string>L 1-30 255 GEORGE ST</string>
                           </DeliveryAddressLines>
                           <CountrySpecificLocalityLine>
                              <string>SYDNEY  NSW  2000</string>
                           </CountrySpecificLocalityLine>
                           <FormattedAddress>
                              <string>NATIONAL AUSTRALIA BANK HOUSE</string>
                              <string>L 1-30 255 GEORGE ST</string>
                              <string>SYDNEY  NSW  2000</string>
                              <string>AUSTRALIA</string>
                           </FormattedAddress>
                           <AddressComplete>NATIONAL AUSTRALIA BANK HOUSE;
                            L 1-30 255 GEORGE ST;SYDNEY  NSW  2000;
                            AUSTRALIA</AddressComplete>
                        </Address>
                     </ResultData>
                     <ResultData>
                        <ResultNumber>4</ResultNumber>
                        <MailabilityScore>2</MailabilityScore>
                        <ResultPercentage>100.00</ResultPercentage>
                        <ElementInputStatus>60600050600000000060</ElementInputStatus>
                        <ElementResultStatus>F0F080E0F000808800E0</ElementResultStatus>
                        <ElementRelevance>10101010100000110010</ElementRelevance>
                        <ExtElementStatus>00000000000000CC0000</ExtElementStatus>
                        <AddressResolutionCode>00000000000000000000
                        </AddressResolutionCode>
                        <AddressType>S</AddressType>
                        <LanguageISO3>ENG</LanguageISO3>
                        <Address>
                           <Building>
                              <string>NATIONAL AUSTRALIA BANK HOUSE</string>
                           </Building>
                           <SubBuilding>
                              <string>SE 1-2</string>
                              <string>L 7-7</string>
                           </SubBuilding>
                           <Street>
                              <string>GEORGE ST</string>
                           </Street>
                           <HouseNumber>
                              <string>255</string>
                           </HouseNumber>
                           <Locality>
                              <string>SYDNEY</string>
                           </Locality>
                           <PostalCode>
                              <string>2000</string>
                           </PostalCode>
                           <Province>
                              <string>NSW</string>
                           </Province>
                           <Country>
                              <string>AUSTRALIA</string>
                           </Country>
                           <DeliveryAddressLines>
                              <string>NATIONAL AUSTRALIA BANK HOUSE</string>
                              <string>SE 1-2 L 7-7</string>
                              <string>255 GEORGE ST</string>
                           </DeliveryAddressLines>
                           <CountrySpecificLocalityLine>
                              <string>SYDNEY  NSW  2000</string>
                           </CountrySpecificLocalityLine>
                           <FormattedAddress>
                              <string>NATIONAL AUSTRALIA BANK HOUSE</string>
                              <string>SE 1-2 L 7-7</string>
                              <string>255 GEORGE ST</string>
                              <string>SYDNEY  NSW  2000</string>
                              <string>AUSTRALIA</string>
                           </FormattedAddress>
                           <AddressComplete>NATIONAL AUSTRALIA BANK HOUSE;
                            SE 1-2 L 7-7;255 GEORGE ST;
                            SYDNEY NSW 2000;AUSTRALIA</AddressComplete>
                        </Address>
                     </ResultData>
                     <ResultData>
                        <ResultNumber>5</ResultNumber>
                        <MailabilityScore>2</MailabilityScore>
                        <ResultPercentage>100.00</ResultPercentage>
                        <ElementInputStatus>60600050600000000060</ElementInputStatus>
                        <ElementResultStatus>F0F080E0F000808800E0</ElementResultStatus>
                        <ElementRelevance>10101010100000110010</ElementRelevance>
```

```
                        <ExtElementStatus>00000000000000CC0000</ExtElementStatus>
                        <AddressResolutionCode>00000000000000000000
                        </AddressResolutionCode>
                        <AddressType>S</AddressType>
                        <LanguageISO3>ENG</LanguageISO3>
                        <Address>
                           <Building>
                              <string>NATIONAL AUSTRALIA BANK HOUSE</string>
                           </Building>
                           <SubBuilding>
                              <string>SE 1-2</string>
                              <string>L 1-6</string>
                           </SubBuilding>
                           <Street>
                              <string>GEORGE ST</string>
                           </Street>
                           <HouseNumber>
                              <string>255</string>
                           </HouseNumber>
                           <Locality>
                              <string>SYDNEY</string>
                           </Locality>
                           <PostalCode>
                              <string>2000</string>
                           </PostalCode>
                           <Province>
                              <string>NSW</string>
                           </Province>
                           <Country>
                              <string>AUSTRALIA</string>
                           </Country>
                           <DeliveryAddressLines>
                              <string>NATIONAL AUSTRALIA BANK HOUSE</string>
                              <string>SE 1-2 L 1-6</string>
                              <string>255 GEORGE ST</string>
                           </DeliveryAddressLines>
                           <CountrySpecificLocalityLine>
                              <string>SYDNEY  NSW  2000</string>
                           </CountrySpecificLocalityLine>
                           <FormattedAddress>
                              <string>NATIONAL AUSTRALIA BANK HOUSE</string>
                              <string>SE 1-2 L 1-6</string>
                              <string>255 GEORGE ST</string>
                              <string>SYDNEY  NSW  2000</string>
                              <string>AUSTRALIA</string>
                           </FormattedAddress>
                           <AddressComplete>NATIONAL AUSTRALIA BANK HOUSE;
                            SE 1-2 L 1-6;255 GEORGE ST;
                            SYDNEY NSW 2000;AUSTRALIA</AddressComplete>
                        </Address>
                     </ResultData>
                  </ResultDataSet>
               </Result>
            </Results>
         </ProcessResult>
      </ProcessResponse>
   </soap:Body>
</soap:Envelope>
```

Related Topics:

- “Single-Line Address Validation” on page 23

Example: AddressCodeLookup

The following example shows how you can retrieve a Japan address by using the Choumei Aza code as the AddressCodeLookup input.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:v1="http://validator5.AddressDoctor.com/Webservice5/v3">
    <soapenv:Header/>
    <soapenv:Body>
        <Process xmlns="http://validator5.AddressDoctor.com/Webservice5/v3">
            <login>******</login>
            <password>******</password>
            <parameters>
                <ProcessMode>ADDRESSCODELOOKUP</ProcessMode>
                <ValidationParameters>
                    <PreferredScript>LATIN</PreferredScript>
                </ValidationParameters>
            </parameters>
            <addresses>
                <Address>
                    <Country>
                        <string>JPN</string>
                    </Country>
                    <AddressCode>
                        <CodeType>JPN_CHOUMEI_AZA_CODE</CodeType>
                        <Value>28201160001</Value>
                    </AddressCode>
                </Address>
            </Addresses>
        </Process>
    </soapenv:Body>
</soapenv:Envelope>
```

