Spectrum Technology Platform

Version 12.0 SP1

Geocoding Guide for United States - REST

Table of Contents

1 - Enterprise Geocoding Module

In this section

Enterprise Geocoding Module

The Enterprise Geocoding Module performs address standardization, address geocoding, and postal code centroid geocoding. You can enter an address and get outputs such as geographic coordinates, which can be used for detailed spatial analysis and demographics assignment. You can also enter a geocode, a point represented by a latitude and longitude coordinate, and receive address information about the provided geocode.

Components

Enterprise Geocoding Module consists of the following stages. The specific stages you have depend on your license.

- **GeocodeAddressAUS**—Takes an address in Australia and returns latitude/longitude coordinates and other information.

  > **Note:** Geocode Address AUS has been deprecated. GNAF PID Location Search is the only stage used from Geocode Address AUS. For all other Australia geocoding functions, use the Geocode Address Global component.

- **GeocodeAddressGBR**—Takes an address in Great Britain and returns latitude/longitude coordinates and other information.

  > **Note:** Geocode Address GBR supports the GBR AddressBase Plus data source. Use Geocode Address Global for the GBR Streets (TomTom) data source.

- **GeocodeAddressGlobal**—Takes an address in any supported country and returns latitude/longitude coordinates and other information. Geocode Address Global geocodes addresses only from countries you have licensed. It does not support Australia and Great Britain.
- **Geocode Address**—Takes an address located in any of the supported countries and returns the city centroid or, for some countries, postal centroid. Geocode Address World cannot geocode to the street address level.
- **Geocode Africa**—Provides street-level geocoding for many African countries. It can also determine city or locality centroids, as well as postal code centroids for selected countries.
- **Geocode Middle East** Provides street-level geocoding for many Middle East countries. It can also determine city or locality centroids. Middle East supports both English and Arabic character sets.
- **Geocode Latin America** Provides street-level geocoding for many Latin American countries. It can also determine city or locality centroids. There is postal code coverage for selected countries.
- **GeocodeUSAddress**—Takes an input address and returns latitude/longitude coordinates and other address information.
- **GNAFPIDLocationSearch**—Identifies the address and latitude/longitude coordinates for a Geocoded National Address File Persistent Identifier (G-NAF PID).

- **ReverseAPNLookup**—Takes an Assessor's Parcel Number (APN), Federal Information Processing Standards (FIPS) county code, and FIPS state code and returns the address of the parcel.
- **ReverseGeocodeUSLocation**—Takes as input a geocode (latitude and longitude coordinate) and returns the address of the location.
- **ReversePBKeyLookup**—Takes a pbKey^{TM} unique identifier as input and returns all standard returns that are provided as part of address matching.

Enterprise Geocoding Databases

The following Enterprise Geocoding Module databases are installed on the Spectrum™ Technology Platform server. Some of the databases are available by subscription from Pitney Bowes and are updated monthly or quarterly. Others are licensed from the USPS®.

U.S. Geocoding Databases (U.S. Only)

These databases contain the spatial data necessary to perform address standardization and geocoding. You must install at least one of these databases to perform geocoding for USA. You set the database that you want to match against with the processing options. Enterprise Geocoding tries to match to the database you indicate. To verify you are matching to the database you want, you can review the value returned in the StreetDataType output field.

These databases use proprietary files called GSD files. For ZIP Code centroid matching, the file us.Z9 contains all the centroid info for all states and normally has a z9 extension.

- **Centrus Enhanced Geocoding**—This database consists of TIGER data provided by the U.S. Geological Survey and address data provided by the U.S. Postal Service.
- **TomTom Geocoding**—This database provides more up-to-date data than the Centrus Enhanced Geocoding database. It requires an additional license. This data is provided by TomTom, a third-party provider of spatial data, and postal data from the U.S. Postal Service.
- **NAVTEQ Geocoding**—This database provides more up-to-date data than the Centrus Enhanced Geocoding database. It requires an additional license. NAVTEQ data is provided by NAVTEQ, a third-party provider of spatial data. For more information about these databases, contact your sales representative.
- **ZIP + 4 Centroid**—This database provides only address standardization and ZIP + 4 centroid matching. It does not provide street-level matching.

Each geocoding database has an optional Statewide Intersections Index. The Statewide Intersection Index is designed to enable fast intersection identification on a statewide basis. For example, the Statewide Intersection Index will allow the database search for "1st and Main St, CO" and return a list of possible matches in Colorado more quickly than searching the entire geocoding database for each instance of the intersection.

U.S. Points Databases (U.S. Only)

Points databases contain data for locating the center of a parcel. These databases provides enhanced geocoding accuracy for internet mapping, property and casualty insurance, telecommunications, utilities, and others.

These databases are optional, but either Centrus Enhanced Points or Centrus Premium Points is required for Reverse Assessor's Parcel Number (APN) Lookup. These databases are also separately licensed.

- **Centrus Points**—This database contains the data necessary to locate the center of a parcel or building. It does not contain Assessor's Parcel Number (APN) or elevation data.
- **Centrus Elevation**—This database contains the same data as Centrus Points, plus elevation data.
- **Centrus Enhanced Points**—This database contains the same data as Centrus Points, plus APN data.
- **Centrus Premium Points**—This database contains the same data as Centrus Points, plus both APN and elevation data.
- **Centrus TomTom Points Database**—The data in this database is provided by TomTom, a third-party provider of spatial data.
- **Master Location Data** — This database provides the best available address point location for every mailable and deliverable address in the United States.

Reverse Geocoding Database (U.S. Only)

This database contains the data you need to convert a latitude/longitude location to an address.

This database is optional, but is required for ReverseGeocodeUS. This database is also separately licensed.

Auxiliary Files (U.S. Only)

Auxiliary files contain user-defined records. You can use auxiliary files to provide custom data to use in address matching and geocode matching.

DPV® Database (U.S. Only)

The Delivery Point Validation database allows you to check the validity of any individual mailing address in the U.S. The DPV database is distributed as an optional feature and can be installed to enhance the geocoding database's ability to validate mailing addresses. Each time an edition of the geocoding database is released, a corresponding edition of the optional DPV database is released. The date of the DPV database must match the date of the geocoding database for DPV processing to function. DPV lookups may not be performed after the expiration date of the DPV database.

This database is optional, but is required for CASS™ processing. The DPV database is also required to determine ZIP + 4 and ZIP + 4 related output (DPBC, USPS record type, etc.). This database is also separately licensed.

Note:

> Postal Service licensing prohibits using DPV for the generation of addresses or address lists, and also prohibits the DPV database being exported outside the United States.

EWS Database (U.S. Only)

The Early Warning System (EWS) database contains data that prevents address records from miscoding due to a delay in postal data reaching the U.S. Postal database.

The USPS® refreshes the EWS file on a weekly basis. Unlike the DPV and LACS^Link databases, the EWS database does not need to have the same date as the geocoding database. You can download the EWS.zip file free of charge from the CASS section of the USPS® RIBBS website at:

https://ribbs.usps.gov//index.cfm?page=doclist

When you download the EWS database, you will receive a file named OUT. You must rename the OUT file to EWS.txt before using it.

LACS ^Link Database (U.S. Only)

The LACS^Link database allows you to correct addresses that have changed as a result of a rural route address converting to street-style address, a PO Box renumbering, or a street-style address changing.

This database is optional, but is required for CASS™ processing. The LACS^Link database is also required in CASS mode to receive ZIP + 4 and ZIP + 4 related output (delivery point bar code, USPS record type, etc.).

The date of the LACS^Link database must match the date of the geocoding database for LACS^Link processing to function.

Note:

> USPS licensing prohibits using LACS^Link for the generation of addresses or address lists, and also prohibits the LACS^Link database being exported outside the United States.

International Geocoding Databases

International geocoding databases contain the spatial data necessary to perform address standardization and geocoding for locations outside the U.S. Each country has its own database, and some countries have optional databases that provide enhanced geocoding.

United Kingdom AddressBase Premium Database

AddressBase Premium is a point database sourced from Ordnance Survey®, Royal Mail, and local authorities.

The AddressBase Premium database provides the highest level of precision, as reflected in S8 result codes. The database includes objects without postal addresses, such as subdivided properties, places of worship and community centers.

The AddressBase Premium database is built around the UPRN (Unique Property Reference Number). The UPRN is the unique identifier that provides a persistent reference to a unique property, regardless of changes in the property name, status, subdivision, use (such as from single occupancy to multiple occupancy) or demolition of the property. All historic, alternative, and provisional addresses are recorded against the same UPRN. The UPRN is returned with every AddressBase Premium candidate, except for Northern Ireland addresses.

Since the Ordnance Survey data source does not contain addresses for Northern Ireland, AddressBase Premium is supplemented with Royal Mail® postcode address data for Northern Ireland. This Northern Ireland data has postcode centroid (result code S3) precision only.

For more information on the AddressBase Premium, see
https://www.ordnancesurvey.co.uk/business-and-government/help-and-support/products/addressbase-premium.html
from Ordnance Survey.

United Kingdom CodePoint Database

The CodePoint Postal Address File (PAF) database provides postcode centroid geocoding. The CodePoint database is suitable for most applications involving address matching, validation, etc.

The CodePoint database is sourced from Royal Mail and covers street addresses for the UK (Great Britain and Northern Ireland). The CodePoint database is licensed for the entire dataset, rather than by region. The postcode centroid precision provided by the CodePoint database is reflected in S3 result codes.

For more information on the Royal Mail data source, see:

http://www.royalmail.com

Australia Geocoded National Address File (G-NAF)

This database provides enhanced geocoding for Australian addresses. This is the only authoritative Australian national index of locality, street and number, validated with geographic coordinates. It contains both officially recognized rural and urban addresses and unofficial addresses (aliases). Postal addresses and PO Boxes are not included. However, because some rural areas do not have adequate rural address information, roadside mail box (RMB) numbers, Lot numbers, and Block & Section numbers have been included in the G-NAF data set.

When you install this database there will be two subfolders:

• **GNAF123**—Contains the point-level dictionary. This has the highest precision of geocoding (characterized by Reliability Level 1, 2, or 3.)
• **GNAF456** —Contains the remainder of address information in G-NAF that does not meet high precision geocoding criteria (characterized by Reliability Level 4, 5, or 6.)

You must specify each of these as separate database resources in the Management Console.

We recommend that you use both databases to validate the existence of addresses but only use the GANF123 for parcel-level geocoding. If you do not require parcel-level geocodes you can use the GANF456 database for geocoding.

New Zealand Point Database

The New Zealand Point Database is based on postal point data which has a roof top precision point of each unique street address. Location X and Y returned for candidates from this database are roof top precision.

This data is maintained by the government authority, Land Information New Zealand. This database is a monthly update from what the local district councils supply.

Other Point Databases for International Countries

A number of other point databases are available for international countries. In addition to USA, UK, Australia, and New Zealand, the Enterprise Geocoding Module point databases are available for the following countries:

- Andorra
- Austria
- Belgium
- Canada
- Czech Republic
- Denmark
- France
- French Guiana
- Germany
- Gibraltar
- Hong Kong
- India
- Ireland
- Japan
- Luxembourg
- Malaysia
- Martinique
- Mayotte
- Mexico
- Monaco
- Morocco
- Netherlands
- Portugal
- Reunion
- Singapore
- Slovakia
- Spain
- Sweden

Additional point databases may be available in future releases. Contact your sales representative for information on licensing the Enterprise Geocoding Module point databases.

Adding an Enterprise Geocoding Module U.S. Database Resource

Whenever you install a new database resource or modify an existing database resource you must define it in Management Console in order for it to become available on your system. This procedure describes how to add or modify a U.S. database resource for the Enterprise Geocoding Module.

1. If you haven't already done so, install the dataset files on your system. For instructions on installing databases, see the *Spectrum™ Technology Platform Installation Guide*.

2. In Management Console, under **Resources**, choose **Spectrum Databases**.

3. To create a new database resource, click the Add button ⊞. To make changes to an existing database resource, select it, then click the Edit button ✎. To delete a database resource, select it, then click the delete button 🗑.

4. If you are creating a new database resource, on the **Add Database** page, enter a name for the database resource in the **Name** field. The name can be anything you choose. If you are creating a new database by copying an existing one, rename the default name as needed. You cannot modify the name of an existing database resource, because any services or jobs that reference the database resource by its original name would fail.

   > **Note:** If you are using the Enterprise Geocoding Module with the Siebel Module, name the U.S. geocoding database **KGDDatasource**.

5. In the **Pool size** field, specify the maximum number of concurrent requests you want this database to handle.

   The optimal pool size varies by module. You will generally see the best results by setting the pool size between one-half to twice the number of CPUs on the server, with the optimal pool size for most modules being the same as the number of CPUs. For example, if your server has four CPUs you may want to experiment with a pool size between 2 (one-half the number of CPUs) and 8 (twice the number of CPUs) with the optimal size possibly being 4 (the number of CPUs).

6. In the **Module** field, select `GeoStan`. The **Type** field, `US Geographic Coders`, is selected by default.

7. To add a new dataset, under **Data Source**, click the Add button ⊞. In the **Path** field, specify the folder that contains the dataset files.You can type in a path or click the Browse button 📄 to locate the dataset you need, then click **Ok**. You can include multiple dataset paths. This allows you to specify a single resource that includes both the primary dataset and any optional datasets you may have, such as DPV, LACS[Link], or elevation data.

   For DPV and LACS[Link] datasets, the interface provides `Yes/No` indicators on whether DPV and LACS[Link] data are detected in any of the defined paths for the database resource.

8. If you have additional datasets to add, click the Add button ⊞. To edit a dataset path, select the dataset, then click the Edit button ✎. To remove a dataset, select the dataset, then click the delete button 🗑.

9. If you have multiple datasets and want to reorder them, select a dataset and click the Up ⬆ or Down ⬇ button to move the file in the dataset list.

    > **Note:** The order of the datasets sets the priority of the dataset to use when a match is made to multiple same-level datasets; for example, a match made to 2 streets datasets. In this case, the dataset listed higher will be used.

10. When you are done with your changes, click **Save**.

Geocoding Concepts

Geocoding is the process of determining the latitude/longitude coordinates of an address. There are different ways that an address can be geocoded. In order of most accurate to least accurate, these methods are:

Point Level Matching

Point-level matching locates the center of the actual building footprint or parcel. This is the most accurate type of geocode and is used in industries such as internet mapping, insurance, telecommunications, and utilities.
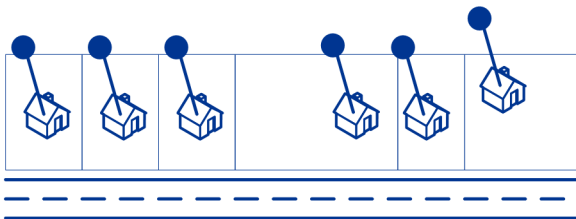
Centerline matching is used with point-level matching to tie a point-level geocode with its parent street segment. This provides you with additional data about the parent street segment that is not retrievable using only the point-level match. The output information also includes the bearing from the point data geocode to the centerline match.

Expanded Centroids

In some cases, more than one point-level geocode is available for an address matched in Master Location Data (MLD). For more information about the different types of point-level geocodes, see the "APnn" definitions in **Address Location Codes** on page 178. When more than one point-level geocode is available from MLD data, only the highest quality geocode is returned with the matched address data.

The Expanded Centroids feature is available with MLD and the presence of an optional database *us_cents.gsc*. If an address match is found in MLD, and the optional database *us_cents.gsc* is added as a database resource, the optional *us_cents.gsc* is searched for additional geocodes for the matched address. If additional geocodes are found for the matched address, these are returned. The returned location code for an Expanded Centroids match will have an "`APnn`" value with a data type of "`MASTER LOCATION`".

Street Matching

Street matching identifies the approximate location of an address on a street segment. In street matching, the location is determined by calculating the approximate location of a house number based on the range of numbers in the location's street. For example, if the address is on a street segment with a range of addresses from 50 to 99, then it is assumed that the house number 75 would be in the middle of the street segment. This method assumes that the addresses are evenly spaced along the street segment. As a result, it is not as exact as point matching since addresses may not be evenly distributed along a street segment.

For example, the following diagram shows the results of street-level matching along a segment with unevenly-spaced buildings. The first three buildings are fairly accurately geocoded because they are evenly spaced. The fourth building, however, resides on a slightly larger parcel than the others along this street. Since street-level matching assumes that the buildings are evenly spaced, the result is that fourth, fifth, and sixth houses are not as precise as the first three. If you were to use point-level geocoding, the results would be more accurate.
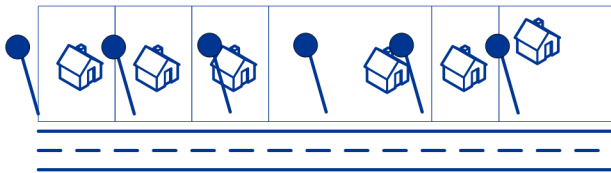
Centroid Matching

ZIP Code centroid matching is a center point of an area defined by either a ZIP Code or a ZIP + 4, and is the least accurate type of geocode. A ZIP Centroid is the center of a ZIP Code; a ZIP + 4 centroid is the center of a ZIP + 4. Since a ZIP + 4 represents a smaller area than a ZIP Code, a ZIP + 4 centroid is more accurate than a ZIP Code centroid.

The following diagram illustrates centroid matching. All six houses would have the same geocode in this example because they all reside in the same ZIP + 4 code.
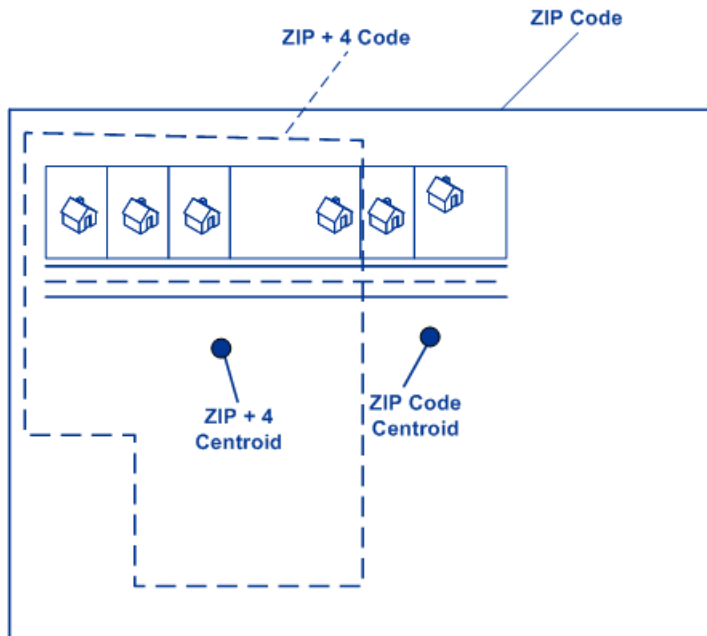
Geocoding Match Strategies for Non-U.S. Locations

The Enterprise Geocoding Module offers a variety of options for controlling geocoding precision and match rate. The following information describes different approaches for matching which you can apply to any country geocoder except the U.S. geocoder (GeocodeUSAddress), which has a different set of options.

Maximizing the Match Rate

To generate the highest match rate possible, do not specify house number, street, and city/locality.

Another way to maximize the match rate is by setting. This means that the geocoder will fall back to the four-digit postcode centroid if a close street level match cannot be made. While this scenario might yield false positives, it may be the best matching solution when you have large databases to geocode.

You should evaluate if the percentage of false positives will affect your analysis. To reduce the number of false positives without sacrificing hit rate, analyze the result codes after a geocoding session and adjust your settings accordingly.

Maximizing Precision

If your analysis requires highly precise geocoded addresses, choose a strategy in which the geocoder returns the maximum percentage of high precision geocodes and the lowest number of imprecise matches (false positives). To do this, use the to require close matches to match on all address elements. Also, set .

This technique may produce a lower percentage match rate, but will provide the best precision.

Balancing Match Rate and Precision

You may want to use a balanced strategy between match rate and geographic precision. That is, you may want to geocode as many records as possible automatically, but at the same time want to minimize the number of weaker matches (false positives). For example, false positives can occur when the geocoder:

• finds a street that sounds like the input street.
• finds the same street in another city (if postal code match is not required).
• finds the street but with a different house number (if house number is not required).

The following settings may achieve a good balance between match rate and precision:

• **CloseMatchesOnly**—Specify "Y".
• **MustMatchHouseNumber**—Specify "Y".
• **MustMatchStreet**—Specify "Y".
• **FallbackToPostal**—Specify "N".

Postal Concepts

The following sections contain information about postal concepts used by the Enterprise Geocoding Module.

> **Note:** This section and the topics of Locatable Address Conversion System, Delivery Point Validation, and Early Warning System are relevant to USA geocoding only.

Dual Addresses

GeocodeUSAddress can process input that contains two addresses for the same record on the same address line. For example, GeocodeUSAddress can process the following input address:

3138 HWY 371
PO BOX 120
PRESCOTT AR 71857

GeocodeUSAddress does not recognize dual addresses where the two addresses are both street addresses. For example, GeocodeUSAddress does NOT recognize 135 Main St 4750 Walnut St Ste 200. GeocodeUSAddress does recognize dual addresses where the two addresses are the same type of address but are not street addresses. For example, GeocodeUSAddress does recognize PO BOX 12 PO BOX 2000.

After GeocodeUSAddress parses the dual address, it searches for a match. GeocodeUSAddress determines which address has preference for a match based on the processing mode. In CASS mode, GeocodeUSAddress ignores the prefer PO Box and prefer street options, and attempts to

find a match based on the following order: PO Box, Street, Rural Route, and General Delivery. In Relaxed mode, GeocodeUSAddress recognizes the Address Preference (AddressPreference) input option.

> **Note:** GeocodeUSAddress does not perform dual address processing in Exact and Close mode. GeocodeUSAddress does not perform dual address processing on multi-line addresses.

Locatable Address Conversion System (LACS)

The USPS® Locatable Address Conversion System (LACS) corrects addresses that have changed as a result of a rural route address converting to street-style address, a PO Box renumbering, or a street-style address changing. The following are examples of LACS^Link conversions:

- Rural Route Converted to Street-Style Address: Old Address: RR 3 Box 45 New Address: 1292 North Ridgeland Drive
- Street Renamed and Renumbered: Old Address: 23 Main Street New Address: 45 West First Avenue
- PO Box Renumbered: Old Address: PO Box 453 New Address: PO Box 10435

LACS^Link is required for CASS processing.

Delivery Point Validation

Delivery Point Validation (DPV®) is a United States Postal Service® (USPS®) technology that validates the accuracy of address information down to the individual mailing address. By using DPV® to validate addresses, you can reduce undeliverable-as-addressed (UAA) mail, thereby reducing postage costs and other business costs associated with inaccurate address information.

> **Note:** DPV® is only available for U.S. addresses.

Without DPV®, the address validation process only verifies that an individual address is within a range of valid addresses for the given street. For example, the USPS data indicates that the range of addresses on Maple Lane is 500 to 1000. You attempt to validate an address of 610 Maple Ln. Without DPV®, this address would appear to be valid because it is in the range of 500 to 1000. However, in reality the address 610 Maple Ln does not exist: the house numbers in this section of the street are 608, 609, 613, and 616. With DPV® processing, you would be alerted to the fact that 610 Maple Ln does not exist and you could take action to correct the address.

DPV® also provides unique address attributes to help produce more targeted mailing lists. For example, DPV® can indicate if a location is vacant and can identify commercial mail receiving agencies (CMRAs) and private mail boxes.

Although DPV® can validate the accuracy of an existing address, you cannot use DPV® to create address lists. For example, you can validate that 123 Elm Street Apartment 6 exists, but you cannot ask if there is an Apartment 7 at the same street address. To prevent the generation of address lists, the DPV® database contains false positive records. False positive records are artificially manufactured addresses that reside in a false positive table. For each negative response that occurs

in a DPV® query, a query is made to the false positive table. A match to this table will stop DPV® processing.

Early Warning System (EWS)

The Early Warning System (EWS) provides up-to-date address information for new and recently changed addresses that have not yet been updated in the monthly USPS database. EWS prevents address records from miscoding due to a delay in postal data reaching the USPS® databases.

The older the U.S. Postal Database, the higher potential you have for miscoding addresses. When a valid address is miscoded because the address it matches to in the U.S. Postal Database is inexact, it will result in a broken address.

EWS data consists of partial address information limited to the ZIP Code™, street name, predirectional, postdirectional, and a suffix. For an address record to be EWS-eligible, it must be an address not present on the most recent monthly production U.S. Postal Database.

The USPS® refreshes the EWS file on a weekly basis. You can download the EWS file from the USPS® website at **https://ribbs.usps.gov/cassmass/documents/tech_guides/**.

2 - GeocodeUSAddress

GeocodeUSAddress takes an address and returns latitude/longitude coordinates. GeocodeUSAddress also standardizes and validates addresses using data from the U.S. Postal Service.

GeocodeUSAddress can also geocode intersections. Instead of entering a mailing address, you can enter and intersection such as "Pearl St. and 28th" and obtain the coordinates of the intersection.

GeocodeUSAddress is part of the Enterprise Geocoding Module. For more information about the Enterprise Geocoding Module, see **Enterprise Geocoding Module** on page 4.

In this section

GeocodeUSAddress

Resource URL

JSON endpoint:

```
http://server:port/rest/GeocodeUSAddress/results.json
```

XML endpoint:

```
http://server:port/rest/GeocodeUSAddress/results.xml
```

Example with JSON Response

The following example requests a JSON response:

```
http://myserver:8080/rest/GeocodeUSAddress/results.json?
Data.AddressLine1=3001+Summer+St&Data.City=Stamford&Data.StateProvince=CT
```

The JSON returned by this request would be:

```
{"output_port": [{
    "BlockSuffix": "",
    "CBSADivisionCode": "",
    "CBSAMetro": "Y",
    "CBSACode": "14860",
    "CensusBlockID": "090010212002010",
    "USFIPSCountyNumber": "001",
    "CSACode": "408",
    "CensusTract": "021200",
    "USFIPSStateCode": "09",
    "USFIPSStateCountyCode": "09001",
    "Latitude": "41.071067",
    "Longitude": "-73.549161",
    "StreetSide": "R",
    "BlockLine1": "",
    "BlockLine2": "",
    "BlockLine3": "",
    "BlockLine4": "",
    "BlockLine5": "",
    "BlockLine6": "",
    "FirmName": "Stamford Square",
    "AddressLine1": "3001 Summer St",
    "AddressLine2": "",
    "LastLine": "Stamford, CT  06905-4317",
    "StreetName": "Summer",
    "CrossStreetName": "",
```

```
"LeadingDirectional": "",
"CrossStreetLeadingDirectional": "",
"HouseNumber": "3001",
"HouseNumber2": "",
"TrailingDirectional": "",
"CrossStreetTrailingDirectional": "",
"StreetSuffix": "St",
"CrossStreetSuffix": "",
"ApartmentLabel": "",
"ApartmentLabel2": "",
"ApartmentNumber": "",
"ApartmentNumber2": "",
"AdditionalInputData": "",
"City": "Stamford",
"StateProvince": "CT",
"PostalCode.Base": "06905",
"PostalCode.AddOn": "4317",
"PostalCode": "06905-4317",
"PrivateMailbox.Designator": "",
"PrivateMailbox": "",
"USUrbanName": "",
"Country": "United States of America",
"RRHC": "",
"USCarrierRouteCode": "C006",
"USBCCheckDigit": "7",
"PostalBarCode": "431799",
"DeliveryPointCode": "99",
"GovernmentBuilding": "",
"USLOTCode": "0012A",
"USCarrierRouteSort": "B",
"USCityDelivery": "Y",
"PostalCodeClass": "",
"PostalFacility": "P",
"PostalCodeUnique": "",
"CityStateRecordName": "Stamford",
"CityPreferredName": "Stamford",
"CityShortName": "Stamford",
"DPV": "D",
"DPVFootnote": "AAN1",
"CMRA": "N",
"Alternate": "B",
"HouseNumberHigh": "3001",
"HouseNumberLow": "3001",
"HouseNumberParity": "O",
"UnitNumberHigh": "",
"UnitNumberLow": "",
"UnitNumberParity": "",
"PostalCodeExtensionHigh": "4317",
"PostalCodeExtensionLow": "4317",
"AddressLineResolved": "",
"EWSMatch": "",
"RecordType.Default": "Y",
"Intersection": "F",
```

```
    "IsAlias": "N01",
    "LACSAddress": "",
    "LocationCode": "AS0",
    "MatchCode": "S90",
    "RecordType": "HighRise",
    "GeoStanMatchScore": "0.0000000",
    "CountryLevel": "A",
    "StreetDataType": "TOMTOM",
    "StreetDataCode": "2",
    "DatabaseVersion": "July 2015",
    "ExpirationDate": "11/27/2015",
    "BlockLeft": "090010213001010",
    "BlockRight": "090010212002010",
    "BlockSuffixLeft": "",
    "BlockSuffixRight": "",
    "RoadClass": "00",
    "SegmentDirection": "F",
    "SegmentHouseNumberHigh": "3001",
    "SegmentHouseNumberLow": "2988",
    "SegmentCode": "72822",
    "SegmentLength": "",
    "SegmentParity": "R",
    "PointCode": "",
    "Confidence": "89",
    "ProcessedBy": "KGL",
    "Geocoder.MatchCode": "S5HPNTSCZA",
    "IsCloseMatch": "Y",
    "GeoConfidenceCode": "ADDRESS",
    "GeoConfidenceCentroidLatitude": "41.071067",
    "GeoConfidenceCentroidLongitude": "-73.549161",
    "PBKey": "",
    "Status": "",
    "Status.Code": "",
    "Status.Description": "",
    "StreetSegmentPoints": [{"SegmentPoints":    [
            {
        "Latitude": "41.071051",
        "Longitude": "-73.549348"
      },
            {
        "Latitude": "41.071117",
        "Longitude": "-73.549338"
      }
    ]}],
    "user_fields": []
}]}
```

Example with XML Response

The following example requests an XML response:

```
http://myserver:8080/rest/GeocodeUSAddress/results.xml?
Data.AddressLine1=3001+Summer+St&Data.City=Stamford&Data.StateProvince=CT
```

The XML returned by this request would be:

```
<xml.GeocodeUSAddressResponse
xmlns="http://www.pb.com/spectrum/services/GeocodeUSAddress">
    <output_port>
        <Address>
            <BlockSuffix/>
            <CBSADivisionCode/>
            <CBSAMetro>Y</CBSAMetro>
            <CBSACode>14860</CBSACode>
            <CensusBlockID>090010212002010</CensusBlockID>
            <USFIPSCountyNumber>001</USFIPSCountyNumber>
            <CSACode>408</CSACode>
            <CensusTract>021200</CensusTract>
            <USFIPSStateCode>09</USFIPSStateCode>
            <USFIPSStateCountyCode>09001</USFIPSStateCountyCode>
            <Latitude>41.071067</Latitude>
            <Longitude>-73.549161</Longitude>
            <StreetSide>R</StreetSide>
            <BlockLine1/>
            <BlockLine2/>
            <BlockLine3/>
            <BlockLine4/>
            <BlockLine5/>
            <BlockLine6/>
            <FirmName>Stamford Square</FirmName>
            <AddressLine1>3001 Summer St</AddressLine1>
            <AddressLine2/>
            <LastLine>Stamford, CT  06905-4317</LastLine>
            <StreetName>Summer</StreetName>
            <CrossStreetName/>
            <LeadingDirectional/>
            <CrossStreetLeadingDirectional/>
            <HouseNumber>3001</HouseNumber>
            <HouseNumber2/>
            <TrailingDirectional/>
            <CrossStreetTrailingDirectional/>
            <StreetSuffix>St</StreetSuffix>
            <CrossStreetSuffix/>
            <ApartmentLabel/>
            <ApartmentLabel2/>
            <ApartmentNumber/>
            <ApartmentNumber2/>
            <AdditionalInputData/>
            <City>Stamford</City>
            <StateProvince>CT</StateProvince>
```

```
<PostalCode.Base>06905</PostalCode.Base>
<PostalCode.AddOn>4317</PostalCode.AddOn>
<PostalCode>06905-4317</PostalCode>
<PrivateMailbox.Designator/>
<PrivateMailbox/>
<USUrbanName/>
<Country>United States of America</Country>
<RRHC/>
<USCarrierRouteCode>C006</USCarrierRouteCode>
<USBCCheckDigit>7</USBCCheckDigit>
<PostalBarCode>431799</PostalBarCode>
<DeliveryPointCode>99</DeliveryPointCode>
<GovernmentBuilding/>
<USLOTCode>0012A</USLOTCode>
<USCarrierRouteSort>B</USCarrierRouteSort>
<USCityDelivery>Y</USCityDelivery>
<PostalCodeClass/>
<PostalFacility>P</PostalFacility>
<PostalCodeUnique/>
<CityStateRecordName>Stamford</CityStateRecordName>
<CityPreferredName>Stamford</CityPreferredName>
<CityShortName>Stamford</CityShortName>
<DPV>D</DPV>
<DPVFootnote>AAN1</DPVFootnote>
<CMRA>N</CMRA>
<Alternate>B</Alternate>
<HouseNumberHigh>3001</HouseNumberHigh>
<HouseNumberLow>3001</HouseNumberLow>
<HouseNumberParity>O</HouseNumberParity>
<UnitNumberHigh/>
<UnitNumberLow/>
<UnitNumberParity/>
<PostalCodeExtensionHigh>4317</PostalCodeExtensionHigh>
<PostalCodeExtensionLow>4317</PostalCodeExtensionLow>
<AddressLineResolved/>
<EWSMatch/>
<RecordType.Default>Y</RecordType.Default>
<Intersection>F</Intersection>
<IsAlias>N01</IsAlias>
<LACSAddress/>
<LocationCode>AS0</LocationCode>
<MatchCode>S90</MatchCode>
<RecordType>HighRise</RecordType>
<GeoStanMatchScore>0.0000000</GeoStanMatchScore>
<CountryLevel>A</CountryLevel>
<StreetDataType>TOMTOM</StreetDataType>
<StreetDataCode>2</StreetDataCode>
<DatabaseVersion>July 2015</DatabaseVersion>
<ExpirationDate>11/27/2015</ExpirationDate>
<BlockLeft>090010213001010</BlockLeft>
<BlockRight>090010212002010</BlockRight>
<BlockSuffixLeft/>
<BlockSuffixRight/>
```

```
            <RoadClass>00</RoadClass>
            <SegmentDirection>F</SegmentDirection>
            <SegmentHouseNumberHigh>3001</SegmentHouseNumberHigh>
            <SegmentHouseNumberLow>2988</SegmentHouseNumberLow>
            <SegmentCode>72822</SegmentCode>
            <SegmentLength/>
            <SegmentParity>R</SegmentParity>
            <PointCode/>
            <Confidence>89</Confidence>
            <ProcessedBy>KGL</ProcessedBy>
            <Geocoder.MatchCode>S5HPNTSCZA</Geocoder.MatchCode>
            <IsCloseMatch>Y</IsCloseMatch>
            <GeoConfidenceCode>ADDRESS</GeoConfidenceCode>

<GeoConfidenceCentroidLatitude>41.071067</GeoConfidenceCentroidLatitude>


<GeoConfidenceCentroidLongitude>-73.549161</GeoConfidenceCentroidLongitude>

            <StreetSegmentPoints>
               <StreetSegments>
                  <SegmentPoints>
                     <SegmentPoint>
                        <Latitude>41.071051</Latitude>
                        <Longitude>-73.549348</Longitude>
                     </SegmentPoint>
                     <SegmentPoint>
                        <Latitude>41.071117</Latitude>
                        <Longitude>-73.549338</Longitude>
                     </SegmentPoint>
                  </SegmentPoints>
               </StreetSegments>
            </StreetSegmentPoints>
            <PBKey/>
            <Status/>
            <Status.Code/>
            <Status.Description/>
            <user_fields/>
         </Address>
      </output_port>
</xml.GeocodeUSAddressResponse>
```

Request

Input

GeocodeUSAddress takes an address as input. To obtain the best performance with GeocodeUSAddress and the most possible matches, your input address should be as complete as possible and free of misspellings and incomplete information. Input addresses should be as close to USPS standards as possible for the highest match rate. For information about USPS standards, see the USPS website **http://www.usps.com**.

Input addresses should contain a street address line and a lastline, or a single line with both address and lastline elements. This helps GeocodeUSAddress accurately identify an area in which to search for a match candidate, based on the city, state, and ZIP Code. However, matching can still be performed in cases where only a city is input in the lastline rather than a full lastline containing the city, state and ZIP Code elements. For more information, see **City-only Lastline Matching** on page 28.

GeocodeUSAddress also accepts a street address line with individual city, state, and ZIP Code lines instead of a last line. You should only use this type of input if you are confident that the input address is free of misspellings and incomplete information.

If you are using GeocodeUSAddress for address standardization, input addresses must have at least a street name, and either a city and state or a ZIP Code to obtain a match. If you are using GeocodeUSAddress to obtain geocoding information, input addresses only need to contain a ZIP + 4 Code to receive geocoding information.

The following table provides information about the format and layout of GeocodeUSAddress input.

GeocodeUSAddress Input Data

Parameter	Format	Description
Data.AddressLine1	String	The first address line or a street intersection.
		To specify a street intersection, use and, &, at, or @. For example, PEARL & 28th. GeocodeUSAddress does not match intersections when processing in CASS mode.
		You may enter an address range instead of an individual address number. For example, 10-12 FRONT ST. For additional information, see **Address Range Matching** on page 28.

Parameter	Format	Description
Data.AddressLine2	String	The second address line or a street intersection.
		To specify a street intersection, use and, &, at, or @. For example, PEARL & 28th. GeocodeUSAddress does not match intersections when processing in CASS mode.
Data.AddressLine2	String	Third address line.
Data.AddressLine4	String	Fourth address line.
Data.AddressLine5	String	Fifth address line.
Data.AddressLine6	String	Sixth address line.
Data.City	String	The name of the municipality, such as a city or town.
		Note: If there is any data in the input fields AddressLine3, AddressLine4, AddressLine5, or AddressLine6, GeocodeUSAddress will ignore data in the City input field.
Data.FirmName	String	The name of a business. The geocoding process attempts to match the input firm name to the recognized firm names in the USPS data for a higher quality match. If the firm name is not in the USPS data, the firm name is ignored when matching and is returned with the output.
Data.LastLine	String	The last line of an address containing the city, state, and ZIP Code.
Data.Latitude	String	Input latitude for matching using the Predictive Lastline feature. The required input format is an integer value; for example, 40018301. Decimal is not a valid input format.
		For more information, see **Predictive Lastline** on page 29.
Data.Longitude	String	Input longitude for matching using the Predictive Lastline feature. The required input format is an integer value; for example: -105240976. Decimal is not a valid input format.
		For more information, see **Predictive Lastline** on page 29.

Parameter	Format	Description
Data.PostalCode	String	The 5-digit ZIP Code or the 9-digit ZIP + 4 code. **Note:** If there is any data in the input fields AddressLine3, AddressLine4, AddressLine5, or AddressLine6, GeocodeUSAddress will ignore data in the PostalCode input field.
Data.StateProvince	String	The name or abbreviation of the state. **Note:** If there is any data in the input fields AddressLine3, AddressLine4, AddressLine5, or AddressLine6, GeocodeUSAddress will ignore data in the StateProvince input field.