The following example shows the response:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/
XMLSchema">
   <soap:Body>
      <ProcessResponse xmlns="http://validator5.AddressDoctor.com/Webservice5/v3">
         <ProcessResult>
            <StatusCode>100</StatusCode>
            <StatusMessage>OK</StatusMessage>
            <Results>
               <Result>
                  <ProcessStatus>A1</ProcessStatus>
                  <CountryISO3>JPN</CountryISO3>
                  <ResultDataSet>
                     <ResultData>
                        <ResultNumber>1</ResultNumber>
                        <MailabilityScore>0</MailabilityScore>
                        <ResultPercentage>100.00</ResultPercentage>
                        <ElementInputStatus>00000000000000000000</ElementInputStatus>
                        <ElementResultStatus>F0FFF0F00000000000E0</ElementResultStatus>
                        <ElementRelevance>10111010000000000010</ElementRelevance>
                        <ExtElementStatus>00000000000000000000</ExtElementStatus>
                        <AddressResolutionCode>00000000000000000000
                        </AddressResolutionCode>
                        <AddressType/>
                        <LanguageISO3/>
                        <Address>
                           <Street>
```

```
                              <string>1 CHŌME</string>
                           </Street>
                           <Locality>
                              <string>HIMEJI-SHI</string>
                              <string/>
                              <string>TADERAHIGA-SHI</string>
                           </Locality>
                           <PostalCode>
                              <string>670-0081</string>
                           </PostalCode>
                           <Province>
                              <string>HYŌGO-KEN</string>
                           </Province>
                           <Country>
                              <string>JAPAN</string>
                           </Country>
                           <DeliveryAddressLines>
                              <string>1 CHŌME</string>
                           </DeliveryAddressLines>
                           <CountrySpecificLocalityLine>
                              <string>HIMEJI-SHI HYŌGO-KEN 670-0081</string>
                           </CountrySpecificLocalityLine>
                           <FormattedAddress>
                              <string>1 CHŌME</string>
                              <string>TADERAHIGA-SHI</string>
                              <string>HIMEJI-SHI HYŌGO-KEN 670-0081</string>
                              <string>JAPAN</string>
                           </FormattedAddress>
                           <AddressComplete>1 CHŌME;TADERAHIGA-SHI;HIMEJI-SHI HYŌGO-KEN
                           670-0081;JAPAN</AddressComplete>
                        </Address>
                     </ResultData>
                  </ResultDataSet>
               </Result>
            </Results>
         </ProcessResult>
      </ProcessResponse>
   </soap:Body>
</soap:Envelope>
```

Related Topics:

- “Address Code Lookup Mode” on page 23

Example: An India Address with Multiple Levels of Locality Information

The following example shows an India address that has multiple levels of locality information. Because the parameters elements are not included in the request, Informatica Address Verification uses the default settings to validate this address.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:v1="http://validator5.AddressDoctor.com/Webservice5/v3">
    <soapenv:Header/>
    <soapenv:Body>
        <Process xmlns="http://validator5.AddressDoctor.com/Webservice5/v3">
            <login>*****</login>
            <password>*****</password>
            <parameters>
                <ProcessMode>INTERACTIVE</ProcessMode>
            </parameters>
            <addresses>
                <Address>
```

```
                    <FormattedAddress>
                        <string>ABC Ltd</string>
                        <string>66 Akurli Road Near Big Bazar </string>
                        <string>Akurli Industrial Premises</string>
                        <string>Kandivali East</string>
                        <string>Mumbai 400101</string>
                        <string>India</string>
                    </FormattedAddress>
                </Address>
            </addresses>
        </Process>
    </soapenv:Body>
</soapenv:Envelope>
```

In the following response you can see that the validated output contains three levels of Locality. The additional information provided along with the street information, Near Big Bazar, is returned in the street element but separated from the street information by a comma.