How GeocodeUSAddress Processes Addresses

GeocodeUSAddress processes addresses in the following order:

1. Parses the address elements.

   GeocodeUSAddress parses input address data into single elements. Parsing occurs on data in the order in which you load the data. Even if a valid address is missing an element, GeocodeUSAddress can find a match. Some elements, such as predirectionals, may not be critical elements of some addresses. By comparing an address as input against all known addresses in a search area, GeocodeUSAddress can usually determine if any of these elements are missing or incorrect.

2. Finds possible matches within the search area.

   GeocodeUSAddress uses the last line elements of an address to determine a search area. You can specify if you want the search area based on a finance area or on an area defined by the city, state, and ZIP Code. (A Finance Area is a collection of ZIP Codes within a contiguous geographic region.) If the city and state are not in the ZIP Code, GeocodeUSAddress performs separate searches for the ZIP Code and city.

   After GeocodeUSAddress has determined the search area, it tries to match the elements from the street address line to the records in the standardized data files and does the following:

   • Checks input address ranges for missing or misplaced hyphens, and alpha-numeric ranges for proper sequence.
   • Searches for any misspellings and standard abbreviations. For example, the GeocodeUSAddress can recognize Mane for Main and KC for Kansas City.
   • Searches for any alias matches to the USPS and Spatial data (TIGER and TomTom). For example, GeocodeUSAddress recognizes that in Boulder, CO Highway 36 is know as 28th Street.

- Searches for any USPS recognized firm names for additional match verification.
- Searches for street intersection matches. Matching to an intersection is extremely useful when you are using address matching to obtain a geocode.
- Searches for addresses lines that contain a house number and unit number as the same element. For example, GeocodeUSAddress recognizes the input 4750-200 Walnut Street and performs recombination to output 4750 WALNUT ST STE 200.

  Note: The USPS does not consider intersections valid addresses for postal delivery. Therefore, the GeocodeUSAddress does not match intersections when processing in CASS mode.

3. Scores each possible match against the parsed input.

   GeocodeUSAddress compares each element in the input address to the corresponding element in the match candidates, and assigns a confidence level. GeocodeUSAddress weighs the confidence level for all of the elements within a match candidate, and assigns a final score to the sum.

   Note: GeocodeUSAddress uses a penalty scoring system. If an element does not exactly match an element in the match candidate, the GeocodeUSAddress adds a penalty to the score of the match candidate. Therefore, scores with lower numbers are better matches.

4. Determines the match.

   GeocodeUSAddress prioritizes each match candidate based on the assigned confidence score and returns as a match the candidate that has the lowest score.

   The match mode you choose determines the range that GeocodeUSAddress allows for a match. GeocodeUSAddress only returns a match if the score of the target address falls within the range designated by the selected match mode.

   In some cases, more than one match candidate may have the lowest score. In this instance, GeocodeUSAddress cannot determine on its own which record is correct, and returns a status indicating multiple matches.

   Note: If you have enabled Delivery Point Validation (DPV) processing, GeocodeUSAddress automatically attempts to resolve multiple matches using DPV.

   Along with a standardized address, GeocodeUSAddress also returns the following:

   - Geocode—Longitude and latitude for the address
   - Match code—Information about the match of the input address to the reference data
   - Location code—Precision level of a geocode
   - Parity—The side of the street on which the match resides.

   GeocodeUSAddress does not return parity when processing in relaxed mode. For more information about GeocodeUSAddress output, see **Response** on page 54.

City-only Lastline Matching

City-only lastline matching permits address matching with only a city in the input lastline. The city should be provided using either the `AddressLine1` (using single-line address input), `AddressLine2`, `LastLine` or `City` input fields.

With city-only lastline input, all of the states are searched in which the input city exists. Therefore, there is the possibility of an increase in multimatches (return of E023 or E030 Match Codes) when matching with city-only input instead of city+state input.

Restrictions:

- City-only lastlne input matching is not supported in CASS mode.
- City-only lastline is not supported when matching to User Dictionaries.
- When matching using city-only lastline, the Option.PreferZipCodeOverCity setting is ignored
- It is strongly recommended to not use city-only lastline matching in Relaxed match mode to avoid the return of false-positive matches.

Address Range Matching

Some business locations are identified by address ranges. For example, a shopping plaza could be addressed as 10-12 Front St. This is how business mail is typically addressed to such a business location. These address ranges can be geocoded to the interpolated mid-point of the range.

Address ranges are different from hyphenated (dashed) addresses that occur in some metropolitan areas. For example, a hyphenated address in Queens County (New York City) could be 243-20 147 Ave. This represents a single residence (rather than an address range) and is geocoded as a single address. If a hyphenated address returns as an exact match, GeocodeUSAddress does not attempt to obtain an address range match.

Address range matching is not available in Exact or CASS modes, since an address range is not an actual, mailable USPS® address. The following fields are not returned by address range geocoding:

- ZIP + 4® (in multiple segment cases)
- Delivery point
- Check digit
- Carrier route
- Record type
- Multi-unit
- Default flag

Address range matching works within the following guidelines:

- There must be two numbers separated by a hyphen.
- The first number must be lower than the second number.
- Both numbers must be of the same parity (odd or even) unless the address range itself has mixed odd and even addresses.

- Numbers can be on the same street segment or can be on two different segments. The segments do not have to be contiguous.
- If both numbers are on the same street segment, the geocoded point is interpolated to the approximate mid-point of the range.
- If the numbers are on two different segments, the geocoded point is based on the last valid house number of the first segment. The ZIP Code and FIPS Code are based on the first segment.
- In all cases, odd/even parity is evaluated to place the point on the correct side of the street.

Predictive Lastline

Predictive lastline allows you to match an address when only an input street address and latitude/longitude coordinates are provided, rather than the traditional street address with lastline input. For example, an input of 4750 Walnut with latitude/longitude coordinates located in Boulder, will return full address information.

Enabling Predictive Lastline

The following tables describes how to enable and set up the Predictive Lastline feature.

Directions

Set `Option.PredictiveLastLine = Y`

Set `Option.Latitude = integer value`

Set `Option.Longitude = integer value`

Set `AddressLine1 = input street address`

Additional feature information

- Predictive Lastline uses a search radius of 150 feet.
- Predictive Lastline is disabled by default.
- When Predictive Lastline is not enabled, if a singleline address is provided, any input latitude and longitude values are ignored and the input address is geocoded.
- If the input latitude/longitude coordinates fall near the borders of multiple cities, the Enterprise Geocoding Module processes all cities and returns the results of the best match. If the results are determined as equal, then a multi-match is returned.
- Predictive lastline does not require a license for reverse geocoding.
- This feature will work with any type of data set.

Input Parameters

Geocoding Options

GeocodeUSAddress Geocoding Options

The following table lists the options that control how a location's coordinates are determined.

Parameter	Description
Option.Dataset	The name of the database resource that contains the data to use in the search process. Use the database name specified in Management Console's Spectrum Databases page.
Option.Offset	Specifies the offset distance from the street segments, in feet. The range is 0 to 5280. Default = 50 feet.

The offset distance is used in street-level geocoding to prevent the geocode from being in the middle of a street. It compensates for the fact that street-level geocoding returns a latitude and longitude point in the center of the street where the address is located. Since the building represented by an address is not on the street itself, you do not want the geocode for an address to be a point on the street. Instead, you want the geocode to represent the location of the building which sits next to the street. For example, an offset of 40 feet means that the geocode will represent a point 40 feet back from the center of the street. The distance is calculated perpendicular to the portion of the street segment for the address. Offset is also used to prevent addresses across the street from each other from being given the same point. The diagram below shows an offset point in relation to the original point.

Street coordinates are accurate to 10,000ths of a degree and interpolated points are accurate to the millionths of a degree.

Parameter	Description
Option.Squeeze	Specifies the distance, in feet, to move the street segment end points toward the middle of the segment. Squeeze is used in street-level matching. Use the squeeze setting to prevent address points from residing in an intersection or too close to the end of a street.

The range is 0 to 2147483647. Default = 50 feet.

The following diagram compares the end points of a street to squeezed end points.

Squeezing the street segment endpoints affects street-level matching by reducing the length of a street segment, thereby reducing the spacing between address points along the segment. For example, if the length of a street segment is 1,000 feet and there are 10 addresses along the segent, street-level matching would result in each address being spaced 100 feet apart (1,000 ÷ 10). If you were to set a squeeze value of 100 feet, moving each street segment endpoint 100 feet torward the center of the street segment, the length of the street segment would be reduced to 800 feet (reduced by 100 feet on each end). Street-level matching would then result in addresses beging spaced 80 feet apart (800 ÷ 10).

Option.LatLonFormat	Specifies the format of the latitude/longitude returned by the geocoder.

Decimal	The latitude/longitude is returned in decimal format. Default. For example: 90.000000-180.000000
Integer	The latitude/longitude is returned in integer format. For example: 90000000-180000000

Parameter	Description
Option.Datum	Determines the North American Datum to use when geocoding datum on the input value. Datum is the mathematical model of the Earth used to calculate the coordinates on any map, chart, or survey system.

> **NAD27**   This datum does not include the Alaskan Islands or Hawaii. Latitudes and longitudes that are surveyed in the NAD27 system are valid only in reference to NAD27 and are not valid for maps outside the U.S.
>
> **NAD83**   This datum is earth-centered and defined with satellite and terrestrial data. NAD83 is compatible with the World Geodetic System 1984 (WGS84), which is the terrestrial reference frame associated with the NAVSTAR Global Positioning System (GPS) used extensively for navigation and surveying. Default.

Parameter	Description
Option.CentroidPreference	Determines the type of centroids returned by the geocoder. A centroid is the center of an area. The centroid coordinates are the average of the sets of coordinates that describe the area.

> **NoCentroids**   Do not return centroids. If an address-level geocode cannot be determined, do not attempt to determine a centroid.
>
> **AddressUnavailable**   Return a ZIP Code centroid if an address-level geocode cannot be determined (default).
>
> **AllCentroids**   Return ZIP Code centroids only. If you select this option, address-level geocodes will not be returned.

Parameter	Description
Option.FallbackToStreet	Specifies whether to attempt to return a street centroid when an address-level geocode cannot be determined. To determine a street centroid, the geocoder searches the input ZIP Code or city for the closest match. If the geocoder is able to locate the street, it returns a geocode along the matched street segment.

For example, if the input address is 5000 Walnut Street, Boulder 80301, and there is no 5000 Walnut Street, the geocoder searches for the closest match to that address within the ZIP Code 80301. If there were no input ZIP Code, the geocoder would search for the closest match to the input address within Boulder.

If the input address is Walnut Street, Boulder 80301, since there is no house number, the geocoder searches for the street within the input ZIP Code.

Street centroid geocodes are indicated by value in the LocationCode output field that begins with "C". For more information, see **Street Centroid Location Codes** on page 186.

> **Note:** This option is not available if you set Option.MatchMode to CASS.

Y	Yes, attempt to determine the street centroid when an address-level geocode cannot be determined.
N	No, do not attempt to determine the street centroid when an address-level geocode cannot be determined. Default.

Parameter	Description
Option.FallbackToGeographic	Specifies whether to attempt to return a city, county, or state centroid when an address-level geocode cannot be determined. The geocoder returns the most precise geographic centroid that it can based on the input. For example, if the input contains a valid city and state, a city centroid would be returned.

> **Note:** There are approximately 300 major cities that can be geocoded to a city centroid level even if a valid state is not provided in the input.

Geographic centroid geocodes are indicated by value in the LocationCode output field that begins with "G". For more information, see **Geographic Centroid Location Codes** on page 193.

> **Note:** This option is not available if you set Option.MatchMode to CASS.

Y	Yes, attempt to determine the geographic centroid when an address-level geocode cannot be determined.
N	No, do not attempt to determine the geographic centroid when an address-level geocode cannot be determined. Default.

Parameter	Description
Option.AddressPointInterpolation	Specifies whether to perform address point interpolation when an exact match for the address cannot be found in the geocoding database. Address point interpolation is a patented process that results in a more accurate interpolated point. It improves upon regular street segment interpolation by using point data in the interpolation process, as opposed to using street segments alone.
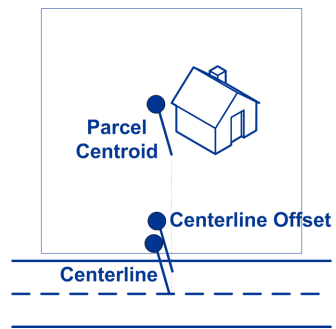
> **Note:** Address point interpolation is only available when using a point-level geocoding database. It is not available when using point addresses in an auxiliary file.

Y   Yes, perform address point interpolation.

N   No, do not perform address point interpolation. Default.

The following illustration shows how address point interpolation works. In the example, the input house number is 71. The geocoding database contains address points for 67 and 77. The street segment has a range of 11 to 501. With address point interpolation, GeocodeUSAddress performs the interpolation for the input house number 71 using the points of 67 and 77. Without address point interpolation, GeocodeUSAddress performs the interpolation with the street segment end points of 11 and 501, resulting in a far less accurate result.

Parameter	Description
Option.CenterlineOffset	The offset distance, in feet, used to calculate the street centerline coordinates. Default = 0 feet.
	If you specify a value other than 0, GeocodeUSAddress calculates the street centerline coordinates by offsetting the centerline point by the distance you specify in the direction of the parcel centroid.
	In an interpolated match, the centerline offset cannot be greater than the distance from the centerline to the interpolated address point. If you specify a centerline offset distance that is greater than this distance, the offset will be limited to the distance to the interpolated point. In effect, the centerline coordinates would be the same as the coordinates for the interpolated point.
Option.RetrieveAPN	Specifies whether to determine the address's APN (assessor's parcel number). The APN is an ID number assigned to a property by the local property tax authority. The APN is returned in the **APN** output field, which is part of the Census output group.

Note: This option requires that you have licensed and installed the Cenrus Enhanced Points or Centrus Premium Points database. APN data is not available for all addresses. See the coverage map included with the points database.

Y   Yes, return the assessor's parcel number.

N   No, do not return the assessor's parcel number. Default.

Parameter	Description
Option.RetrieveElevation	Specifies whether to return the elevation of the address. Elevation is the distance above or below sea level of a given location. The elevation is returned in the **Elevation** output field, which is part of the Latitude/Longitude output group.

> **Note:** This option requires that you have licensed and installed the Centrus Premium Points database. Elevation data is not available for all addresses. See the coverage map included with the points database.

Y	Yes, return the elevation of the address.
N	No, do not return the elevation of the address. Default.

Parameter	Description
Option.AlwaysFindCandidates	Specifies whether to enable centerline matching in order to get street centerline information.

> **Note:** Centerline matching requires that a point-level database is installed.

Y	Yes, enable centerline matching. To return centerline data in the output, you must also set `Option.OutputRecordType = N`.
N	No, do not enable centerline matching. Default.

Parameter	Description
Option.FIND_APPROXIMATE_PBKEY	When Option.FIND_APPROXIMATE_PBKEY is enabled, if an address match is not made to Master Location Data (MLD), but to a different dataset, the pbKey™ unique identifier of the nearest MLD point located within the search distance is returned. To distinguish when a fallback pbKey™ unique identifier is returned, the `PBKey` return value contains a leading character of "X" rather than "P", for example: X00001XSF1IF. Note, all of the other fields returned for the address match, including the geocode and all associated data, reflect the match results for the input address. The fallback pbKey™ unique identifier can then be used for the lookup to the GeoEnrichment dataset(s), and the attribute data for the fallback location is returned for the match. **Note:** This option requires that you have licensed and installed the Master Location Dataset. For more information, see **#geocodeusaddress_options_geocoding/conref_pbKeyFallback** on page 37. The search distance for the nearest MLD point is configurable using the `Option.SearchDistance` parameter with an allowable search radius of 0-5280 feet and a default value of 150 feet. **Y** When a matched address does not have an associated pbKey™ unique identifier, attempt to return the nearest address record's pbKey™ unique identifier. **N** When a matched address does not have an associated pbKey™ unique identifier, do not return the nearest address record's pbKey™ unique identifier. Default.
Option.SearchDistance	When the PBKey Fallback option is enabled, this field sets the distance to use when searching for the nearest address record with an associated pbKey™ unique identifier. The allowable range is 0-5280 feet. Default = 150 feet. **Note:** Supported only in forward geocoding.

A pbKey™ unique identifier is returned when a match is made to the Master Location Dataset (MLD). This field is a persistent identifier for an address. The pbKey™ serves as a lookup key with Pitney Bowes GeoEnrichment datasets to add attribute data for an address. Depending on the GeoEnrichment dataset(s) you install, the attribute data can include property ownership, real estate, census, consumer expenditure, demographic, geographic, fire and flood protection, and/or telecommunication and wireless systems information and more. Some of these datasets return point location specific data, such as property ownership and real estate, whereas others provide polygonal-based data, for example, fire and flood protection, which can identify flood plains, wildfire or rating territories.

When using PBKey Fallback, if an address match is not made to Master Location Data, but a match is made to a different dataset, the pbKey of the nearest MLD point located within the search distance

is returned. To distinguish when a fallback pbKey is returned, the PBKey return value contains a leading character of "X" rather than "P", for example: X00001XSF1IF. Note, all of the other fields returned for the address match, including the geocode and all associated data returns, reflect the match results for the input address. The fallback pbKey can then be used for the lookup to the GeoEnrichment dataset(s), and the attribute data for the fallback location is returned for the match.

The relevance and accuracy of the returned attribute data using a PBKey Fallback location is highly dependent on the type of GeoEnrichment data, as well as the PBKey Fallback search distance. PBKey Fallback is intended for use with GeoEnrichment datasets that have polygonal-based data, rather than point-specific data. For example, the PBKey Fallback option may be suitable for determining the FEMA flood zone for a given location using the Flood Risk Pro GeoEnrichment dataset since it contains data that represents a polygonal region rather than a single coordinate. However, it is important to note that the accuracy of the returned data would very much depend on the size and nature of the individual polygonal features described in the GeoEnrichment data, combined with the search distance used to locate the nearest Master Location Data point. The search distance is configurable with an allowable search radius of 0-5280 feet and a default value of 150 feet.

> **Note:** This option requires that you have licensed and installed the Master Location Dataset.

Returning ZIP Centroids with Master Location Data

ZIP Centroid returns are optionally available when matching to Master Location Data (MLD) in the Geocode US Address stage. For addresses that don't have a high-quality location, this provides access to the pbKey^{TM} unique identifier, which can be used to unlock additional information about an address using GeoEnrichment data, as well as to realize operational processing efficiencies. This allows us to ensure maximum address coverage and integrity in geocoding. The inclusion of these addresses enables us to provide a higher match rate, lower false-positive match rate, and access to the pbKey^{TM} unique identifier for all known addresses in the US.

In order to return ZIP centroids, you must add these files as database resources:

• Master Location Data
• `zipsmld.gsd` - this is a new file that contains the ZIP centroids.

To enable returning ZIP centroids, follow this procedure:

1. Go to the directory listed below. The *SpectrumDirectory* refers to the directory where you have installed the Spectrum^{TM} Technology Platform server.

    • On Windows: `\`*SpectrumDirectory*`\server\modules\geostan\`
    • On Unix/Linux: `/`*SpectrumDirectory*`/server/modules/geostan/`

2. Using a text editor, open the `java.properties` file.
3. Go to the `Return pbKey Zip Centroids` property.
4. Set `egm.us.return.pbkey.zip.centroids=true`. By default, this property is disabled (false).
5. Restart the Spectrum server.

6. Set the centroid preference to enable ZIP centroid returns: In the request to the GeocodeUSAddress resouce, set: `Option.CentroidPreference=AddressUnavailable`

Matching Options

GeocodeUSAddress Matching Options

Matching options are used to determine how address searches are performed. They let you set match preferences, criteria and restrictions, and multiple match settings so that the matching can be as strict or relaxed as you need.