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/
XMLSchema">
   <soap:Body>
      <ProcessResponse xmlns="http://validator5.AddressDoctor.com/Webservice5/v3">
         <ProcessResult>
            <StatusCode>100</StatusCode>
            <StatusMessage>OK</StatusMessage>
            <Results>
               <Result>
                  <ProcessStatus>V2</ProcessStatus>
                  <CountryISO3>IND</CountryISO3>
                  <ResultDataSet>
                     <ResultData>
                        <ResultNumber>1</ResultNumber>
                        <MailabilityScore>4</MailabilityScore>
                        <ResultPercentage>100.0</ResultPercentage>
                        <ElementInputStatus>60660060200000002060</ElementInputStatus>
                        <ElementResultStatus>F0FF88F04000000040E0</ElementResultStatus>
                        <ElementRelevance>10110010100000001010</ElementRelevance>
                        <ExtElementStatus>00000600000000000000</ExtElementStatus>
                        <AddressResolutionCode>00000000000000000000
                        </AddressResolutionCode>
                        <AddressType>S</AddressType>
                        <LanguageISO3/>
                        <Address>
                           <Organization>
                              <string>ABC Ltd</string>
                           </Organization>
                           <Street>
                              <string>Akurli Road, Near Big Bazar</string>
                           </Street>
                           <HouseNumber>
                              <string>66</string>
                           </HouseNumber>
                           <Locality>
                              <string>Mumbai</string>
                              <string>Kandivali East</string>
                              <string>Akurli Industrial Premises</string>
                           </Locality>
                           <PostalCode>
                              <string>400101</string>
                           </PostalCode>
                           <Province>
                              <string>Maharashtra</string>
                              <string>Mumbai</string>
                           </Province>
                           <Country>
                              <string>INDIA</string>
                           </Country>
                           <RecipientLines>
                              <string>ABC Ltd</string>
                           </RecipientLines>
                           <DeliveryAddressLines>
                              <string>66 Akurli Road, Near Big Bazar</string>
                           </DeliveryAddressLines>
                           <CountrySpecificLocalityLine>
                              <string>Mumbai - 400101</string>
                           </CountrySpecificLocalityLine>
                           <FormattedAddress>
                              <string>ABC Ltd</string>
                              <string>66 Akurli Road, Near Big Bazar</string>
                              <string>Akurli Industrial Premises</string>
                              <string>Kandivali East</string>
                              <string>Mumbai - 400101</string>
                              <string>INDIA</string>
                           </FormattedAddress>
                           <AddressComplete>ABC Ltd;66 Akurli Road, Near Big Bazar;
                            Akurli Industrial Premises;Kandivali East;
                            Mumbai - 400101;INDIA</AddressComplete>
                        </Address>
                     </ResultData>
                  </ResultDataSet>
               </Result>
            </Results>
         </ProcessResult>
      </ProcessResponse>
   </soap:Body>
</soap:Envelope>
```

Example: Address with Residue Information

The following example shows an input address that contains superfluous information that Informatica Address Verification returns in the `Residue` element of the response. The following request contains two main localities, Maxdorf and Mannheim. Because the street information and the postal code belong to the locality Maxdorf, Informatica Address Verification moves the other locality in the input address, Mannheim, to the Residue field in the response.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:v1="http://validator5.AddressDoctor.com/Webservice5/v3">
    <soapenv:Header/>
    <soapenv:Body>
        <Process xmlns="http://validator5.AddressDoctor.com/Webservice5/v3">
            <login>*****</login>
            <password>*****</password>
            <parameters>
                <ProcessMode>INTERACTIVE</ProcessMode>
             </parameters>
            <addresses>
                <Address>
                    <FormattedAddress>
                        <string>Röntgenstr. 9</string>
                        <string>67133 Maxdorf</string>
                        <string>Mannheim</string>
                        <string>DEU</string>
                    </FormattedAddress>
                </Address>
            </addresses>
        </Process>
    </soapenv:Body>
</soapenv:Envelope>
```

In the response, Informatica Address Verification moves Mannheim, which is not required for mail delivery, to the `Residue` field. Note that the information in the `Residue` field is not part of the `FormattedAddress` or `AddressComplete` elements.

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/
XMLSchema">
   <soap:Body>
      <ProcessResponse xmlns="http://validator5.AddressDoctor.com/Webservice5/v3">
         <ProcessResult>
            <StatusCode>100</StatusCode>
            <StatusMessage>OK</StatusMessage>
            <Results>
               <Result>
                  <ProcessStatus>C4</ProcessStatus>
                  <CountryISO3>DEU</CountryISO3>
                  <ResultDataSet>
                     <ResultData>
                        <ResultNumber>1</ResultNumber>
                        <MailabilityScore>4</MailabilityScore>
                        <ResultPercentage>100.00</ResultPercentage>
                        <ElementInputStatus>60600060600000000060</ElementInputStatus>
                        <ElementResultStatus>F0F080F0F000000000E0</ElementResultStatus>
                        <ElementRelevance>10100010100000000010</ElementRelevance>
                        <ExtElementStatus>00005000000000000000</ExtElementStatus>
                        <AddressResolutionCode>00000000000000000000
                        </AddressResolutionCode>
                        <AddressType>S</AddressType>
                        <LanguageISO3>DEU</LanguageISO3>
                        <Address>
                           <Street>
                              <string>Röntgenstr.</string>
                           </Street>
                           <HouseNumber>
                              <string>9</string>
                           </HouseNumber>
                           <Locality>
                              <string>Maxdorf</string>
                           </Locality>
                           <PostalCode>
                              <string>67133</string>
                           </PostalCode>
                           <Province>
                              <string>Rheinland-Pfalz</string>
                           </Province>
                           <Country>
                              <string>GERMANY</string>
                           </Country>
                           <Residue>
                              <string>Mannheim</string>
                           </Residue>
                           <DeliveryAddressLines>
                              <string>Röntgenstr. 9</string>
                           </DeliveryAddressLines>
                           <CountrySpecificLocalityLine>
                              <string>67133 Maxdorf</string>
                           </CountrySpecificLocalityLine>
                           <FormattedAddress>
                              <string>Röntgenstr. 9</string>
                              <string>67133 Maxdorf</string>
                              <string>GERMANY</string>
                           </FormattedAddress>
                           <AddressComplete>Röntgenstr. 9;67133 Maxdorf;GERMANY
                           </AddressComplete>
                        </Address>
                     </ResultData>
                  </ResultDataSet>
               </Result>
            </Results>
         </ProcessResult>
      </ProcessResponse>
   </soap:Body>
</soap:Envelope>
```

Example: A United Kingdom Address with Building-Level Rooftop GeoCoding

The following example shows a United Kingdom address for which Informatica Address Verification returns the rooftop geocoordinates even though the address does not contain a house number.

The request specifies the `GeoCodingType` as ROOFTOP. For United Kingdom addresses, you receive rooftop geocoordinates if you set the geocoding type to rooftop. The request also specifies the `Enrichment Type` as GEOCODING.

```
<soapenv:Envelope   xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:v1="http://validator5.AddressDoctor.com/Webservice5/v3">
<soapenv:Header/>
<soapenv:Body>
<Process   xmlns="http://validator5.AddressDoctor.com/Webservice5/v3">
    <login>*****</login>
    <password>*****</password>
    <parameters>
        <ProcessMode>BATCH</ProcessMode>
        <ValidationParameters>
            <PreferredLangauge>ENGLISH</PreferredLangauge>
            <PreferredScript>ASCII_SIMPLIFIED</PreferredScript>
            <GeoCodingType>ROOFTOP</GeoCodingType>
    </ValidationParameters>
    </parameters>
    <addresses>
        <Address>
            <FormattedAddress>
                <string>REDBOURNE HOUSE</string>
                <string>REDBOURNE AVENUE </string>
                <string>LONDON</string>
                <string>N3 2BT GBR  </string>
            </FormattedAddress>
        </Address>
    </addresses>
    <enrichments>
        <Enrichment>
            <Type>GEOCODING</Type>
        </Enrichment>
    </enrichments>
</Process>
</soapenv:Body></soapenv:Envelope>
```

In the result, Informatica Address Verification returns the rooftop geocoordinates along with the validated output.

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/
XMLSchema">
   <soap:Body>
      <ProcessResponse xmlns="http://validator5.AddressDoctor.com/Webservice5/v3">
         <ProcessResult>
            <StatusCode>100</StatusCode>
            <StatusMessage>OK</StatusMessage>
            <Results>
               <Result>
                  <ProcessStatus>V4</ProcessStatus>
                  <CountryISO3>GBR</CountryISO3>
                  <ResultDataSet>
                     <ResultData>
```

```
                        <ResultNumber>1</ResultNumber>
                        <MailabilityScore>5</MailabilityScore>
                        <ResultPercentage>100.00</ResultPercentage>
                        <ElementInputStatus>60600060000060000060</ElementInputStatus>
                        <ElementResultStatus>F0F000F00000F00000E0</ElementResultStatus>
                        <ElementRelevance>10100010000010000010</ElementRelevance>
                        <ExtElementStatus>00000000000000100000</ExtElementStatus>
                        <AddressResolutionCode>00000000000000000000</
AddressResolutionCode>
                        <AddressType>B</AddressType>
                        <LanguageISO3>ENG</LanguageISO3>
                        <Address>
                           <Building>
                              <string>REDBOURNE HOUSE</string>
                           </Building>
                           <Street>
                              <string>REDBOURNE AVENUE</string>
                           </Street>
                           <Locality>
                              <string>LONDON</string>
                           </Locality>
                           <PostalCode>
                              <string>N3 2BT</string>
                           </PostalCode>
                           <Country>
                              <string>UNITED KINGDOM</string>
                           </Country>
                           <DeliveryAddressLines>
                              <string>REDBOURNE HOUSE</string>
                              <string>REDBOURNE AVENUE</string>
                           </DeliveryAddressLines>
                           <CountrySpecificLocalityLine>
                              <string>LONDON</string>
                           </CountrySpecificLocalityLine>
                           <FormattedAddress>
                              <string>REDBOURNE HOUSE</string>
                              <string>REDBOURNE AVENUE</string>
                              <string>LONDON</string>
                              <string>N3 2BT</string>
                              <string>UNITED KINGDOM</string>
                           </FormattedAddress>
                           <AddressComplete>REDBOURNE HOUSE;REDBOURNE AVENUE;LONDON;N3
2BT;UNITED KINGDOM</AddressComplete>
                        </Address>
                     </ResultData>
                  </ResultDataSet>
                  <Enrichments>
                     <Enrichment>
                        <Type>GEOCODING</Type>
                        <PayloadXml><![CDATA[<GEOCODING><Status>EGCB</
Status><COMPLETE>51.6019848373084, -0.189663122839373 WGS84</
COMPLETE><LATITUDE>51.6019848373084</LATITUDE><LONGITUDE>-0.189663122839373</
LONGITUDE><LAT_LONG_UNIT>WGS84</LAT_LONG_UNIT></GEOCODING>]]></PayloadXml>
                     </Enrichment>
                  </Enrichments>
               </Result>
            </Results>
         </ProcessResult>
      </ProcessResponse>
   </soap:Body>
</soap:Envelope>
```

RELATED TOPICS:

- “GeoCodingType” on page 44
- “Enrichments Parameter” on page 56
- “Example: A V4 Address with GeoCoding” on page 77

Example: Bulgaria Address Validation Results in Cyrillic and Latin Scripts

The following examples show the result outputs when you validate a Bulgaria address with and without the `PreferredScript` setting.