Parameter	Description
Option.AddressPreference	Determines which address to use when more than one address is present in the address block. **PreferPOBox** — Uses the P.O Box. **PreferBottom** — Uses the second line entered. Default. You must select this value if you. **PreferStreetAddress** — Uses the street address.
Option.FirmNameSearch	Specifies whether to use firm name matching logic to enhance address matching. Firm matching logic matches a business name in the input to recognized business names. The input firm name does not need to be spelled correctly to obtain a match. A soundex algorithm is used to match the firm name. A suite or unit number is not required to make the match. **Note:** This type of match is not available when processing in CASS mode. One of the following: **Always** — Always attempt to match using firm name matching. If firm name matching fails, attempt to match using address matching. **OnAddressLineFail** — Use firm matching only if a match cannot be determined using address matching. **Never** — Do not use firm matching. Default. Note that the firm name may be corrected even if you specify **Never** if a match can be found using the address line data.

Parameter	Description
Option.BuildingSearch	Specifies whether to attempt to obtain a street address when the input address contains a building name with no suite or unit number.
	When this option is disabled, the geocoder is able to match to building names only if there is a unit number in the input. For example, if the building search option were disabled and you entered this input:
	5001 Chrysler Bldg New York, NY 10174
	The street address would be returned:
	405 Lexington Ave RM 5001 New York, NY 10174-5002
	With this option enabled, the geocoder is also able to obtain a street address when only a building name with no unit number is provided. For example, if you enable this option and provide this address:
	Chrysler Bldg New York, NY 10174
	You will get the street address:
	405 Lexington Ave New York, NY 10174-00
	Note: This type of match is not available when processing in CASS mode.
	Y Use firm name matching logic. Default.
	N Do not use firm name matching logic.

Parameter	Description
Option.FirstLetterSearch	Specifies whether to look for the correct first letter of a street name if the first letter is missing or incorrect. If enabled, the geocoder searches through the alphabet looking for the correct first letter to complete the street address.

> **Note:** This option is not available if the match mode is set to exact.

Y	Perform first letter search.
N	Do not perform first letter search. Default.

This example includes an incorrect first letter:

Input: 4750 nalnut boulder co 80301
Output: 4750 Walnut St Boulder CO 80301-2532

This example excludes a first letter:

Input: 4750 alnut boulder co 80301
Output: 4750 Walnut St Boulder CO 80301-2532

This example includes an extra first letter:

Input: 4750 wwalnut boulder co 80301
Output: 4750 Walnut St Boulder CO 80301-2532

Parameter	Description
Option.PredictiveLastLine	Specifies whether GeocodeUSAddress should match using the street address and input latitiude/longitude coordinates, rather than the traditional street address with lastline input.

Y	Enable Predictive Lastline processing.
N	Disable Predictive Lastline processing. Default.

For more information, see **Predictive Lastline** on page 29.

Parameter	Description
Option.PerformDPV	Specifies whether to process addresses using Delivery Point Validation (DPV). DPV is a United States Postal Service (USPS) technology that validates the accuracy of address information down to the physical delivery point. You must have licensed the optional DPV processing option to use this feature. You must also install the DPV database. To use DPV, enable this processing option and specify D in Option.OutputRecordType. **Y** Perform DPV. **N** Do not perform DPV. Default. If you use DPV, multiple matches are automatically resolved. False-positive addresses, also known as seed records, are addresses the USPS monitors to ensure users are not attempting to create a mailing list from the DPV data. If the geocoder matches an address in your input data to a false-positive address, you receive a message indicating you have encountered a false-positive address. Processing continues to the end of your job, but DPV processing is not available for this job and subsequent jobs until you have reported the false-positive address encounter to technical support and have received a new security key.
Option.PerformLACSLink	Specifies whether to process addresses using LACS^{Link}. **Y** Perform LACS^{Link} **N** Do not perform LACS^{Link}. Default. If you use LACS^{Link}, be sure to choose to specify output record types P and Q so that the fields USLACS, USLACS.ReturnCode, and LACSADDRESS are included in the output. For more information, see **Locatable Address Conversion System (LACS)** on page 15.

Parameter	Description
Option.PreferZipCodeOverCity	Specifies whether to prefer candidates that match the input ZIP over candidates that match to input city.

Note: This option is not available when processing in CASS mode.

Y	Prefer candidates that match the input ZIP Code.
N	Prefer candidates that match the input city. Default.

For example, consider this input address:

301 BRYANT ST
SAN FRANCISCO CA 94301

Without this option enabled, the best match would be the one that matches the input city name:

301 BRYANT ST
SAN FRANCISCO CA 94107-4167

With this option enabled, the best match would be the one that matches the input ZIP Code:

301 BRYANT ST
PALO ALTO CA 94301-1408

Parameter	Description
Option.FIND_SEARCH_AREA	These options set the search constraints to use when matching. These can assist in finding a match when the input address contains limited or inaccurate city or ZIP Code information. For more information, see **Search Area** on page 52. **Note:** In CASS match mode, only the search area options described in `FIND_SEARCH_AREA_DEFAULT` are available.
FIND_SEARCH_AREA_DEFAULT	The impact of the `FIND_SEARCH_AREA_DEFAULT` setting depends on the match mode you're using for matching. When `Option.FIND_SEARCH_AREA_DEFAULT` is set and you're matching using either CASS or Relaxed match mode, the search area is determined based on the `Option.CentroidPreference` setting: • If `Option.CentroidPreference` = `AllCentroids`, the `FIND_SEARCH_AREA_CITY` search area is used. • If `Option.CentroidPreference` is set to either `NoCentroids` or `AddressUnavailable`, the `FIND_SEARCH_AREA_FINANCE` search area is used. When `FIND_SEARCH_AREA_DEFAULT` is set and you're matching using any other match mode - Custom, Exact, Close or Interactive - the `FIND_SEARCH_AREA_FINANCE` search area is used.
FIND_SEARCH_AREA_FINANCE	Searches the entire Finance Area for possible streets. **Note:** This option has no effect when performing a ZIP centroid match or a geographic geocode.
FIND_SEARCH_AREA_CITY	Searches the specified city.
FIND_SEARCH_AREA_EXPANDED	Enables the setting of the search radius distance to use when matching. See `Option.FIND_SEARCH_AREA_DISTANCE` below.
FIND_EXPND_SRCH_LIM_TO_STATE	Limits the search to the state, within the search radius distance. The default search radius is 25 miles.

Parameter	Description
Option.FIND_SEARCH_AREA_DISTANCE	When the FIND_SEARCH_AREA_EXPANDED option is selected, this field allows you to enter the search radius distance to use when matching. Valid values = 0-99 (miles). Default = 25 miles.

> **Note:** Ignored in CASS match mode.

Parameter	Description
Option.KeepMultimatch	Select this option to return the list of possible matches when there is more than one possible match for the input address and a single best match cannot be identified.

Y	Return the addresses that are possible matches for the input address. Default.
N	Do not return the ambiguous matches.

Parameter	Description
Option.KeepCandidates	Select this option to return candidate addresses whenever the match attempt produces candidates. If you enable this option, the geocoder will return candidates both when the input address matches to a single address and when the input address matches multiple addresses.

This parameter differs from `Option.KeepMultimatch` in that the `Option.KeepMultimatch` parameter does not return candidates if the input address matches to a single address.

Y	Return candidates for all match attempts.
N	Do not return candidates for all matches. Default.

Parameter	Description
Option.CloseMatchesOnly	If you specify `Option.KeepCandidates=Y` you can choose to return just those candidates that are considered to be a close match. The criteria used to determine whether a candidate is a close match are those you specify in the Option.MatchMode parameter.

Y	Return close match candidates only. Default.
N	Return all candidates.

Parameter	Description

Option.MatchMode

Parameter	Description

Determines the leniency used to find a match. One of the following:

Custom	Allows you to select the specific criteria to use when matching the input address to an address in the postal database.
Exact	Requires a very tight match. This is a restrictive mode that generates the fewest number of match candidates to search, which decreases the time to obtain a match. When using this mode, ensure that your input address list is very clean; free of misspellings and incomplete addresses.
Close	Requires a moderately confident match. Generates a moderate number of match candidates.
Relax	This is the loosest match mode and generates the most match candidates, which increases the processing time and results in more multiple matches. Use this mode if your address list may contain misspellings and incomplete addresses. This mode does not respect the street parity for an address match. Default.
Interactive	Available in single-line address matching only. This mode is designed to better handle the specific matching challenges presented by interactive matching. Interactive mode permits for more flexible matching patterns and may, in some cases, return additional possible matches than relaxed match mode. This mode recognizes and parses two unit numbers on the same address line, for example a building and unit number. This mode does not respect the range parity when making an address match.

Capabilities and restrictions:

- Interactive match mode allows users to break the cardinal rule: If the user enters 123 S Main and there is only 123 N Main, a match is made and a match code is returned that reflects the modified directional.
- Interactive match mode handles cases where users transpose pre-directionals with post-directionals without penalty.
- Interactive match mode ignores the 'Prefer ZIP Code over city' setting. When the city and ZIP Code don't match correctly, the best geocoding result will be returned based on an analysis of all the input address elements.
- When operating in interactive mode, in cases where a point address or interpolated street address result cannot be determined, ZIP-9 or ZIP-7 centroid(s) may be returned.

CASS	Imposes additional rules to ensure compliance with the USPS regulations for CASS. The purpose of this mode is to create a list of mailable addresses. This mode generates a large number of match candidates. This mode deviates from the other modes in its processing. This mode does not perform intersection, building name, or spatial alias (TIGER and TomTom street name alias) matches. It does not match to candidates from data sources that do not have USPS equivalent records. This mode recognizes

Parameter	Description
	and parses two unit numbers on the same address line, for example a building and unit number.
Option.ExtendedMatchCode	Specifies whether to return the Extended Match Code. For more information, see **Extended Match Codes** on page 50.
	Y Yes, return the Extended Match Code.
	N No, do not return the Extended Match Code. Default.
Option.MustMatchInput	Specifies whether candidates must match all non-blank input fields. For example, if an input address contains a city and postal code, then candidates for this address must match the city and postal code.
	Y Yes, candidates must match all input.
	N No, candidates do not have to match all input. Default.
Option.MustMatchStreet	Specifies whether candidates must match the street name.
	Y Yes, candidates must match the street name.
	N No, candidates do not have to match the street name. Default.
Option.MustMatchStateProvince	Specifies whether candidates must match the state.
	Y Yes, candidates must match the state.
	N No, candidates do not have to match the state. Default.

Parameter	Description
Option.MustMatchHouseNumber	Specifies whether candidates must match the house number. If the input house number is not within a range from the street, GeocodeUSAddress selects the nearest range on the street which has the same parity (even or odd house number) as the input address number. GeocodeUSAddress returns one or more of the closest matches inside this range that preserves street parity. This requires GeocodeUSAddress to change the house number. The new house number is equal to one of the range's endpoints, possibly plus or minus one to preserve street parity.

> **Note:** Even when this option is disabled and an inexact match on the house number is found, GeocodeUSAddress still returns an error code.

When this option is disabled and no exact matching house number is found, a match code of either E029 (no matching range, single street segment found), or E030 (no matching range, multiple street segment) is returned.

GeocodeUSAddress does not change the house number on the output address. In order to access the inexact address number candidates, you must specify `Option.KeepMultimatch=Y`. If there are inexact house number candidates returned, the corresponding match codes begin with the letter 'H' indicating that the house number was not matched.

Additionally, even when one or more exact candidates are found, inexact matches to the house number are still on the list of possible candidates, and these can be differentiated from the others by their Hxx match codes. For more information about match codes, see **Match Codes** on page 170.

One of the following:

Y   Yes, candidates must match the house number. Default.

N   No, candidates do not have to match the house number.

Parameter	Description
Option.MustMatchCity	Specifies whether candidates must match the city. If you do not require exact matches on city, the geocoder searches on the street address matched to the particular postal code, and considers other cities that do not match the name, but do match the postal code.

Y   Yes, candidates must match the city.

N   No, candidates do not have to match the city. Default.

Parameter	Description
Option.MustMatchPostalCode	Specifies whether candidates must match the postal code. If you do not require exact match on postal codes, the geocoder searches a wider area for a match. While this results in slower performance, the match rate is higher because the request does not need to match exactly when it compares match candidates.

Y   Yes, candidates must match the postal code.

N   No, candidates do not have to match the postal code. Default.

Difference Between Match Criteria for U.S. and Non-U.S. Geocoding

The "must match criteria" used in the custom match mode of Geocode US Address work differently than the "close match criteria" in non-U.S. geocoders. For Geocode US Address, the custom match criteria specify which address elements must match the reference database in order for the match to be returned as a candidate. All candidates returned by Geocode US Address will match the elements you specify as long as those elements are available in the reference database. However, in non-U.S. geocoders, the "close match" criteria are used to determine which candidates are close matches and which are non-close matches. Non U.S. geocoders can return both close candidates and non-close candidates, depending on whether you enable the `Option.CloseMatchesOnly` option. In summary, the "must match" criteria used by Geocode US Address automatically limit the candidates returned, whereas the "close match criteria" used by non-U.S. geocoders do not limit the candidates returned.

Extended Match Codes

The Extended Match Codes option enables the return of additional information about any changes in the house number, unit number and unit type fields. In addition, it can indicate whether there was address information that was ignored. The Extended Match Code is only returned for address-level matches (match codes that begin with A, G, H, J, Q, R, S, T or U), in which case a 3rd hex digit is appended to the match code (see **Match Codes** on page 170).

> **Note:** A typical match code contains up to 4 characters: a beginning alpha character followed by 2 or 3 hex digits. The third hex digit is only populated for intersection matches or as part of the Extended Match Code.
>
> For information about the 3rd hex digit values for:
>
> • Intersection matches, see **Definitions for 1st-3rd hex digit match code values** on page 172
> • Extended Match Codes, see **Definitions for Extended Match Code (3rd hex digit values)** on page 173

"Address information ignored" is specified when any of these conditions apply:

• The output address has content in the `AdditionalInputData` field.
• The output address has a second address line (`AddressLine2`).
• The input address is a dual address (two complete addresses in the input address). For example, "4750 Walnut St. P.O Box 50".
• The input last line has extra information that is not a city, state or ZIP Code, and is ignored. For example, "Boulder, CO 80301 USA", where "USA" is ignored when matching.

The table below provides descriptions of the Extended Match Code 3rd hex digit return values.

Input Addressline	Output Addressline	Extended Code	Description
4750 WALNUT ST STE 200	4750 WALNUT ST STE 200	0	Matched on all address information on line, including Unit Number and Unit Type if included.
4750 WALNUT ST C/O JOE SMITH	4750 WALNUT ST	1	Matched on Unit Number and Unit Type if included. Extra information on address line ignored. Extra information not considered for matching moved to `AddressLine2` or `AdditionalInputData` field.
4750 WALNUT ST UNIT 200	4750 WALNUT ST STE 200	2	Matched on Unit Number. Unit Type changed.
4750 WALNUT ST UNIT 200 C/O JOE SMITH	4750 WALNUT ST STE 200	3	Matched on Unit Number. Unit Type changed. Extra information on address line ignored. Extra information not considered for matching moved to `AddressLine2` or `AdditionalInputData` field.
4750 WALNUT ST STE 2-00	4750 WALNUT ST STE 200	4	Unit Number changed or ignored.
4750 WALNUT ST STE 2-00 C/O JOE SMITH	4750 WALNUT ST STE 200	5	Unit Number changed or ignored. Extra information on address line ignored. Extra information not considered for matching moved to `AddressLine2` or `AdditionalInputData` field.
4750 WALNUT ST STE 400	4750 WALNUT ST STE 400	6	Unit Number changed or ignored. Unit Type changed or ignored. In this example, Suite 400 is not valid for the input address, but the address match is not prevented because of an invalid unit number.
4750 WALNUT ST UNIT 2-00 C/O JOE SMITH	4750 WALNUT ST STE 200	7	Unit Number changed or ignored. Unit Type changed or ignored. Extra information on address line ignored. Extra information not considered for matching moved to `AddressLine2` or `AdditionalInputData` field.
47-50 WALNUT ST STE 200	4750 WALNUT ST STE 200	8	Matched on Unit Number and Unit Type if included. House number changed or ignored.
47-50 WALNUT ST STE 200 C/O JOE SMITH	4750 WALNUT ST STE 200	9	Matched on Unit Number and Unit Type if included. House number changed or ignored. Extra information not considered for matching moved to `AddressLine2` or `AdditionalInputData` field.
47-50 WALNUT ST UNIT 200	4750 WALNUT ST STE 200	A	Matched on Unit Number. Unit Type changed. House Number changed or ignored.

Input Addressline	Output Addressline	Extended Code	Description
47-50 WALNUT ST UNIT 200 C/O JOE SMITH	4750 WALNUT ST STE 200	B	Matched on Unit Number. Unit Type changed. House Number changed or ignored. Extra information on address line ignored. Extra information not considered for matching moved to `AddressLine2` or `AdditionalInputData` field.
47-50 WALNUT ST STE 20-0	4750 WALNUT ST STE 200	C	House Number changed or ignored. Unit Number changed or ignored.
47-50 WALNUT ST STE 20-0 C/O JOE SMITH	4750 WALNUT ST STE 200	D	House Number changed or ignored. Unit Number changed or ignored. Extra information on address line ignored. Extra information not considered for matching moved to `AddressLine2` or `AdditionalInputData` field.
47-50 WALNUT ST UNIT 20-0	4750 WALNUT ST STE 200	E	House Number changed or ignored. Unit Number changed or ignored. Unit Type changed or ignored.
47-50 WALNUT ST UNIT 2-00 C/O JOE SMITH	4750 WALNUT ST STE 200	F	House Number changed or ignored. Unit Number changed or ignored. Unit Type changed or ignored. Extra information on address line ignored. Extra information not considered for matching moved to `AddressLine2` or `AdditionalInputData` field.

Search Area

The search area options set the search constraints to use when matching, such as to a city, finance area, or within a specified search radius distance that can be limited to the input state. The search area options can assist in finding a match when the input address contains limited or inaccurate city or ZIP Code information.

The following example illustrates the different match results for an input address that contains an incorrect ZIP Code when setting the Search Area to City, then to Finance Area.

Input Address	Match with Search Area set to City	Match with Search Area set to Finance Area
100 Main St East Aurora, NY 14166 *Input address has an incorrect ZIP Code.*	100 MAIN ST EAST AURORA NY 14052-1633 *Match is made to East Aurora 14052 as there is no candidate in the 14166 input ZIP Code.*	100 MAIN ST DUNKIRK NY 14048-1844 *Same finance area as the input ZIP Code 14166.*

Output Data

GeocodeUSAddress Output Data Options

The following table shows the GeocodeUSAddress options that control which data GeocodeUSAddress returns in the output.

Parameter	Description
Option.OutputRecordType	Specifies optional data to include in the output. Note that GeocodeUSAddress always returns the default data listed in **Default Output** on page 64. The data you select here is returned with the default output data. • **X**—Auxiliary • **B**—Block Address • **C**—Census • **N**—Centerline Projection • **D**—DPV • **Z**—Geo Confidence • **L**—Latitude/Longitude • **E**—Parsed Elements • **P**—Postal Data • **Q**—Qualifiers • **R**—Range • **S**—Segment For a description of the fields in each output group, see **Response** on page 54. If you do not want all of the fields in a record type returned, do not use `Option.OutputRecordType;` instead, use `Option.OutputFields` to specify the desired individual output fields.
Option.OutputFields	Specifies the individual output fields you want returned. List fields with a pipe (`|`) between each field. You can use this option instead of the `Option.OutputRecordType` option to limit the output to those fields that are important to your data needs. By default, these are the address fields returned: `AddressLine1|LastLine|Longitude|Latitude|MatchCode|LocationCode` For a list of all the fields included in each data field, see **Response** on page 54.