Bulgaria Address Validation Without Specifying the Preferred Script

The following request does not contain the `PreferredScript` parameter. The input address is in English and uses the Latin script.

```
<soapenv:Envelope   xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:v1="http://validator5.AddressDoctor.com/Webservice5/v3">
<soapenv:Header/>
<soapenv:Body>
<Process   xmlns="http://validator5.AddressDoctor.com/Webservice5/v3">
    <login>*****</login>
    <password>*****</password>
    <parameters>
        <ProcessMode>BATCH</ProcessMode>
    </parameters>
    <addresses>
        <Address>
            <FormattedAddress>
                <string>Suite 8</string>
                <string>32 Parchevich Str.</string>
                <string>1000 Sofia</string>
                <string>Bulgaria</string>
            </FormattedAddress>
        </Address>
    </addresses>
</Process>
</soapenv:Body></soapenv:Envelope>
```

Because the request does not specify the preferred script, Informatica Address Verification returns the validated address in the Cyrillic script:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/
XMLSchema">
   <soap:Body>
      <ProcessResponse xmlns="http://validator5.AddressDoctor.com/Webservice5/v3">
         <ProcessResult>
            <StatusCode>100</StatusCode>
            <StatusMessage>OK</StatusMessage>
            <Results>
               <Result>
                  <ProcessStatus>C4</ProcessStatus>
                  <CountryISO3>BGR</CountryISO3>
                  <ResultDataSet>
                     <ResultData>
                        <ResultNumber>1</ResultNumber>
                        <MailabilityScore>4</MailabilityScore>
                        <ResultPercentage>84.35</ResultPercentage>
                        <ElementInputStatus>60500040600000200060</ElementInputStatus>
                        <ElementResultStatus>F0E08070F000004000E0</ElementResultStatus>
                        <ElementRelevance>10101000000000000010</ElementRelevance>
                        <ExtElementStatus>00000000000000B00000</ExtElementStatus>
                        <AddressResolutionCode>00000000000000000000</
AddressResolutionCode>
                        <AddressType>S</AddressType>
                        <LanguageISO3>ENG</LanguageISO3>
                        <Address>
                           <SubBuilding>
                              <string>Suite 8</string>
                           </SubBuilding>
                           <Street>
                              <string>ул.Петър Парчевич</string>
                           </Street>
                           <HouseNumber>
                              <string>32</string>
                           </HouseNumber>
                           <Locality>
                              <string>София</string>
                           </Locality>
                           <PostalCode>
                              <string>1000</string>
                           </PostalCode>
                           <Province>
                              <string>София</string>
                           </Province>
                           <Country>
                              <string>BULGARIA</string>
                           </Country>
                           <DeliveryAddressLines>
                              <string>Ул.Петър Парчевич 32, Suite 8</string>
                           </DeliveryAddressLines>
                           <CountrySpecificLocalityLine>
                              <string>1000 София</string>
                           </CountrySpecificLocalityLine>
                           <FormattedAddress>
                              <string>Ул.Петър Парчевич 32, Suite 8</string>
                              <string>1000 София</string>
                              <string>BULGARIA</string>
                           </FormattedAddress>
                           <AddressComplete>Ул.Петър Парчевич 32, Suite 8;1000
София;BULGARIA</AddressComplete>
                        </Address>
                     </ResultData>
                  </ResultDataSet>
               </Result>
            </Results>
         </ProcessResult>
      </ProcessResponse>
   </soap:Body>
</soap:Envelope>
```

Bulgaria Address Validation with the PreferredScript Parameter in the Request

In the following request, the `PreferredScript` parameter is set to LATIN. The input address is in English and uses the LATIN script.

```
<soapenv:Envelope   xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:v1="http://validator5.AddressDoctor.com/Webservice5/v3">
<soapenv:Header/>
<soapenv:Body>
<Process   xmlns="http://validator5.AddressDoctor.com/Webservice5/v3">
    <login>*****</login>
    <password>*****</password>
    <parameters>
        <ProcessMode>BATCH</ProcessMode>
        <ValidationParameters>
            <PreferredScript>LATIN</PreferredScript>
        </ValidationParameters>
    </parameters>
    <addresses>
        <Address>
            <FormattedAddress>
                <string>Suite 8</string>
                <string>32 Parchevich Str.</string>
                <string>1000 Sofia</string>
                <string>Bulgaria</string>
            </FormattedAddress>
        </Address>
    </addresses>
```

```
</Process>
</soapenv:Body></soapenv:Envelope>
```

The result output is in English and uses the Latin script.

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/
XMLSchema">
   <soap:Body>
      <ProcessResponse xmlns="http://validator5.AddressDoctor.com/Webservice5/v3">
         <ProcessResult>
            <StatusCode>100</StatusCode>
            <StatusMessage>OK</StatusMessage>
            <Results>
               <Result>
                  <ProcessStatus>C4</ProcessStatus>
                  <CountryISO3>BGR</CountryISO3>
                  <ResultDataSet>
                     <ResultData>
                        <ResultNumber>1</ResultNumber>
                        <MailabilityScore>4</MailabilityScore>
                        <ResultPercentage>84.35</ResultPercentage>
                        <ElementInputStatus>60500040600000200060</ElementInputStatus>
                        <ElementResultStatus>F0E08070F000004000E0</ElementResultStatus>
                        <ElementRelevance>10101000000000000010</ElementRelevance>
                        <ExtElementStatus>00000000000000B00000</ExtElementStatus>
                        <AddressResolutionCode>00000000000000000000</
AddressResolutionCode>
                        <AddressType>S</AddressType>
                        <LanguageISO3>ENG</LanguageISO3>
                        <Address>
                           <SubBuilding>
                              <string>Suite 8</string>
                           </SubBuilding>
                           <Street>
                              <string>ul.Petar Parchevich</string>
                           </Street>
                           <HouseNumber>
                              <string>32</string>
                           </HouseNumber>
                           <Locality>
                              <string>Sofia</string>
                           </Locality>
                           <PostalCode>
                              <string>1000</string>
                           </PostalCode>
                           <Province>
                              <string>Sofia</string>
                           </Province>
                           <Country>
                              <string>BULGARIA</string>
                           </Country>
                           <DeliveryAddressLines>
                              <string>Ul.Petar Parchevich 32, Suite 8</string>
                           </DeliveryAddressLines>
                           <CountrySpecificLocalityLine>
                              <string>1000 Sofia</string>
                           </CountrySpecificLocalityLine>
                           <FormattedAddress>
                              <string>Ul.Petar Parchevich 32, Suite 8</string>
                              <string>1000 Sofia</string>
                              <string>BULGARIA</string>
                           </FormattedAddress>
                           <AddressComplete>Ul.Petar Parchevich 32, Suite 8;1000
Sofia;BULGARIA</AddressComplete>
                        </Address>
                     </ResultData>
                  </ResultDataSet>
               </Result>
            </Results>
         </ProcessResult>
      </ProcessResponse>
   </soap:Body>
</soap:Envelope>
```