Output Format

GeocodeUSAddress Output Format Options

The following table lists the GeocodeUSAddress options that control the format of the output.

Parameter	Description

Option.OutputCasing	Specifies the casing of the output data. One of the following:
	
	M Returns the output in mixed case. Default. For example:
	
	    123 Main St
	    Mytown FL 12345
	
	U Returns the output in upper case. For example:
	
	    123 MAIN ST
	    MYTOWN FL 12345

Option.OutputFormattedOnFail	Specifies whether to normalize addresses that fail to match, and addresses that are unchanged. Normalization formats an address to the USPS guidelines without validating the address. **Y** Perform standardization. Default. **N** Do not perform standardization.
Option.OutputPostalCodeSeparator	Specifies whether to include the dash in full postal code output. **Y** Include the dash. Default. **N** Do not include the dash.
Option.OutputVerbose	Specifies whether to provide an additional description field as output. These fields provide the text equivalent to a field represented by a code. For example, LocationCode returns a code that indicates the accuracy (quality) of the assigned geocode. LocationCode.Description provides the description for the code returned. **Y** Include verbose fields. **N** Do not include verbose fields. Default.

Response

GeocodeUSAddress always returns a default set of output fields that contain the latitude/longitude, standardized address, and result indicators. For information about these fields, see **Default Output** on page 64. You can also choose to include optional categories of output data.

Auxiliary

Auxiliary data output fields contain information about the an auxiliary file match. For more information about using an auxiliary file, see **Auxiliary File Overview** on page 161. GeocodeUSAddress only returns values when matching against an auxiliary file. To include auxiliary data fields in the output, set `Option.OutputRecordType = X`.

Response Element	Max. Field Length with null terminator	Description
AuxiliaryData	301	The user data field in an auxiliary file match. **Note:** GeocodeUSAddress does not process this information. It simply includes the user data contained in the auxiliary file.
MCDCode	6	The Minor Civil Division (MCD) code. A Minor Civil Division is a subdivision of a county, such as a township. There are Minor Civil Divisions in 28 states, the District of Columbia, Puerto Rico, and Island Areas. Minor Civil Divisions are defined by U.S. Census Bureau.
MCDName	41	The Minor Civil Division (MCD) name. A Minor Civil Division is a subdivision of a county, such as a township. There are Minor Civil Divisions in 28 states, the District of Columbia, Puerto Rico, and Island Areas. Minor Civil Divisions are defined by U.S. Census Bureau.

Block Address

The BlockLine<1-6> output fields are populated with the input AddressLine<1-6> fields when a match is not made and **Block Address** output data is enabled.

To include block data in the output, set `Option.OutputRecordType = B`.

If there are any empty lines in the input fields AddressLine1 through AddressLine6, GeocodeUSAddress moves the output lines to the first empty BlockLine output field, eliminating the blank lines.

Response Element	Max. Field Length with null terminator	Description
BlockLine<1-6>	104	When a match is not made, the BlockLine<1-6> output fields are populated with the input AddressLine<1-6> fields. Each BlockLine output field has a maximum field length with null terminator of 104.

Census

Census output fields contain U.S. Census information about the address. To include census data in the output, set `Option.OutputRecordType = C`.

> **Note:** The following Census output fields - `CBSADivisionName`, `CBSAName`, `CSAName` and `USCountyName` - are only returned when you set the option `Option.OutputVerbose=Y`.

Response Element	Max. Field Length with null terminator	Description
APN	46	The assessor's parcel number of the property. The assessor's parcel number is an ID number assigned to a property by the local property tax authority.
BlockSuffix	2	The block suffix for the Census block in which the address is located. A block suffix is a single character assigned to subsections of U.S. Census blocks that are split by a higher-level boundary, such as a municipal boundary. A block suffix is either "A" or "B". For information about U.S. Census block suffixes, see the *Geographic Areas Reference Manual*, available at the U.S. Census Bureau website: **www.census.gov/geo/www/garm.html** Block suffixes are only available if you are using Centrus Enhanced data.

Response Element	Max. Field Length with null terminator	Description
CBSACode	6	The code for the Core Based Statistical Area (CBSA) in which the address is located. A CBSA is a collective term that refers to both metropolitan and micropolitan areas. A metropolitan area has a population of more than 50,000, and a micropolitan area has a population between 10,000 and 49,999. For more information, see *Metropolitan and Micropolitan Statistical Areas* section of the U.S. Census Bureau website: **www.census.gov/population/www/metroareas/metroarea.html**
CBSADivisionCode	6	The code of the Core Based Statistical Area (CBSA) division in which the address is located. A CBSA division is a metropolitan statistical area with a population of at least 2.5 million that has been subdivided to form smaller groupings of counties referred to as "metropolitan divisions." For more information, see *Metropolitan and Micropolitan Statistical Areas* section of the U.S. Census Bureau website: **www.census.gov/population/www/metroareas/metroarea.html**
CBSADivisionName	128	The name of the Core Based Statistical Area (CBSA) division in which the address is located. A CBSA division is a metropolitan statistical area with a population of at least 2.5 million that has been subdivided to form smaller groupings of counties referred to as "metropolitan divisions." For more information, see *Metropolitan and Micropolitan Statistical Areas* section of the U.S. Census Bureau website: **www.census.gov/population/www/metroareas/metroarea.html** **Note:** The CBSA division name is only returned if you set the option `Option.OutputVerbose=Y`.

Response Element	Max. Field Length with null terminator	Description
CBSAMetro	2	Indicates whether the core based statistical area (CBSA) in which the address is located is a metropolitan area or a micropolitan area. One of the following:

Y		Yes, the address is located in a metropolitan statistical area. Metropolitan areas have a population greater than 50,000.
N		No, the address is not located in a metropolitan area. It is located in a micropolitan area. Micropolitan areas have a population between 10,000 and 49,999.
null		There is no data available to determine whether the address is in a metropolitan or micropolitan area.

A CBSA is a collective term that refers to both metropolitan and micropolitan areas. A metropolitan area has a population of more than 50,000, and a micropolitan area has a population between 10,000 and 49,999. For more information, see *Metropolitan and Micropolitan Statistical Areas* section of the U.S. Census Bureau website:

www.census.gov/population/www/metroareas/metroarea.html

Response Element	Max. Field Length with null terminator	Description
CBSAName	128	The name of the core based statistical area (CBSA) in which the address is located.

A CBSA is a collective term that refers to both metropolitan and micropolitan areas. A metropolitan area has a population of more than 50,000, and a micropolitan area has a population between 10,000 and 49,999. For more information, see *Metropolitan and Micropolitan Statistical Areas* section of the U.S. Census Bureau website:

www.census.gov/population/www/metroareas/metroarea.html

> **Note:** The CBSA name is only returned if you set the option `Option.OutputVerbose=Y`.

Response Element	Max. Field Length with null terminator	Description
CensusBlockID	16	The 15-digit identification number of the census block in which the address is located. Census blocks are the smallest geographic area for which the Bureau of the Census collects and tabulates decennial census data. Census blocks are formed by streets, roads, railroads, streams and other bodies of water, other visible physical and cultural features, and the legal boundaries shown on Census Bureau maps. For more information about U.S. Census blocks, see the *Geographic Areas Reference Manual*, available at the U.S. Census Bureau website:

www.census.gov/geo/www/garm.html

The Census block ID is in the format:

`ssccctttttgbbb`

Where:

ss	The two-digit state FIPS code.
ccc	The three-digit county FIPS code.
tttttt	The six-digit Census tract FIPS code.
g	The single-digit block group FIPS code.
bbb	The block FIPS code.

> **Note:** The value in the CensusBlockID field does not contain a period for the Census tract FIPS code. This may deviate from the industry standard.

Response Element	Max. Field Length with null terminator	Description
CensusTract	7	The six-digit ID of the Census tract in which the address is located. Census tracts are small, relatively permanent geographic entities within counties (or the statistical equivalents of counties). Generally, census tracts have between 2,500 and 8,000 residents and boundaries that follow visible features. For more information about U.S. Census tracts, see the *Geographic Areas Reference Manual*, available at the U.S. Census Bureau website:

www.census.gov/geo/www/garm.html

Response Element	Max. Field Length with null terminator	Description
CSACode	4	Denotes the code for a geographic entity that consists of 2 or more adjacent CBSAs with employment interchange measures of at least 15.

Response Element	Max. Field Length with null terminator	Description
CSAName	128	The name of the combined statistical area (CSA) in which the address is located.
		A CSA is a combination of two or more adjacent Core Based Statistical Areas (CBSAs) with a high employment interchange measure. The employment interchange measure is the sum of the percentage of employed residents of the smaller entity who work in the larger entity and the percentage of the employment in the smaller entity that is accounted for by workers who reside in the larger entity. Pairs of CBSAs with employment interchange measures of at least 25% combine automatically. Pairs of CBSAs with employment interchange measures of at least 15%, but less than 25%, may combine if local opinion in both areas favors combination.
		Note: The CSA name is only returned if you set the option `Option.OutputVerbose=Y`.
USCountyName	128	The name of the county or parish in which the address is located.
		Note: The county/parish name is only returned if you set the option `Option.OutputVerbose=Y`.
USFIPSCountyNumber	4	The three-digit FIPS county code of the county in which the address is located.
USFIPSStateCode	3	The two-digit FIPS state code of the state in which the address is located.
USFIPSStateCountyCode	6	The five-digit FIPS code for the state and county in which the address is located.

Centerline

Centerline Output Fields

Centerline output fields contain information specific to a centerline match. For more information about setting up centerline matching, refer to **Geocoding Options** on page 30. To include centerline fields in the output, set `Option.OutputRecordType = N`.

Response Element	Max. Field Length with null terminator	Description
CenterlineBearing	6	The compass direction, in decimal degrees, from the point data match to the street centerline match. The compass direction is measured clockwise from 0 degrees north. For example, if the centerline match is directly north of the point match, the centerline bearing would be 0.
CenterlineBlockLeft	16	The Census FIPS Code that indicates the address is on the left side of the street.
CenterlineBlockRight	16	The Census FIPS Code that indicates the address is on the right side of the street.
CenterlineBlockSuffixLeft	2	The block suffix of the block on the left side of the street. A block suffix is a single character assigned to subsections of U.S. Census blocks that are split by a higher-level boundary, such as a municipal boundary. A block suffix is either "A" or "B". For information about U.S. Census block suffixes, see the *Geographic Areas Reference Manual*, available at the U.S. Census Bureau website: **www.census.gov/geo/www/garm.html** Block suffixes are only available if you are using Centrus Enhanced data.
CenterlineBlockSuffixRight	2	The block suffix of the block on the right side of the street. A block suffix is a single character assigned to subsections of U.S. Census blocks that are split by a higher-level boundary, such as a municipal boundary. A block suffix is either "A" or "B". For information about U.S. Census block suffixes, see the *Geographic Areas Reference Manual*, available at the U.S. Census Bureau website: **www.census.gov/geo/www/garm.html** Block suffixes are only available if you are using Centrus Enhanced data.

Response Element	Max. Field Length with null terminator	Description
CenterlineDataCode	3	Indicates the data used to obtain a centerline match for the address. One of the following:

0	USPS data in either the Centrus Enhanced, Centrus TomTom, or Centrus NAVTEQ database.
1	TIGER data in the Centrus Enhanced database.
2	TomTom data in the Centrus TomTom database.
6	NAVTEQ data in the Centrus NAVTEQ database.
7	TomTom point-level data in the Centrus TomTom Points database.
8	Point-level data from the Centrus Points database.
9	Auxiliary file data.
10	User Dictionary.
11	NAVTEQ point-level data.
12	Master Location data.

For more information about these databases, see **Enterprise Geocoding Databases** on page 5

Response Element	Max. Field Length with null terminator	Description
CenterlineDirection	2	Indicates the order of numbers on a segment for a centerline match.

F	Forward
R	Reversed
B	Both
U	Undetermined

Response Element	Max. Field Length with null terminator	Description
CenterlineDistance	8	Distance, in feet, from the point-level match to the centerline match.
CenterlineHouseNumberHigh	12	The highest address number in the range of addresses on the street segment. For example, if the address range for the street segment is 1000 to 2000, the CenterlineHouseNumberHigh would be 2000.

Response Element	Max. Field Length with null terminator	Description
CenterlineHouseNumberLow	12	The lowest address number in the range of addresses on the street segment. For example, if the address range for the street segment is 1000 to 2000, the CenterlineHouseNumberLow would be 1000.
CenterlineIsAlias	4	Three characters indicating that GeocodeUSAddress located a centerline match by an index alias. The first is an N for normal street match or A for alias match (including buildings, aliases, firms, etc.). The next two characters are:

		01	Basic index (normal address match)
		02	USPS street name alias index
		03	USPS building index
		04	USPS firm name index
		05	Statewide intersection alias match (when using the Usw.gsi or Use.gsi file)
		06	Spatial data street name alias (when using, the Us_pw.gsi, Us_pe.gsi, Us_psw.gsi, or Us_pse.gsi file is required)
		07	Alternate index (when using Zip9.gsu, Zip9e.gsu, and Zip9w.gsu)
		08	LACS[Link]
		09	Auxiliary file match
		10	Centrus Alias index (when using usca.gsi)

Response Element	Max. Field Length with null terminator	Description
CenterlineLatitude	11	A 7-digit number in degrees and calculated to 4 decimal places for a centerline match. This field is only returned if
CenterlineLeadingDirectional	3	The street directional that precedes the street name for a centerline match. For example, the N in 138 N Main Street.
CenterlineLongitude	12	A 7-digit number in degrees and calculated to 4 decimal places (in the format specified) for a centerline match. This field is only returned if

Response Element	Max. Field Length with null terminator	Description
CenterlineParity	2	Indicates which side of the street has odd numbers for a centerline match.

<table>
<tr><td></td><td></td><td>L</td><td>The left side of the street has odd numbers.</td></tr>
<tr><td></td><td></td><td>R</td><td>The right side of the street has odd numbers.</td></tr>
<tr><td></td><td></td><td>B</td><td>Both sides of the street have odd numbers.</td></tr>
<tr><td></td><td></td><td>U</td><td>Undetermined.</td></tr>
</table>

Response Element	Max. Field Length	Description
CenterlineRoadClass	3	The type of road for a centerline match:

<table>
<tr><td></td><td></td><td>1</td><td>Major</td></tr>
<tr><td></td><td></td><td>2</td><td>Minor</td></tr>
</table>

Response Element	Max. Field Length	Description
CenterlineSegmentCode	11	The unique 10-digit street segment ID assigned by the street network data provider.
CenterlineStreetName	41	The name of the street.
CenterlineStreetSuffix	5	The street type of the matched centerline location. For example, AVE in "Washington AVE".
CenterlineTrailingDirectional	3	The street directional that follows the street name. For example, the N in 456 Washington AVE N.

Default Output

Default Output Fields

The geocoder always returns fields that contain the latitude/longitude, standardized address, and result indicators. Result indicators describe how well the geocoder matched the input address to a known address and assigned a location. Result indicators also describe the overall status of a match attempt.

Response Element	Max. Field Length with null terminator	Description
AdditionalInputData	61	This field may contain input data that was not used for matching; for example, input address information that appears after a mail stop or attention designator.
AddressLine1	104	The first line of the address. For example: **1 Global View** Troy, NY 12180-8371
AddressLine2	104	The second line of the address. For example: 4200 Parliament Pl **STE 600** Lanham, MD 20706-1882
City	29	The municipality name.

Response Element	Max. Field Length with null terminator	Description
Confidence	4	Indicates the confidence in the output provided, from 0 to 100. The higher the score, the higher the probability that the match is correct. If the match is exact, the confidence score is 100. For all other matches, the confidence score is calculated based on which portions of the input address had to be changed to obtain a match. Specifically, the confidence score is calculated by subtracting values from 100 as follows:

- If the state is changed to obtain a match:

  - Added the state -3.75
  - No state -7.5

- If the city is changed to obtain a match:

  - Added city -2.5
  - No city -5.0

- If the house number is changed to obtain a match:

  - Added house number -3.75
  - No house number -7.5

- If the street name is changed to obtain a match:

  - Added street name -3.75
  - No street name -7.5

- If the trailing directional is changed to obtain a match:

  - Added trailing directional -1.25
  - No trailing directional -2.5

- If the leading directional is changed to obtain a match:

  - Added leading directional -1.25
  - No leading directional -2.5

- If the street suffix is changed to obtain a match:

  - Added street suffix -1.25
  - No street suffix -2.5

- If the postal code is changed to obtain a match: -11.25

If you have enabled the option to return centroids, the confidence value indicates the type of centroid returned:

- 60 for a street centroid
- 50 for a postal code centroid
- 35 for a city centroid
- 30 for a county centroid
- 25 for a state centroid

Response Element	Max. Field Length with null terminator	Description
Country	25	The name of the country. This field always contains **United States of America**.
FirmName	41	The name of the business if the address is a business address.
LastLine	61	The complete last address line (city, state, and postal code).
Latitude	11	Seven-digit number in degrees and calculated to four decimal places (in the format specified).
LocationCode	5	A value indicating the accuracy (quality) of the assigned geocode. For more information, see **Address Location Codes** on page 178.
Longitude	12	Seven-digit number in degrees and calculated to four decimal places (in the format specified).
MatchCode	5	Indicates the portions of the address that matched to the geocoding directory file. For more information, see **Match Codes** on page 170.
PBKey	13	A unique address identifier that is returned when an address match is made using the Master Location Dataset. The pbKey™ unique identifier is used as a lookup key to a GeoEnrichment dataset, in order to return attribute data for the match.
PostalCode	10	Nine-digit ZIP Code with or without a hyphen.
PostalCode.AddOn	5	Four-digit ZIP Code extension.
PostalCode.Base	6	Five-digit ZIP Code.

Response Element	Max. Field Length with null terminator	Description
ProcessedBy	4	The underlying software that processed the request. EnterpriseGeocoding for GeocodeUSAddress.
StateProvince	3	Two-character state abbreviation.
Status	2	Reports the success or failure of the match attempt

		null	Success
		F	Failure

Status.Code	23	If GeocodeUSAddress could not process the address, this field will show the reason. • Internal System Error • No Geocode Found • Insufficient Input Data

Status.Description	128	If GeocodeUSAddress could not process the address, this field will show a description of the failure.

		Problem + explanation	Returned when Status.Code = Internal System Error.
		Geocoding Failed	Returned when Status.code = No Geocode Found.
		No location returned	Returned when Status.code = No Geocode Found.

Response Element	Max. Field Length with null terminator	Description	
StreetDataType	20	The data set used to geocode the address.	
		USPS	USPS
		TIGER	TIGER
		TOMTOM	TomTom Streets data set
		SANBORN POINT DATA	Sanborn Points data set
		NAVTEQ	NAVTEQ Streets data set
		TOMTOM POINT DATA	TomTom Points data set
		AUXILIARY	Auxiliary file
		CENTRUS POINT DATA	Centrus Points data set
		USER DICTIONARY	User Dictionary
		NAVTEQ POINT DATA	NAVTEQ Points data set
		MASTER LOCATION	Master Location Data
StreetSide	2	Indicates the side of the street the address range occupies. One of the following:	
		L	The address range occupies the left side of the street.
		R	The address range occupies the right side of the street.
		B	The address range occupies both sides of the street.
		U	Undetermined.
USUrbanName	31	Urbanization name. Used for addresses in Puerto Rico.	

DPV

DPV data output fields contain information about a match made using DPV data. GeocodeUSAddress only returns values when matching against DPV data. To include DPV data in the output, set `Option.OutputRecordType = D`.