Example: Address Output with Address Type Values

The following examples show validated result output for a high-rise address in the United States and a street address in Germany.

A High-Rise Address in the United States

```
<soapenv:Envelope   xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:v1="http://validator5.AddressDoctor.com/Webservice5/v3">
<soapenv:Header/>
<soapenv:Body>
<Process   xmlns="http://validator5.AddressDoctor.com/Webservice5/v3">
    <login>*******</login>
    <password>*******</password>
    <parameters>
        <ProcessMode>BATCH</ProcessMode>
    </parameters>
    <addresses>
        <Address>
            <FormattedAddress>
                <string>15501 Weston Parkway</string>
                <string>Suite 150</string>
                <string>Cary, NC 27513</string>
                <string>USA</string>
            </FormattedAddress>
        </Address>
    </addresses>
</Process>
</soapenv:Body></soapenv:Envelope>
```

The following result output shows the address type as H, which denotes a high-rise building in the United States:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/
XMLSchema">
   <soap:Body>
      <ProcessResponse xmlns="http://validator5.AddressDoctor.com/Webservice5/v3">
         <ProcessResult>
            <StatusCode>100</StatusCode>
            <StatusMessage>OK</StatusMessage>
            <Results>
               <Result>
                  <ProcessStatus>C4</ProcessStatus>
                  <CountryISO3>USA</CountryISO3>
                  <ResultDataSet>
                     <ResultData>
                        <ResultNumber>1</ResultNumber>
                        <MailabilityScore>5</MailabilityScore>
                        <ResultPercentage>90.09</ResultPercentage>
                        <ElementInputStatus>60606050600000500060</ElementInputStatus>
                        <ElementResultStatus>F8F0F8E0F00000E000E0</ElementResultStatus>
                        <ElementRelevance>11101010100000100010</ElementRelevance>
                        <ExtElementStatus>00000000000000000000</ExtElementStatus>
                        <AddressResolutionCode>00000000000000000000
                        </AddressResolutionCode>
                        <AddressType>H</AddressType>
                        <LanguageISO3>ENG</LanguageISO3>
                        <Address>
                           <SubBuilding>
                              <string>STE 150</string>
                           </SubBuilding>
                           <Street>
                              <string>WESTON PKWY</string>
                           </Street>
                           <HouseNumber>
                              <string>15501</string>
                           </HouseNumber>
                           <Locality>
                              <string>CARY</string>
                           </Locality>
                           <PostalCode>
                              <string>27513-8638</string>
                           </PostalCode>
                           <Province>
                              <string>NC</string>
                              <string>WAKE</string>
                           </Province>
                           <Country>
                              <string>UNITED STATES</string>
                           </Country>
                           <DeliveryAddressLines>
                              <string>15501 WESTON PKWY STE 150</string>
                           </DeliveryAddressLines>
                           <CountrySpecificLocalityLine>
                              <string>CARY NC 27513-8638</string>
                           </CountrySpecificLocalityLine>
                           <FormattedAddress>
                              <string>15501 WESTON PKWY STE 150</string>
                              <string>CARY NC 27513-8638</string>
                              <string>UNITED STATES</string>
                           </FormattedAddress>
                           <AddressComplete>15501 WESTON PKWY STE 150;CARY NC
27513-8638;UNITED STATES</AddressComplete>
                        </Address>
                     </ResultData>
                  </ResultDataSet>
               </Result>
            </Results>
         </ProcessResult>
      </ProcessResponse>
   </soap:Body>
</soap:Envelope>
```

A Street Address in Germany

```
<soapenv:Envelope   xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:v1="http://validator5.AddressDoctor.com/Webservice5/v3">
<soapenv:Header/>
<soapenv:Body>
<Process   xmlns="http://validator5.AddressDoctor.com/Webservice5/v3">
    <login>*****</login>
    <password>*******</password>
    <parameters>
        <ProcessMode>BATCH</ProcessMode>
    </parameters>
      <addresses>
        <Address>
            <FormattedAddress>
                <string>Lyoner Strasse 15</string>
                <string>60528 Frankfurt</string>
                <string>Germany</string>
            </FormattedAddress>
        </Address>
    </addresses>
</Process>
</soapenv:Body></soapenv:Envelope>
```

The following output shows the address type as **S**, which denotes a street address in Germany:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/
XMLSchema">
```

```
   <soap:Body>
      <ProcessResponse xmlns="http://validator5.AddressDoctor.com/Webservice5/v3">
         <ProcessResult>
            <StatusCode>100</StatusCode>
            <StatusMessage>OK</StatusMessage>
            <Results>
               <Result>
                  <ProcessStatus>V3</ProcessStatus>
                  <CountryISO3>DEU</CountryISO3>
                  <ResultDataSet>
                     <ResultData>
                        <ResultNumber>1</ResultNumber>
                        <MailabilityScore>5</MailabilityScore>
                        <ResultPercentage>100.00</ResultPercentage>
                        <ElementInputStatus>60500050600000000060</ElementInputStatus>
                        <ElementResultStatus>F0D880E0F000000000E0</ElementResultStatus>
                        <ElementRelevance>10100010100000000010</ElementRelevance>
                        <ExtElementStatus>00005000000000000000</ExtElementStatus>
                        <AddressResolutionCode>00000000000000000000
                        </AddressResolutionCode>
                        <AddressType>S</AddressType>
                        <LanguageISO3>DEU</LanguageISO3>
                        <Address>
                           <Street>
                              <string>Lyoner Str.</string>
                           </Street>
                           <HouseNumber>
                              <string>15</string>
                           </HouseNumber>
                           <Locality>
                              <string>Frankfurt am Main</string>
                              <string>Niederrad</string>
                           </Locality>
                           <PostalCode>
                              <string>60528</string>
                           </PostalCode>
                           <Province>
                              <string>Hessen</string>
                           </Province>
                           <Country>
                              <string>GERMANY</string>
                           </Country>
                           <DeliveryAddressLines>
                              <string>Lyoner Str. 15</string>
                           </DeliveryAddressLines>
                           <CountrySpecificLocalityLine>
                              <string>60528 Frankfurt am Main</string>
                           </CountrySpecificLocalityLine>
                           <FormattedAddress>
                              <string>Lyoner Str. 15</string>
                              <string>60528 Frankfurt am Main</string>
                              <string>GERMANY</string>
                           </FormattedAddress>
                           <AddressComplete>Lyoner Str. 15;60528 Frankfurt am 
Main;GERMANY</AddressComplete>
                        </Address>
                     </ResultData>
                  </ResultDataSet>
               </Result>
            </Results>
         </ProcessResult>
      </ProcessResponse>
   </soap:Body>
</soap:Envelope>
```

Example: South Korea Address Conversion

The following example shows how to convert a non-current version of a South Korea address to the current version. A non-current South Korea address might contain land lot-based address information and a six-digit post code. A current South Korea address contains street address information and a five-digit post code. The South Korea reference database includes the land lot-based information and street information for a given address. The reference database also includes the six-digit post code and five-digit post code for a given address.

To convert a non-current address to a current address, complete the following steps:

1. Validate the non-current address in the batch or interactive process mode.

   Set the enrichment type to SUPPLEMENTARY_KR and set the `MatchingExtendedArchive` parameter to true or 1.

   The output contains the validated address in the non-current format plus an ADDRESS_ID enrichment value.

2. Change the final character in the ADDRESS_ID value to A.

   The final character in the ADDRESS_ID value signifies the current status of the address that the value represents.

3. Submit the updated ADDRESS_ID value as a request in the address code lookup process mode.

   Address Verification returns the current address information.

Retrieve the ADDRESS_ID Value

```
Request:
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope" xmlns:v3="http://
validator5.AddressDoctor.com/Webservice5/v3">
   <soap:Header/>
   <soap:Body>
      <v3:Process>
         <v3:login>*****</v3:login>
         <v3:password>********</v3:password>
         <v3:parameters>
            <v3:ProcessMode>BATCH</v3:ProcessMode>
            <v3:ValidationParameters>
               <v3:MatchingExtendedArchive>true</v3:MatchingExtendedArchive>
               <v3:PreferredScript>Latin</v3:PreferredScript>
            </v3:ValidationParameters>
         </v3:parameters>
         <v3:addresses>
            <v3:Address>
               <v3:Country>
                  <v3:string>KOR</v3:string>
               </v3:Country>
               <v3:FormattedAddress>
                  <v3:string>247 Dangjeong-dong</v3:string>
                  <v3:string>Gunpo-si, Gyeonggi-do</v3:string>
                  <v3:string>435-713</v3:string>
               </v3:FormattedAddress>
            </v3:Address>
         </v3:addresses>
         <v3:enrichments>
            <v3:Enrichment>
               <v3:Type>Supplementary_KR</v3:Type>
            </v3:Enrichment>
         </v3:enrichments>
      </v3:Process>
   </soap:Body>
</soap:Envelope>

Response:
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope" xmlns:xsi="http://
```

```
www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
   <soap:Body>
      <ProcessResponse xmlns="http://validator5.AddressDoctor.com/Webservice5/v3">
         <ProcessResult>
            <StatusCode>100</StatusCode>
            <StatusMessage>OK</StatusMessage>
            <Results>
               <Result>
                  <ProcessStatus>C4</ProcessStatus>
                  <CountryISO3>KOR</CountryISO3>
                  <ResultDataSet>
                     <ResultData>
                        <ResultNumber>1</ResultNumber>
                        <MailabilityScore>4</MailabilityScore>
                        <ResultPercentage>91.48</ResultPercentage>
                        <ElementInputStatus>60004050600000000060</ElementInputStatus>
                        <ElementResultStatus>F08070E0F000000000E0</ElementResultStatus>
                        <ElementRelevance>10101010100000000010</ElementRelevance>
                        <ExtElementStatus>F00000F0000000000000</ExtElementStatus>
                        <AddressResolutionCode>00000000000000000000</
AddressResolutionCode>
                        <AddressType>S</AddressType>
                        <LanguageISO3>KOR</LanguageISO3>
                        <Address>
                           <Street>
                              <string>DANGJEONG-DONG</string>
                           </Street>
                           <HouseNumber>
                              <string>247</string>
                           </HouseNumber>
                           <Locality>
                              <string>GUNPO-SI</string>
                           </Locality>
                           <PostalCode>
                              <string>435-713</string>
                           </PostalCode>
                           <Province>
                              <string>GYEONGGI-DO</string>
                           </Province>
                           <Country>
                              <string>REPUBLIC OF KOREA</string>
                           </Country>
                           <DeliveryAddressLines>
                              <string>247 DANGJEONG-DONG</string>
                           </DeliveryAddressLines>
                           <CountrySpecificLocalityLine>
                              <string>GUNPO-SI, GYEONGGI-DO 435-713</string>
                           </CountrySpecificLocalityLine>
                           <FormattedAddress>
                              <string>247 DANGJEONG-DONG</string>
                              <string>GUNPO-SI, GYEONGGI-DO 435-713</string>
                              <string>REPUBLIC OF KOREA</string>
                           </FormattedAddress>
                           <AddressComplete>247 DANGJEONG-DONG;GUNPO-SI, GYEONGGI-DO
435-713;REPUBLIC OF KOREA</AddressComplete> -> Old Lot-Based Address with old 6-digit
postal code
                        </Address>
                     </ResultData>
                  </ResultDataSet>
                  <Enrichments>
                     <Enrichment>
                        <Type>SUPPLEMENTARY_KR</Type>
                        <PayloadXml><![CDATA[<SUPPLEMENTARY_KR><Status>EKR1</
Status><ADDRESS_ID>9899900BE6A56708D9D663275D</ADDRESS_ID></SUPPLEMENTARY_KR>]]></
PayloadXml>
                     </Enrichment>
                  </Enrichments>
               </Result>
            </Results>
         </ProcessResult>
      </ProcessResponse>
   </soap:Body>
</soap:Envelope>
```