Response Element	Max. Field Length with null terminator	Description
CMRA	2	Indicates whether the address is for a Commercial Mail Receiving Agent (CMRA). A CMRA is a private company that rents out mailboxes. A customer of a commercial mail receiving agency can receive mail and other deliveries at the street address of the CMRA rather than the customer's own street address. Depending on the agreement between the customer and the CMRA, the CMRA can forward the mail to the customer or hold it for pickup.
		Y Yes, the address is a CMRA.
		N No, the address is not a CMRA.
		null DPV data is not available. DPV data is required to determine if an address is a CMRA.
DPV	2	Indicates whether the address is confirmed to be a deliverable address by USPS Delivery Point Validation (DPV).
		N Nothing confirmed
		Y Everything confirmed (ZIP+4, primary, and secondary)
		S ZIP+4 and primary (house number) confirmed
		D ZIP+4 and primary (house number) confirmed and a default match
		U Non-matched input address to USPS ZIP+4 data, or DPV data not loaded

Response Element	Max. Field Length with null terminator	Description
DPVFootnote	3 per footnote	Contains detailed information about the address. The DPV footnote codes are combined together consecutively.

DPV footnotes include:

- FOOTNOTE1 provides information about the matched DPV records.

  - **AA**—ZIP+4 matched record
  - **A1**—Failure to match a ZIP+4 record
  - **null**—Address not presented to hash table or DPV data not loaded

- FOOTNOTE2 provides information about the matched DPV records.

  - **BB**—All DPV categories matched
  - **CC**—Matched primary/house number, where the secondary/unit number did not match (present but invalid)
  - **M1**—Missing primary/house number
  - **M3**—Invalid primary/house number
  - **N1**—Matched primary/house number, with a missing highrise secondary number
  - **P1**—Missing PS, RR, or HC Box number
  - **P3**—Invalid PS, RR or HC Box number
  - **F1**—All military addresses
  - **G1**—All general delivery addresses
  - **U1**—All unique ZIP Code addresses
  - **null**—Address not presented to hash table or DPV data not loaded

- FOOTNOTE3 provides information about the matched DPV records.

  - **R1**—Matched CMRA, without a present secondary/unit number
  - **RR**—Matched CMRA
  - **null**—Address not presented to hash table or DPV data not loaded

    Note: A unique ZIP Code is a ZIP Code assigned to a company, agency, or entity with sufficient mail volume to have its own ZIP Code.

Geoconfidence

Geoconfidence data output fields contain information about the type of geoconfidence polygon returned. To include geoconfidence fields in the output, set `Option.OutputRecordType` = Z.

Response Element	Max. Field Length with null terminator	Description	
GeoConfidenceCode	13	The value returned in this field indicates which geoconfidence surface type has been returned. The possible values are:	
		INTERSECTION	A geocode point for the intersection of two streets.
		ADDRESS	An array of street segment points representing the street segment where the address is located.
		POINT	If the geocoder was able to match the address using point data, the point geometry where the address is located.
		POSTAL1	A geocode point for the ZIP centroid.
		POSTAL2	An array of points for all street segments in the ZIP + 2 in which the address is located.
		POSTAL3	An array of points for street segments in the ZIP + 4 in which the address is located.
		ERROR	An error has occurred.
StreetSegmentPoints	1024	An array of latitude/longitude values that represent the street segment points.	
		Note: This field contains values only if the `GeoConfidenceCode` field returns a value of `ADDRESS`, `POSTAL2`, or `POSTAL3`.	
GeoConfidenceCentroidLatitude	11	The latitude of the centroid of the geoconfidence polygon.	
GeoConfidenceCentroidLongitude	12	The longitude of the centroid of the geoconfidence polygon.	

Latitude/Longitude

The latitude/longitude output fields contain the geographic coordinates of the address. To include latitude/longitude output fields in the output, set `Option.OutputRecordType = L`.

Response Element	Max. Field Length with null terminator	Description
Elevation	11	The location's elevation in feet above or below sea level.
Latitude	11	The latitude of the address. The latitude is a seven-digit number in degrees, calculated to six decimal places.
Longitude	12	The longitude of the address. The longitude is a seven-digit number in degrees, calculated to six decimal places.

Parsed Elements

The parsed elements output fields contain standard address information as individual units, such as street suffixes (for example AVE, ST, or RD) and leading directionals (for example N and SE). To include parsed elements in the output, set `Option.OutputRecordType = E`.

Response Element	Max. Field Length with null terminator	Description
ApartmentLabel	5	The type of unit, such as apartment, suite, or lot.
ApartmentLabel2	5	The type of unit, such as apartment, suite, or lot, for addresses that contain two units, such as: 123 E Main St APT 3, 4th **Floor** .

Response Element	Max. Field Length with null terminator	Description
ApartmentNumber	12	Apartment number. For example: 123 E Main St APT **3**
ApartmentNumber2	12	Secondary apartment number. For example: 123 E Main St APT 3, **4th** Floor
CrossStreetLeadingDirectional	3	Leading directional, for example: 123 **E** Main St Apt 3 **Note:** Cross street information is only returned if you entered an intersection as an address. For example, entering Pearl and 28th, Boulder, CO returns cross street information. Entering 2800 Pearl, Boulder, CO does not return cross street information.
CrossStreetName	41	Name of cross street. **Note:** Cross street information is only returned if you entered an intersection as an address. For example, entering Pearl and 28th, Boulder, CO returns cross street information. Entering 2800 Pearl, Boulder, CO does not return cross street information.
CrossStreetSuffix	5	Street suffix, for example: 123 E Main **St** Apt 3 **Note:** Cross street information is only returned if you entered an intersection as an address. For example, entering Pearl and 28th, Boulder, CO returns cross street information. Entering 2800 Pearl, Boulder, CO does not return cross street information.
CrossStreetTrailingDirectional	3	Trailing directional, for example: 123 Pennsylvania Ave **NW** **Note:** Cross street information is only returned if you entered an intersection as an address. For example, entering Pearl and 28th, Boulder, CO returns cross street information. Entering 2800 Pearl, Boulder, CO does not return cross street information.
HouseNumber	12	Building number for the address.

Response Element	Max. Field Length with null terminator	Description
HouseNumber2	12	If an address consists of a range of house numbers, this field contains the second house number. The HouseNumber field contains the first number. For example, given this address: 5-7 Maple Ave. The HouseNumber field would contain "5" and the HouseNumber2 field would contain "7".
LeadingDirectional	3	Leading directional, for example: 123 **E** Main St Apt 3
PrivateMailbox	9	Private mailbox. Not returned for multiline input.
PrivateMailbox.Designator	5	Private mailbox description. Not returned for multiline input.
RRHC	2	Rural Route/Highway Contract portion of the address.
StreetName	41	The name of the street, not including any directionals or suffixes. For example, the word "Main" in this address: 123 E **Main** St Apt 3
StreetSuffix	5	The street type of the matched location. For example, AVE for Avenue.
TrailingDirectional	3	Street directional that follows the street name. For example, the N in 456 Washington N.

Postal Data

Postal data output fields contain detailed postal information for the address, such as the preferred city name and the US carrier route. To include postal data fields in the output, set `Option.OutputRecordType = P.`

Response Element	Max. Field Length with null terminator	Description
CityPreferredName	29	The USPS® preferred city name for the ZIP Code of the address.
CityShortName	29	The USPS®-approved abbreviation for the city, if there is one. The USPS® provides abbreviations for city names that are 14 characters long or longer. City abbreviations are 13 characters or less and can be used when there is limited space on the mailing label. If there is no short city name for the city, then the full city name is returned.
CityStateRecordName	29	USPS® city state city name.
DeliveryPointCode	3	Two-digit delivery point barcode.
GovernmentBuilding	2	Indicates if a building is used by the city, state, or federal government.
		A City government building
		B Federal government building
		C State government building
		D Firm only
		E City government building and firm only
		F Federal government building and firm only
		G State government building and firm only
		The values A, B, C, E, F, and G are valid for Alternate records only. The value D is valid for both base and alternate records.
PostalBarCode	7	Six-digit combination of ZIP+4 Code and the delivery point barcode.
PostalCodeClass	2	ZIP Classification code.
		null Standard ZIP Code
		M Military ZIP Code
		P ZIP Code has P.O. boxes only
		U Unique ZIP Code (ZIP Code assigned to a single organization)

Response Element	Max. Field Length with null terminator	Description
PostalCodeUnique	2	Indicates if the ZIP Code is a unique ZIP Code assigned to an individual company or agency.
		Y Unique ZIP name
		null No unique ZIP name
PostalFacility	2	USPS City State Name Facility code.
		A Airport Mail Facility (AMF)
		B Branch
		C Community Post Office (CPO)
		D Area Distribution Center (ADC)
		E Sectional Center Facility (SCF)
		F Delivery Distribution Center (DDC)
		G General Mail Facility (GMF)
		K Bulk Mail Center (BMC)
		M Money Order Unit
		N Non-Postal Community Name, Former Postal Facility, or Place Name
		P Post Office
		S Station
		U Urbanization
USBCCheckDigit	2	Check-digit for delivery point barcode.
USCarrierRouteCode	5	Carrier Route code.

Response Element	Max. Field Length with null terminator	Description
USCarrierRouteSort	2	Indicates if the USPS uses a carrier route sort, and what type of sort the USPS allows.
		A Automation cart allowed, optional cart merging allowed
		B Automation cart allowed, no optional cart merging allowed
		C No automation cart allowed, optional cart merging allowed
		D No automation cart allowed, no optional cart merging allowed
USCityDelivery	2	Indicates if has city-delivery carrier routes.
		Y Has city-delivery carrier routes
		N Does not have city-delivery carrier routes.
USLACS	2	Indicates if LACS^{Link} match occurred.
		Y Matched LACS^{Link} record
		N LACS^{Link} match not found
		F False-positive LACS^{Link} record
		S Secondary information (unit number) removed to make a LACS^{Link} match
		null Records not processed through LACS^{Link}
		For more information, see **Locatable Address Conversion System (LACS)** on page 15.
USLACS.ReturnCode	3	Indicates LACS^{Link} results.
		A Matched LACS^{Link} record
		00 LACS^{Link} match was not found
		09 Matched to highrise default, but noLACS^{Link} conversion
		14 Found LACS^{Link} match, but no LACS^{Link} conversion
		92 Secondary information (unit number) was removed to make a LACS^{Link} match
		null Records not processed through LACS^{Link}
		For more information, see **Locatable Address Conversion System (LACS)** on page 15.

Response Element	Max. Field Length with null terminator	Description
USLOTCode	2	A combination of the 4-digit Line of Travel (LOT) Code and the ascending (A) or descending (D) indicator.

Qualifiers

Qualifier output fields contain qualification information about the match, such as the location code and the match code. To include postal data fields in the output, set `Option.OutputRecordType = Q`.

Response Element	Max. Field Length with null terminator	Description
AddressLineResolved	5	For two-line addresses, indicates which address line was used to obtain the match.
		0 The address could not be matched, or the address matched to multiple addresses.
		1 AddressLine1 was used to obtain the match.
		2 AddressLine2 was used to obtain the match.
		3 Both address lines were used in their original order.
		4 Both address lines were used but the order of the lines was switched to obtain the match.
		5 The input address was a one line address.
CountryLevel	2	The category of the postal data used to validate the address. Always returns "A" for U.S. addresses. Category A means that the postal data is sufficient to validate and correct addresses, including providing missing postal code, city name, state/county name, street address elements, and country name.
DatabaseVersion		The publish date of the USPS data used to validate the address, in the format `Month Year`.

Response Element	Max. Field Length with null terminator	Description
EWSMatch	2	Indicates whether the address was not matched because it is in the Early Warning System (EWS) data.
		Y The address matched to an address in the EWS data so the match was denied.
		null The address did not match to an address in the EWS data.
ExpirationDate	11	Date the database expires, in the format `MM/DD/YYYY`.
Geocoder.MatchCode	4	Indicates how closely the input address matches the candidate address.
		Note: The match codes returned in this field are different from the match codes described in **Match Codes** on page 170. Instead, the match codes returned in this field are taken from a set of match codes that are compatible with all other country geocoders. For more information, see **Result Codes for International Geocoding**.
GeoStanMatchScore	13	Record matching score (for multimatches only).
Intersection	2	Indicates if the input address matched to a cross-street.
		T True
		F False

Response Element	Max. Field Length with null terminator	Description
IsAlias	4	Indicates whether the address matched an index alias in the postal data. Returns three characters. The first is an N for normal street match or A for alias match (including buildings, aliases, firms, etc.). The next two characters are:

01		Basic index (normal address match)
02		USPS street name alias index
03		USPS building index
04		USPS firm name index
05		Statewide intersection alias match
06		Spatial data street name alias
07		Alternate index
08		LACS[Link]
09		Auxiliary file match
10		Centrus Alias index (when using usca.gsi)

Response Element	Max. Field Length with null terminator	Description
IsCloseMatch	2	Indicates whether or not the address was a unique match or if there were candidate addresses.

Y		Yes, the address is a close match. This field always contains "Y" if there is only one match.
N		No, the address is not a close match. The record is a candidate.

Response Element	Max. Field Length with null terminator	Description
LACSAddress	2	Indicates if the input address was converted due to the Locatable Address Conversion System (LACS).

L		Converted
null		Not converted

Response Element	Max. Field Length with null terminator	Description
LocationCode.Description	128	LocationCode converted to text. Only returned when you set the configuration options to return additional descriptions (verbose).
MatchCode.Description	128	MatchCode converted to text. Only returned when you set the configuration options to return additional descriptions (verbose).

Response Element	Max. Field Length with null terminator	Description
RecordType	18	Indicates the record type: • GeneralDelivery • HighRise • FirmRecord • Normal • PostOfficeBox • RRHighwayContract • Geographic (non USPS TIGER match) • Auxiliary (match to an auxiliary file)
RecordType.Default	2	Indicates type of match that occurred for the record type HighRise or RRHighwayContract: **Y** Default match **N** Exact match **U** Not matched
StreetDataCode	3	Indicates the data used to geocode the address. One of the following: **0** USPS data in either the Centrus Enhanced, Centrus TomTom, or Centrus NAVTEQ database. **1** TIGER data in the Centrus Enhanced database. **2** TomTom data in the Centrus TomTom database. **6** NAVTEQ data in the Centrus NAVTEQ database. **7** TomTom point-level data in the Centrus TomTom Points database. **8** Point-level data from the Centrus Points database. **9** Auxiliary file data. **10** User Dictionary. **11** Point-level data from NAVTEQ Points database. **12** Master Location Data (MLD). For more information about these databases, see **Enterprise Geocoding Databases** on page 5.

Response Element	Max. Field Length with null terminator	Description
StreetDataType	20	Indicates the data initially used for the match attempt. Note that the output field StreetDataCode shows which data was actually used to obtain the match.
		The data indicated in StreetDataType may be different from that in StreetDataCode if a match cannot be made in the initial match attempt. For example, if a points database is loaded, GeocodeUSAddress will first attempt a match to the point data because this is the most accurate type of match. If a point-level match cannot be made, GeocodeUSAddress will attempt to match to street data. If the match is made using street data, then the SreetDataType would indicate the point-level data and the StreetDataCode would indicate the street data.
		For more information, see **How GeocodeUSAddress Processes Addresses** on page 26.

Range

Range output fields contain information about the street range, such as the high and low unit numbers. To include range data fields in the output, set `Option.OutputRecordType = R`.

Response Element	Max. Field Length with null terminator	Description
Alternate	2	USPS code that specifies whether a record is a base or alternate record.
		B Base record. Base records can represent a range of addresses or an individual address, such as a firm record.
		A Alternate record. Alternate records are individual delivery points.
HouseNumberHigh	12	The highest house number in the range.
HouseNumberLow	12	The lowest house number in the range.

Response Element	Max. Field Length with null terminator	Description
HouseNumberParity	2	Indicates if the house number range contains even or odd numbers.
		E Even
		O Odd
		B Both
PostalCodeExtensionHigh	5	The highest four-digit ZIP Code extension in the range. The ZIP Code extension is the four digits at the end of the ZIP Code. For example: 60510-**1134**.
PostalCodeExtensionLow	5	The lowest four-digit ZIP Code extension in the range. The ZIP Code extension is the four digits at the end of the ZIP Code. For example: 60510-**1134**.
UnitNumberHigh	12	The highest unit number in the range.
UnitNumberLow	12	The lowest unit number in the range.
UnitNumberParity	2	Indicates if the unit number range contains even or odd numbers.
		E Even
		O Odd
		B Both

Segment

Segment output fields contain information about the street segment identified by the data provider. To include segment data fields in the output, set `Option.OutputRecordType = S`.

Response Element	Max. Field Length with null terminator	Description
BlockLeft	16	The Census FIPS Code that indicates the address is on the left side of the street.
BlockRight	16	The Census FIPS Code that indicates the address is on the right side of the street.
BlockSuffixLeft	2	The block suffix of the block on the left side of the street. A block suffix is a single character assigned to subsections of U.S. Census blocks that are split by a higher-level boundary, such as a municipal boundary. A block suffix is either "A" or "B". For information about U.S. Census block suffixes, see the *Geographic Areas Reference Manual*, available at the U.S. Census Bureau website: **www.census.gov/geo/www/garm.html** Block suffixes are only available if you are using Centrus Enhanced data.
BlockSuffixRight	2	The block suffix of the block on the right side of the street. A block suffix is a single character assigned to subsections of U.S. Census blocks that are split by a higher-level boundary, such as a municipal boundary. A block suffix is either "A" or "B". For information about U.S. Census block suffixes, see the *Geographic Areas Reference Manual*, available at the U.S. Census Bureau website: **www.census.gov/geo/www/garm.html** Block suffixes are only available if you are using Centrus Enhanced data.
PointCode	11	Unique point ID of the matched record when matched to point-level data. *Blank* if the matched record is not from point-level data.
RoadClass	3	The type of road: **1** Major **2** Minor

Response Element	Max. Field Length with null terminator	Description
SegmentCode	11	The unique 10-digit street segment ID assigned by the street network data provider.
SegmentDirection	2	Indicates the order of numbers on a segment.
		F Forward
		R Reversed
		B Both
		U Undetermined
SegmentHouseNumberHigh	12	The highest house number in the segment.
SegmentHouseNumberLow	12	The lowest house number in the segment.
SegmentLength		The length, in feet, of a block segment. *This field is deprecated.*
SegmentParity	2	Indicates which side of the street has odd numbers.
		L Left side of the street
		R Right side of the street
		B Both sides of the street
		U Undetermined

Response Element	Max. Field Length with null terminator	Description
StreetSide.NAVTEQ	2	Indicates which side of the street the address is located on. The value in this field is determined by using the NAVTEQ reference nodes for the street segment. A street segment represents part of a street. Each segment has a node at each end: the reference node and the non-reference node. The reference node is the node with the lower latitude (southernmost). If the latitudes of both nodes are identical, the reference node is the end node with the lower longitude (westernmost). The street side corresponds to the street sides you would see if you were standing at the reference node and looking at the non-reference node.

One of the following:

L	The address is on the left side of the street.
R	The address is on the right side of the street.
B	The address occupies both sides of the street.
U	The street side is unknown.
null	NAVTEQ data was not used, or segment output data was not selected, or the address did not match a street segment (for example, the address was geocoded to a centroid).

3 - ReverseAPNLookup

ReverseAPNLookup allows you to look up an address using:

- An Assessor's Parcel Number (APN). An APN is an ID number assigned to a piece of land by a county assessor. An APN is unique only within a county.
- A FIPS county code. A Federal Information Processing Standard (FIPS) code is an ID number assigned to a county by the U.S. Federal government.
- A FIPS state code. A FIPS state code is an ID number assigned to each state by the U.S. Federal government.

These three pieces of information, used together, can uniquely identify a specific parcel. You must use all three pieces of information to perform a lookup using ReverseAPNLookup.

> **Note:** ReverseAPNLookup only works for U.S. addresses for which APN data is available. See the coverage map included with the points database for more information.

ReverseAPNLookup is part of the Enterprise Geocoding Module. For more information about the Enterprise Geocoding Module, see **Enterprise Geocoding Module** on page 4.

In this section

ReverseAPNLookup

Resource URL

JSON endpoint:

```
http://server:port/rest/ReverseAPNLookup/results.json
```

XML endpoint:

```
http://server:port/rest/ReverseAPNLookup/results.xml
```

Example with JSON Response

The following example requests a JSON response:

```
http://myserver:8080/rest/ReverseAPNLookup/results.json?
Data.APN=12-21-405-011&Data.USFIPSStateCode=17&Data.USFIPSCountyNumber=089
```

The JSON returned by this request would be:

```
{"output_port": [{
    "USFIPSCountyNumber": "089",
    "USFIPSStateCode": "17",
    "Latitude": "41.846399",
    "Longitude": "-88.325771",
    "StreetSide": "L",
    "Distance": "",
    "PercentGeocode": "",
    "FirmName": "",
    "AddressLine1": "1012 Main St",
    "AddressLine2": "",
    "LastLine": "Batavia, IL  60510-1602",
    "AdditionalInputData": "",
    "City": "Batavia",
    "StateProvince": "IL",
    "PostalCode.Base": "60510",
    "PostalCode.AddOn": "1602",
    "PostalCode": "60510-1602",
    "USUrbanName": "",
    "Country": "United States of America",
    "LocationCode": "AP02",
    "MatchCode": "P01",
    "StreetDataType": "CENTRUS POINT DATA",
    "APN": "12-21-405-011",
    "Confidence": "95",
    "ProcessedBy": "KGL",
```

```
    "Status": "",
    "Status.Code": "",
    "Status.Description": "",
    "user_fields": []
}]}
```

Example with XML Response

The following example requests an XML response:

```
http://myserver:8080/rest/ReverseAPNLookup/results.xml?
Data.APN=12-21-405-011&Data.USFIPSStateCode=17&Data.USFIPSCountyNumber=089
```

The XML returned by this request would be:

```
<xml.ReverseAPNLookupResponse
xmlns="http://www.pb.com/spectrum/services/ReverseAPNLookup">
    <output_port>
        <Address>
            <USFIPSCountyNumber>089</USFIPSCountyNumber>
            <USFIPSStateCode>17</USFIPSStateCode>
            <Latitude>41.846399</Latitude>
            <Longitude>-88.325771</Longitude>
            <StreetSide>L</StreetSide>
            <Distance/>
            <PercentGeocode/>
            <FirmName/>
            <AddressLine1>1012 Main St</AddressLine1>
            <AddressLine2/>
            <LastLine>Batavia, IL  60510-1602</LastLine>
            <AdditionalInputData/>
            <City>Batavia</City>
            <StateProvince>IL</StateProvince>
            <PostalCode.Base>60510</PostalCode.Base>
            <PostalCode.AddOn>1602</PostalCode.AddOn>
            <PostalCode>60510-1602</PostalCode>
            <USUrbanName/>
            <Country>United States of America</Country>
            <LocationCode>AP02</LocationCode>
            <MatchCode>P01</MatchCode>
            <StreetDataType>CENTRUS POINT DATA</StreetDataType>
            <APN>12-21-405-011</APN>
            <Confidence>95</Confidence>
            <ProcessedBy>KGL</ProcessedBy>
            <Status/>
            <Status.Code/>
            <Status.Description/>
            <user_fields/>
        </Address>
    </output_port>
</xml.ReverseAPNLookupResponse>
```

Request

Input

Reverse APNLookup Input Data

ReverseAPNLookup takes an APN, FIPS county code, and FIPS state code as input. The following table provides information about the format and layout of the input.

Parameter	Format	Description
Data.APN	String [45]	The assessor's parcel number (APN) for the property you want to look up.
Data.InputKeyValue	String	User-defined data, such as a record ID or source code.
Data.USFIPSCountyNumber	String [5]	The FIPS county code for the county in which the property resides.
Data.USFIPSStateCode	String [2]	The FIPS state code for the state in which the property resides.

Input Parameters

The following table lists the options that control ReverseAPNLookup processing.

Note: As the Enterprise Geocoding Module transitions its administrative tasks to a web-based Management Console, labels for the options may use different wording than what you see in Enterprise Designer. There is no difference in behavior.

Parameter	Description
Option.Dataset	Specifies the database to use to look up the parcel. Use the database name specified in Management Console's Spectrum Databases page.
Option.LatLongFormat	Specifies the format for returned latitude/longitude. **Decimal** — (90.000000-180.000000) Default. **Integer** — (90000000-180000000)
Option.RetrieveElevation	Specifies whether ReverseAPNLookup returns the elevation of the address. Elevation is the distance above or below sea level of a given location. The elevation is returned in the **Elevation** output field, which is part of the Latitude/Longitude output group. **Note:** This option requires that you have licensed and installed the Centrus Premium Points database. Elevation data is not available for all addresses. See the coverage map included with the points database. **Y** — Return elevation data. **N** — Do not return elevation data. Default.
Option.OutputCasing	Specifies the casing of the output data. **M** — Returns the output in mixed case. Default. For example: 123 Main St Mytown FL 12345 **U** — Returns the output in upper case. For example: 123 MAIN ST MYTOWN FL 12345
Option.OutputVerbose	Specifies whether ReverseAPNLookup provides an additional description field as output. This field provides the text equivalent to a field represented by a code. For example, `LocationCode` returns a code that indicates the accuracy (quality) of the assigned geocode. `LocationCode.Description` provides the description for the code returned. **Y** — Include verbose fields. **N** — Do not include verbose fields. Default.

Parameter	Description
Option.OutputRecordType	Specifies optional data to include in the output. Note that ReverseAPNLookup always returns the default data listed in **Default Output** on page 96. The data you select here is returned with the default output data. • **C**—Census • **L**—Latitude/Longitude • **E**—Parsed Elements • **Q**—Qualifiers • **R**—Range • **S**—Segment For a list of fields included in each record type, see **Response** on page 93. If you do not want all of the fields in a group returned, do not select the group, and instead list only those fields you want returned in `Option.OutputFields`.
Option.OutputFields	Indicates the individual output fields you want returned. List fields with a pipe (|) between each field. You can use this field instead of the `Option.OutputRecordType` field to limit the output to the specific fields you want. Default list: AddressLine1|LastLine|Longitude|Latitude|MatchCode|LocationCode

Response

ReverseAPNLookup always returns a default set of output fields that contain the address, geocode, and result indicators. For information about these fields, see **Default Output** on page 96. You can also choose to include optional categories of output data.

Census

Census output fields contain U.S. Census information about the address. To include census data in the output, set `Option.OutputRecordType = C`.

Note: The following Census output fields - `CBSADivisionName, CBSAName, CSAName` and `USCountyName` - are only returned when you set the option `Option.OutputVerbose`=Y.

Response Element	Max. Field Length with null terminator	Description
BlockSuffix	2	Single character block suffix for split Census blocks. Returns A or B. Only available in Centrus Enhanced data.
CBSACode	6	Indicates Core Based Statistical Area (CBSA).
CBSADivisionCode	6	Denotes a subdivision of a CBSA.
CBSADivisionName	128	The name of the Core Based Statistical Area (CBSA) division in which the address is located. A CBSA division is a metropolitan statistical area with a population of at least 2.5 million that has been subdivided to form smaller groupings of counties referred to as "metropolitan divisions." For more information, see *Metropolitan and Micropolitan Statistical Areas* section of the U.S. Census Bureau website: **www.census.gov/population/www/metroareas/metroarea.html** Note: The CBSA division name is only returned if you set the option `Option.OutputVerbose=Y.`
CBSAMetro	2	Metropolitan Statistical Area. Valid values include: **Y** Metro statistical area **N** Micro statistical area **null** Data unavailable
CBSAName	128	The name of the core based statistical area (CBSA) in which the address is located. A CBSA is a collective term that refers to both metropolitan and micropolitan areas. A metropolitan area has a population of more than 50,000, and a micropolitan area has a population between 10,000 and 49,999. For more information, see *Metropolitan and Micropolitan Statistical Areas* section of the U.S. Census Bureau website: **www.census.gov/population/www/metroareas/metroarea.html** Note: The CBSA name is only returned if you set the option `Option.OutputVerbose=Y.`

Response Element	Max. Field Length with null terminator	Description
CensusBlockID	16	The ID of the Census Federal Information Processing Standard (FIPS) code.
CensusTract	7	Six digits extracted from the CensusBlockID.
CSACode	4	Denotes the code for a geographic entity that consists of 2 or more adjacent CBSAs with employment interchange measures of at least 15.
CSAName	128	The name of the combined statistical area (CSA) in which the address is located. A CSA is a combination of two or more adjacent Core Based Statistical Areas (CBSAs) with a high employment interchange measure. The employment interchange measure is the sum of the percentage of employed residents of the smaller entity who work in the larger entity and the percentage of the employment in the smaller entity that is accounted for by workers who reside in the larger entity. Pairs of CBSAs with employment interchange measures of at least 25% combine automatically. Pairs of CBSAs with employment interchange measures of at least 15%, but less than 25%, may combine if local opinion in both areas favors combination. **Note:** The CSA name is only returned if you set the option `Option.OutputVerbose=Y`.
USCountyName	128	The name of the county or parish in which the address is located. **Note:** The county/parish name is only returned if you set the option `Option.OutputVerbose=Y`.
USFIPSStateCountyCode	6	Five-digit FIPS code for state and county extracted from the CensusBlockID.

Default Output

Default Output Fields

ReverseAPNLookup always returns the address, geocode, and result indicators.

Response Element	Max. Field Length with null terminator	Description
AdditionalInputData	61	This field may contain input data that was not used for matching; for example, input address information that appears after a mail stop or attention designator.
AddressLine1	104	First line of the address.
AddressLine2	104	Second line of the address.
APN	46	The Assessor's Parcel Number that was specified in the input.
City	29	Municipality name.
Confidence	4	Indicates the confidence in the output provided. The range is from 0 (zero) to 100, with 0 being no match and 100 being an exact match.
Country	25	The name of the country. Since ReverseAPNLookup only works for U.S. locations, this field will always contain **United States of America**.
Distance	8	The distance, in feet, from the input geocode to nearest matched address or intersection.
Elevation	11	The distance in feet above or below sea level of the parcel.
FirmName	41	Name of the company.

Response Element	Max. Field Length with null terminator	Description
LastLine	61	Complete last address line (municipality, state, and postal code).
Latitude	11	Seven-digit number in degrees and calculated to 4 decimal places (in the format specified).
LocationCode	5	Indicates the accuracy (quality) of the assigned geocode. For more information, see **Address Location Codes** on page 178.
Longitude	12	Seven-digit number in degrees and calculated to 4 decimal places (in the format specified).
MatchCode	5	Indicates the portions of the address that matched to the directory file. For more information, see **Match Codes** on page 170.
PercentGeocode	6	The percent along the street segment that matches the geocode. For example, if the returned geocode falls 1/3 along the way of the entire street segment, the percent is 33.000. **Note:** This value is always 0.0 for matches to point-level data and intersections.
PostalCode	10	Nine-digit ZIP Code with or without a hyphen.
PostalCode.AddOn	5	Four-digit ZIP Code extension.
PostalCode.Base	6	Five-digit ZIP Code.
ProcessedBy	4	The feature code for the stage that processed the request. The value is EnterpriseGeocoding for ReverseAPNLookup.
StateProvince	3	Two-character state abbreviation.

Response Element	Max. Field Length with null terminator	Description	
Status	2	Reports the success or failure of the match attempt	
		null	Success
		F	Failure
Status.Code	23	Reason for failure: • Internal System Error • No Address Found • Insufficient Input Data	
Status.Description	128	Description of the problem:	
		Problem + explanation	Returned when Status.Code = Internal System Error.
		Geocoding Failed	Returned when Status.code = No Address Found.
		No location returned	Returned when Status.code = No Address Found.
StreetDataType	20	The data set that ReverseAPNLookup attempted to match against.	
		USPS	USPS
		TIGER	TIGER
		TOMTOM	TomTom Streets data set
		SANBORN POINT DATA	Sanborn Points data set
		NAVTEQ	NAVTEQ Streets data set
		TOMTOM POINT DATA	TomTom Points data set
		AUXILIARY	Auxiliary file
		CENTRUS POINT DATA	Centrus Points data set
		USER DICTIONARY	User Dictionary
		NAVTEQ POINT DATA	NAVTEQ Points data set
		MASTER LOCATION	Master Location Data

Response Element	Max. Field Length with null terminator	Description
StreetSide	2	Indicates the side of the street the range occupies. One of the following:
		L The range occupies the left side of the street.
		R The range occupies the right side of the street.
		B The range occupies both sides of the street.
		U Undetermined.
USFIPSCountyNumber	4	Three-digit FIPS county code specified in the input.
USFIPSStateCode	3	Two-digit FIPS state code specified in the input.
USUrbanName	6	USPS® urbanization name. Puerto Rican addresses only.

Latitude/Longitude

The latitude/longitude output fields contain the geographic coordinates of the location. To include latitude/longitude output fields in the output, set `Option.OutputRecordType = L`.

Response Element	Max. Field Length with null terminator	Description
Latitude	11	7-digit number in degrees and calculated to 4 decimal places (in the format specified).
Longitude	12	7-digit number in degrees and calculated to 4 decimal places (in the format specified).

Parsed Elements

The Parsed Elements output fields contain standard address information as individual units, such as street suffixes (AVE) and leading directionals (N and SE). To include parsed elements in the output, set `Option.OutputRecordType = E`.

Response Element	Max. Field Length with null terminator	Description
ApartmentLabel	5	Apartment designator (such as STE or APT), for example: 123 E Main St. APT 3
ApartmentLabel2	5	Secondary apartment designator, for example: 123 E Main St. APT 3, 4th Floor
ApartmentNumber	12	Apartment number, for example: 123 E Main St. APT 3
ApartmentNumber2	12	Secondary apartment number, for example: 123 E Main St. APT 3, 4th Floor
City	29	Municipality name.
CrossStreetLeadingDirectional	3	Leading directional, for example: 123 E Main St. Apt 3
CrossStreetName[1]	41	Cross street name, for example: 123 E Main St. Apt 3
CrossStreetSuffix	5	Cross street suffix, for example: 123 E Main St. Apt 3

---

[1] ReverseAPNLookup only returns Cross street outputs if you entered an intersection as an address. For example, entering Pearl and 28th, Boulder, CO returns cross street information. Entering 2800 Pearl, Boulder, CO does NOT return cross street information.

Response Element	Max. Field Length with null terminator	Description
CrossStreetTrailingDirectional	3	Cross street trailing directional, for example: 123 Pennsylvania Ave NW
HouseNumber	12	Building number, for example: 123 E Main St. Apt 3 **Note:** This is an approximate building number based on the APN, FIPS county code, and FIPS state code provided. This approximate address may not exist or may not accept mail delivery.
LeadingDirectional	3	Leading directional, for example: 123 E Main St. Apt 3
PrivateMailbox	9	Private mailbox indicator. Not output for multiline input.
PrivateMailbox.Designator	5	The type of private mailbox. Possible values include: • Standard • Non-Standard
RRHC	2	Rural Route/Highway Contract indicator.
StreetName	41	Street name, for example: 123 E Main St. Apt 3
StreetSuffix	5	Street suffix, for example: 123 E Main St. Apt 3
TrailingDirectional	3	Trailing directional, for example: 123 Pennsylvania Ave NW

Qualifiers

The qualifiers output fields contain qualification information about the match, such as the location code and the match code. To include qualifier output fields in the output, set `Option.OutputRecordType = Q.`

Response Element	Max. Field Length with null terminator	Description
CountryLevel	2	The category of postal data available. Always returns A in ReverseAPNLookup—Validates, corrects, and provides missing postal code, city name, state/county name, street address elements, and country name.
DatabaseVersion	15	USPS publish date, in the format Month Year.
EWSMatch	2	Indicates if ReverseAPNLookup denied a match because of Early Warning System (EWS) data.
		Y EWS denied a match.
		null EWS did not deny a match.
		For more information about EWS, see **Early Warning System (EWS)** on page 16.
ExpirationDate	11	Date the database expires, in the format MM/DD/YYYY.
GeoStanMatchScore	13	Record matching score (for multimatches only).
Intersection	2	Indicates if ReverseAPNLookup found a cross-street match.
		T True, a cross-street match was found.
		F False, a cross-street match was not found.

Response Element	Max. Field Length with null terminator	Description
IsAlias	4	ReverseAPNLookup located a matched record by an index alias. Returns 3 characters. The first is an **N** for normal street match or **A** for alias match (including buildings, aliases, firms, etc.). The next 2 characters are:
		01 Basic index (normal address match)
		02 USPS street name alias index
		03 USPS building index
		04 USPS firm name index
		05 Statewide intersection alias match (when using the Usw.gsi or Use.gsi file)
		06 Spatial data street name alias (when using the Us_pw.gsi, Us_pe.gsi, Us_psw.gsi, or Us_pse.gsi file)
		07 Alternate index (when using Zip9.gsu, Zip9e.gsu, and Zip9w.gsu)
		08 LACS^{Link}
		09 Auxiliary file match
		10 Centrus Alias index (when using usca.gsi)
LACSAddress	2	Indicates if ReverseAPNLookup converted an address due to the Locatable Address Conversion System (LACS).
		L Converted
		null Not converted
		For more information about LACS, see **Locatable Address Conversion System (LACS)** on page 15.
LocationCode.Description	128	LocationCode converted to text. Only returned when you set the configuration options to return additional descriptions (verbose).
MatchCode.Description	128	MatchCode converted to text. Only returned when you set the configuration options to return additional descriptions (verbose).

Response Element	Max. Field Length with null terminator	Description
RecordType	18	Indicates the record type: • GeneralDelivery • HighRise • FirmRecord • Normal • PostOfficeBox • RRHighwayContract
RecordType.Default	2	Indicates type of match that occurred for the record type HighRise or RRHighwayContract: **Y** Default match **N** Exact match **U** Not matched
StreetDataCode	3	Indicates the data used to obtain a match. **0** USPS data in either the Centrus Enhanced, Centrus TomTom, or Centrus NAVTEQ database. **1** TIGER data in the Centrus Enhanced database. **2** TomTom data in the Cenrus TomTom database. **6** NAVTEQ data in the Centrus NAVTEQ database. **7** TomTom point-level data in the Centrus TomTom Points database. **8** Point-level data from the Centrus Points database. **9** Auxiliary file data **10** User Dictionary. **11** Point-level data from NAVTEQ Points database. **12** Master Location Data (MLD). For more information about these databases, see **Enterprise Geocoding Databases** on page 5.

Range

The range output fields contain information about the street range, such as the high and low unit numbers. To include range data fields in the output, set `Option.OutputRecordType = R`.

Response Element	Max. Field Length with null terminator	Description
Alternate	2	USPS code that specifies whether a record is a base or alternate record. **B** Base record. Base records can represent a range of addresses or an individual address, such as a firm record. **A** Alternate record. Alternate records are individual delivery points.
HouseNumberHigh	12	House number high.
HouseNumberLow	12	House number low.
HouseNumberParity	2	Indicates if the house number range contains even or odd numbers. **E** Even **O** Odd **B** Both
PostalCodeExtensionHigh	5	4-digit ZIP Code extension high.
PostalCodeExtensionLow	5	4-digit Zip Code extension low.
UnitNumberHigh	12	Unit number high.
UnitNumberLow	12	Unit number low.

Response Element	Max. Field Length with null terminator	Description
UnitNumberParity	2	Indicates if the unit number range contains even or odd numbers.

		E	Even
		O	Odd
		B	Both

Segment

The segment output fields contain information about the street segment identified by the data provider. To include segment data fields in the output, set `Option.OutputRecordType = S`.