Submit the ADDRESS_ID Value in Address Code Lookup Mode

```
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope" xmlns:v3="http://
validator5.AddressDoctor.com/Webservice5/v3">
   <soap:Header/>
   <soap:Body>
      <v3:Process>
         <v3:login>*****</v3:login>
         <v3:password>********</v3:password>
         <v3:parameters>
            <v3:ProcessMode>ADDRESSCODELOOKUP</v3:ProcessMode>
            <v3:ValidationParameters>
              <v3:MatchingExtendedArchive>1</v3:MatchingExtendedArchive>
              <v3:PreferredScript>Latin</v3:PreferredScript>
            </v3:ValidationParameters>
         </v3:parameters>
         <v3:addresses>
            <v3:Address>
               <v3:Country>
                  <v3:string>KOR</v3:string>
               </v3:Country>
                <v3:AddressCode>
                  <v3:CodeType>KOR_ADDRESS_ID</v3:CodeType>
                  <v3:Value>9899900BE6A56708D9D663275A</v3:Value>  -> The last value of
the ADDRESS_ID changed from D to A (A = most current address)
               </v3:AddressCode>
            </v3:Address>
         </v3:addresses>
      </v3:Process>
   </soap:Body>
</soap:Envelope>

Response:

<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope" xmlns:xsi="http://
www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
   <soap:Body>
      <ProcessResponse xmlns="http://validator5.AddressDoctor.com/Webservice5/v3">
         <ProcessResult>
            <StatusCode>100</StatusCode>
            <StatusMessage>OK</StatusMessage>
            <Results>
               <Result>
                  <ProcessStatus>A1</ProcessStatus>
                  <CountryISO3>KOR</CountryISO3>
                  <ResultDataSet>
                     <ResultData>
                        <ResultNumber>1</ResultNumber>
                        <MailabilityScore>5</MailabilityScore>
                        <ResultPercentage>100.00</ResultPercentage>
                        <ElementInputStatus>00000000000000000000</ElementInputStatus>
                        <ElementResultStatus>F0F0F0F0F000000000E0</ElementResultStatus>
                        <ElementRelevance>10101010100000000010</ElementRelevance>
                        <ExtElementStatus>00000000000000000000</ExtElementStatus>
                        <AddressResolutionCode>00000000000000000000</
AddressResolutionCode>
                        <AddressType/>
                        <LanguageISO3/>
                        <Address>
                           <Street>
                              <string>NONGSIM-RO</string>
                           </Street>
                           <HouseNumber>
                              <string>35</string>
                           </HouseNumber>
                           <Locality>
                              <string>GUNPO-SI</string>
                           </Locality>
                           <PostalCode>
                              <string>15842</string>
                           </PostalCode>
                           <Province>
                              <string>GYEONGGI-DO</string>
                           </Province>
                           <Country>
                              <string>REPUBLIC OF KOREA</string>
                           </Country>
                           <DeliveryAddressLines>
                              <string>35 NONGSIM-RO</string>
                           </DeliveryAddressLines>
                           <CountrySpecificLocalityLine>
                              <string>GUNPO-SI, GYEONGGI-DO 15842</string>
                           </CountrySpecificLocalityLine>
                           <FormattedAddress>
                              <string>35 NONGSIM-RO</string>
                              <string>GUNPO-SI, GYEONGGI-DO 15842</string>
                              <string>REPUBLIC OF KOREA</string>
                           </FormattedAddress>
                           <AddressComplete>35 NONGSIM-RO;GUNPO-SI, GYEONGGI-DO
15842;REPUBLIC OF KOREA</AddressComplete> -> New Street-Based Address with new 5-digit
postal code
                        </Address>
                     </ResultData>
                  </ResultDataSet>
               </Result>
            </Results>
         </ProcessResult>
      </ProcessResponse>
   </soap:Body>
</soap:Envelope>
```