Response Element	Max. Field Length with null terminator	Description
BlockLeft	16	Provides the Census FIPS Code that indicates the address is on the left side of the street.
BlockRight	16	Provides the Census FIPS Code that indicates the address is on the right side of the street.
BlockSuffixLeft	2	Current left Block suffix for Census 2010 Geography. Returns A or B. Only available in Centrus Enhanced data.
BlockSuffixRight	2	Current right Block suffix for Census 2010 Geography. Returns A or B. Only available in Centrus Enhanced data.
RoadClass	3	The type of road:

		1	The road is a major road.
		2	The road is a minor road.

Response Element	Max. Field Length with null terminator	Description
PointCode	11	Unique point ID assigned by the data provider. This field is blank if the matched record is not from point-level data.
SegmentCode	11	Unique 10-digit segment ID assigned by the street network provider.
SegmentDirection	2	Indicates the order of numbers on a segment. **F** Forward **R** Reversed **B** Both **U** Undetermined
SegmentHouseNumberHigh	12	A high range number in the segment.
SegmentHouseNumberLow	12	A low range number in the segment.
SegmentLength		The length, in feet, of a block segment. *This field is deprecated.*
SegmentParity	2	Indicates which side of the street has odd numbers. **L** Left side of the street **R** Right side of the street **B** Both sides of the street **U** Undetermined

4 -

ReverseGeocodeUSLocation

ReverseGeocodeUSLocation takes a latitude and longitude point as input and returns the address that is the best match for that point.

ReverseGeocodeUSLocation is part of the Enterprise Geocoding Module.

In this section

ReverseGeocodeUSLocation

ReverseGeocodeUSLocation takes a latitude and longitude point as input and returns the address that is the best match for that point. For example, you could enter the following information:

`Longitude:` -105239771 `Latitude:` 40018912 `Search Distance:` 150 feet

This input would result in the following output:

```
4750 WALNUT ST BOULDER, CO 80301-2538
MatchCode = NS0
LocCode = AS0
Lon = -105239773
Lat = 40018911
Distances:
Search = 150
Offset = 50
Squeeze = 50
Nearest = 50.0
Pct Geocode = 94.0
SegID = 472881795
PtID = GDT
Block = 080130122032066
County Name = BOULDER COUNTY
DPBC = 50
```

> **Note:** The address returned is an approximate address based on the latitude and longitude provided. This approximate address may not exist or may not accept mail delivery.

ReverseGeocodeUSLocation processes geocodes in the following order:

1. ReverseGeocodeUSLocation defines a small rectangle based on your input geocode and search distance.
2. ReverseGeocodeUSLocation computes the distance between each street segment and the input location.
3. If one segment is closest, ReverseGeocodeUSLocation finds the offset and interpolated percentage (using the squeeze factor) and the street side. It then computes an approximate house number based on this information.

   If there is more than one segment that is equally close to the input location, a multi-match occurs. ReverseGeocodeUSLocation returns the information for all of the equally close segments so that you can determine which segment is applicable.

4. ReverseGeocodeUSLocation returns the address information, including the segment range, the approximate house number, and the parity of the range along with other standard address information.

    Note: Although many of the standard address matching outputs apply to the reverse geocoding option, several outputs are unavailable (such as LACS[Link] information and unit numbers). ReverseGeocodeUSLocation returns these outputs as blank. ReverseGeocodeUSLocation also has outputs specific to reverse geocode processing, such as specific match codes and the distance from the input location to the matched segment.

To use ReverseGeocodeUSLocation, you need additional data files, called GSX files. There is an option to install these files when you install the geocoding database. The GSX files must be installed the GSX subdirectory of the geocoding database. If you install the Centrus Enhanced Points, Centrus Premium Points, or Centrus TomTom Points database, you must recreate the GSX files. Consult with Pitney Bowes Software Technical Support if you need more information about GSX files.

Resource URL

JSON endpoint:

```
http://server:port/rest/ReverseGeocodeUSLocation/results.json
```

XML endpoint:

```
http://server:port/rest/ReverseGeocodeUSLocation/results.xml
```

Example with JSON Response

The following example requests a JSON response:

```
http://myserver:8080/rest/ReverseGeocodeUSLocation/results.json?
Data.Longitude=-73549161&Data.Latitude=41071067
```

The JSON returned by this request would be:

```
{"output_port": [{
    "Latitude": "41.071067",
    "Longitude": "-73.549161",
    "StreetSide": "R",
    "Distance": "50.4",
    "PercentGeocode": "50.0",
    "FirmName": "",
    "AddressLine1": "2995 Summer St",
    "AddressLine2": "",
    "LastLine": "Stamford, CT  06905",
    "AdditionalInputData": "",
    "City": "Stamford",
    "StateProvince": "CT",
    "PostalCode.Base": "06905",
    "PostalCode.AddOn": "",
```

```
    "PostalCode": "06905",
    "USUrbanName": "",
    "Country": "United States of America",
    "LocationCode": "AS0",
    "MatchCode": "NS1",
    "StreetDataType": "TOMTOM",
    "Confidence": "100.0",
    "ProcessedBy": "KGL",
    "PBKey": "",
    "Status": "",
    "Status.Code": "",
    "Status.Description": "",
    "user_fields": []
}]}
```

Example with XML Response

The following example requests an XML response:

```
http://myserver:8080/rest/ReverseGeocodeUSLocation/results.xml?
Data.Latitude=41071067&Data.Longitude=-73549161
```

The XML returned by this request would be:

```
<xml.ReverseGeocodeUSLocationResponse
xmlns="http://www.pb.com/spectrum/services/ReverseGeocodeUSLocation">
    <output_port>
        <Address>
            <Latitude>41.071067</Latitude>
            <Longitude>-73.549161</Longitude>
            <StreetSide>R</StreetSide>
            <Distance>50.4</Distance>
            <PercentGeocode>50.0</PercentGeocode>
            <FirmName/>
            <AddressLine1>2995 Summer St</AddressLine1>
            <AddressLine2/>
            <LastLine>Stamford, CT  06905</LastLine>
            <AdditionalInputData/>
            <City>Stamford</City>
            <StateProvince>CT</StateProvince>
            <PostalCode.Base>06905</PostalCode.Base>
            <PostalCode.AddOn/>
            <PostalCode>06905</PostalCode>
            <USUrbanName/>
            <Country>United States of America</Country>
            <LocationCode>AS0</LocationCode>
            <MatchCode>NS1</MatchCode>
            <StreetDataType>TOMTOM</StreetDataType>
            <Confidence>100.0</Confidence>
            <ProcessedBy>KGL</ProcessedBy>
            <PBKey/>
            <Status/>
```

```
            <Status.Code/>
            <Status.Description/>
            <user_fields/>
        </Address>
    </output_port>
</xml.ReverseGeocodeUSLocationResponse>
```

Request

Input

ReverseGeocodeUSLocation Input Data

ReverseGeocodeUSLocation takes longitude and latitude information as input. The following table provides information about the format and layout of the input.

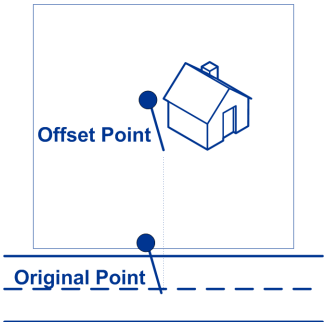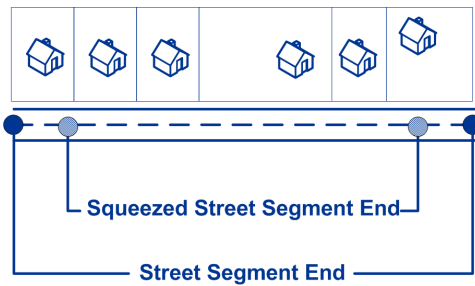| Parameter	Format	Description
Data.Latitude	String	Latitude of the point for which you want address information returned. Specify latitude in millionths of decimal degrees.
Data.Longitude	String	Longitude of the point for which you want address information returned. Specify longitude in millionths of decimal degrees.

Input Parameters

Configuration Options

ReverseGeocodeUSLocation Configuration Options

The following table lists the configuration options for ReverseGeocodeUSLocation.

Parameter	Description
Option.Dataset	The name of the database that contains the data to use in the search process.
Option.SearchDistance	Specifies the radius, in feet, that ReverseGeocodeUSLocation searches for matches. The allowable range is 0 - 5280 feet. Default = 150 feet.
Option.FindClosestPoint	Enables matching to the nearest point address within the search radius, rather than to the closest feature - such as a street segment or intersection as well as point addresses. **Note:** This option requires that at least one streets data set and one points data set are loaded; otherwise, the match will be made to the closest feature. **Y** Match to the nearest point address within the search radius. **N** Match to the closest point address or feature. Default.
Option.FindNearestAddress	Specifies whether ReverseGeocodeUSLocation should find the nearest interpolated address to the input geocode. **Y** Find the nearest address. Default. **N** Do not find the nearest address. **Note:** You can use this option with the `Option.FindNearestIntersection` option to geocode to both addresses and intersections.
Option.FindNearestUnranged	Specifies whether ReverseGeocodeUSLocation can match to a street segment that does not have a number range. This option is active when `Option.FindNearestAddress=Y`. **Y** Allow ReverGeocodeUSLocation to match to an unranged street segment. Default. **N** Do not allow ReverseGeocodeUSLocation to match to an unranged street segment. **Note:** If you are using the point-level data option, ReverseGeocodeUSLocation ignores the Nearest Unranged option.

Parameter	Description
Option.FindNearestIntersection	Specifies whether ReverseGeocodeUSLocation should find the nearest street intersection to the input geocode.

Y   Find the nearest street intersection. Default.

N   Do not find the nearest street intersection.

> **Note:** You can use this option with the `Option.FindNearestAddress` option to geocode to both addresses and intersections.

Parameter	Description
Option.Offset	Specifies the offset distance from the street segments. The allowable range is 0 - 5280 feet. Default = 50 feet.

The offset distance is used in street-level geocoding to prevent the geocode from being in the middle of a street. It compensates for the fact that street-level geocoding returns a latitude and longitude point in the center of the street where the address is located. Since the building represented by an address is not on the street itself, you do not want the geocode for an address to be a point on the street. Instead, you want the geocode to represent the location of the building which sits next to the street. For example, an offset of 40 feet means that the geocode will represent a point 40 feet back from the center of the street. The distance is calculated perpendicular to the portion of the street segment for the address. Offset is also used to prevent addresses across the street from each other from being given the same point. The diagram below shows an offset point in relation to the original point.

Street coordinates are accurate to 1/10,000th of a degree and interpolated points are accurate to 1/1,000,000th of a degree.

Parameter	Description

Option.Squeeze

Specifies the distance, in feet, to squeeze the street end points in street-level geocoding. The range is 0 -2147483647 feet. Default = 50 feet. The following diagram compares the end points of a street to squeezed end points.

Option.LatLonFormat

Specifies the format to use for returned latitude/longitude.

Decimal	The format is 90.000000-180.000000. Default.
Integer	The format is 90000000-180000000.

Option.InputLatLonFormat

Specifies the format to use for input latitude/longitude.

Decimal	The format is 90.000000-180.000000.
Integer	The format is 90000000-180000000. Default.

Option.RetrieveElevation

Specifies whether ReverseGeocodeUSLocation returns the elevation of the address. Elevation is the distance above or below sea level of a given location. The elevation is returned in the **Elevation** output field, which is part of the Latitude/Longitude output group.

> **Note:** This option requires that you have licensed and installed the Centrus Premium Points database. Elevation data is not available for all addresses. See the coverage map included with the points database.

Y	Return elevation data.
N	Do not return elevation data. Default.

Parameter	Description
Option.RetrieveAPN	Specifies whether ReverseGeocodeUSLocation should determine the address's APN (assessor's parcel number). The APN is an ID number assigned to a property by the local property tax authority. The APN is returned in the **APN** output field, which is part of the Census output group.
	Note: This option requires that you have licensed and installed the Cenrus Enhanced Points or Centrus Premium Points database. APN data is not available for all addresses. See the coverage map included with the points database.
	Y Return APN data.
	N Do not return APN data. Default.
Option.FIND_APPROXIMATE_PBKEY	When Option.FIND_APPROXIMATE_PBKEY is enabled, if an address match is not made to Master Location Data (MLD), but to a different dataset, the pbKey™ unique identifier of the nearest MLD point located within the search distance is returned. To distinguish when a fallback pbKey™ unique identifier is returned, the `PBKey` return value contains a leading character of "X" rather than "P", for example: X00001XSF1IF. Note, all of the other fields returned for the address match, including the geocode and all associated data, reflect the match results for the input address. The fallback pbKey™ unique identifier can then be used for the lookup to the GeoEnrichment dataset(s), and the attribute data for the fallback location is returned for the match.
	Note: This option requires that you have licensed and installed the Master Location Dataset.
	For more information, see **#reversegeocode_options_configurationoptions/conref_PBKeyFallback** on page 116.
	The search distance for the nearest MLD point is configurable using the ReverseGeocodeUSLocation Option.SearchDistance field. The allowable range is 0 - 5280 feet. Default = 150 feet.
	Y When a matched address does not have an associated pbKey™ unique identifier, attempt to return the nearest address record's pbKey™ unique identifier.
	N When a matched address does not have an associated pbKey™ unique identifier, do not return the nearest address record's pbKey™ unique identifier. Default.

A pbKey™ unique identifier is returned when a match is made to the Master Location Dataset (MLD). This field is a persistent identifier for an address. The pbKey™ serves as a lookup key with Pitney Bowes GeoEnrichment datasets to add attribute data for an address. Depending on the GeoEnrichment dataset(s) you install, the attribute data can include property ownership, real estate, census, consumer expenditure, demographic, geographic, fire and flood protection, and/or telecommunication and wireless systems information and more. Some of these datasets return point

location specific data, such as property ownership and real estate, whereas others provide polygonal-based data, for example, fire and flood protection, which can identify flood plains, wildfire or rating territories.

When using PBKey Fallback, if an address match is not made to Master Location Data, but a match is made to a different dataset, the pbKey of the nearest MLD point located within the search distance is returned. To distinguish when a fallback pbKey is returned, the PBKey return value contains a leading character of "X" rather than "P", for example: X00001XSF1IF. Note, all of the other fields returned for the address match, including the geocode and all associated data returns, reflect the match results for the input address. The fallback pbKey can then be used for the lookup to the GeoEnrichment dataset(s), and the attribute data for the fallback location is returned for the match.

The relevance and accuracy of the returned attribute data using a PBKey Fallback location is highly dependent on the type of GeoEnrichment data, as well as the PBKey Fallback search distance. PBKey Fallback is intended for use with GeoEnrichment datasets that have polygonal-based data, rather than point-specific data. For example, the PBKey Fallback option may be suitable for determining the FEMA flood zone for a given location using the Flood Risk Pro GeoEnrichment dataset since it contains data that represents a polygonal region rather than a single coordinate. However, it is important to note that the accuracy of the returned data would very much depend on the size and nature of the individual polygonal features described in the GeoEnrichment data, combined with the search distance used to locate the nearest Master Location Data point. The search distance is configurable with an allowable search radius of 0-5280 feet and a default value of 150 feet.

> **Note:** This option requires that you have licensed and installed the Master Location Dataset.

Output Format

ReverseGeocodeUSLocation Output Format Options

The following table lists the options that control the format of the output.

Parameter	Description
Option.OutputCasing	Specifies the casing of the output data.
	M Returns the output in mixed case. Default. For example:
	123 Main St Mytown FL 12345
	U Returns the output in upper case. For example:
	123 MAIN ST MYTOWN FL 12345

Parameter	Description
Option.OutputVerbose	Specifies whether ReverseGeocodeUSLocation provides an additional description field as output. This field provides the text equivalent to a field represented by a code. For example, `LocationCode` returns a code that indicates the accuracy (quality) of the assigned geocode. `LocationCode.Description` provides the description for the code returned.
	Y Include verbose fields.
	N Do not include verbose fields. Default.

Output Data

ReverseGeocodeUSLocation Output Data Options

The following table lists the options that control which data is returned by ReverseGeocodeUSLocation.

Parameter	Description
Option.OutputRecordType	Specifies the optional data to include in the output. Note that ReverseGeocodeUSLocation always returns the data listed in **Default Output** on page 121. The data you select here is returned with the default output data.
	• **C**—Census • **E**—Parsed Elements • **L**—Latitude/Longitude • **R**—Range • **S**—Segment • **Q**—Qualifiers
	For a list of the fields included in each data type, see **Response** on page 119.
	If you do not want all of the fields in a record type returned, do not use `Option.OutputRecordType`; instead, use `Option.OutputFields` to specify the desired individual output fields.
Option.OutputFields	Specifies the individual output fields you want returned. List fields with a pipe (|) between each field. You can use this field instead of the Output Record Type to limit the output to those fields that are important to your data needs.
	Default list: AddressLine1|LastLine|Longitude|Latitude|MatchCode|LocationCode

Response

ReverseGeocodeUSLocation always returns a default set of output fields that contain the address, geocode, and result indicators. For information about these fields, see **Default Output** on page 121. You can also choose to include optional categories of output data.

Census

Census output fields contain U.S. Census information about the address. To include census data in the output, set `Option.OutputRecordType = C`.

> **Note:** The following Census output fields - `CBSADivisionName, CBSAName, CSAName` and `USCountyName` - are only returned when you set the option `Option.OutputVerbose=Y`.

Response Element	Max. Field Length with null terminator	Description
APN	46	The assessor's parcel number of the property. The assessor's parcel number is an ID number assigned to a property by the local property tax authority.
BlockSuffix	2	Single character block suffix for split Census blocks. Returns A or B. Only available in Centrus Enhanced data.
CBSACode	6	Indicates Core Based Statistical Area (CBSA).
CBSADivisionCode	6	Denotes a subdivision of a CBSA.

Response Element	Max. Field Length with null terminator	Description
CBSADivisionName	128	The name of the Core Based Statistical Area (CBSA) division in which the address is located.
		A CBSA division is a metropolitan statistical area with a population of at least 2.5 million that has been subdivided to form smaller groupings of counties referred to as "metropolitan divisions." For more information, see *Metropolitan and Micropolitan Statistical Areas* section of the U.S. Census Bureau website:
		www.census.gov/population/www/metroareas/metroarea.html
		Note: The CBSA division name is only returned if you set the option `Option.OutputVerbose=Y`.
CBSAMetro	2	Metropolitan Statistical Area. Valid values include:
		Y — Metro statistical area.
		N — Micro statistical area.
		null — Data unavailable.
CBSAName	128	The name of the core based statistical area (CBSA) in which the address is located.
		A CBSA is a collective term that refers to both metropolitan and micropolitan areas. A metropolitan area has a population of more than 50,000, and a micropolitan area has a population between 10,000 and 49,999. For more information, see *Metropolitan and Micropolitan Statistical Areas* section of the U.S. Census Bureau website:
		www.census.gov/population/www/metroareas/metroarea.html
		Note: The CBSA name is only returned if you set the option `Option.OutputVerbose=Y`.
CensusBlockID	16	The ID of the Census Federal Information Processing Standard (FIPS) code.
CensusTract	7	6-digits extracted from the CensusBlockID.
CSACode	4	Denotes the code for a geographic entity that consists of 2 or more adjacent CBSAs with employment interchange measures of at least 15.

Response Element	Max. Field Length with null terminator	Description
CSAName	128	The name of the combined statistical area (CSA) in which the address is located. A CSA is a combination of two or more adjacent Core Based Statistical Areas (CBSAs) with a high employment interchange measure. The employment interchange measure is the sum of the percentage of employed residents of the smaller entity who work in the larger entity and the percentage of the employment in the smaller entity that is accounted for by workers who reside in the larger entity. Pairs of CBSAs with employment interchange measures of at least 25% combine automatically. Pairs of CBSAs with employment interchange measures of at least 15%, but less than 25%, may combine if local opinion in both areas favors combination. **Note:** The CSA name is only returned if you set the option `Option.OutputVerbose=Y.`
USCountyName	128	The name of the county or parish in which the address is located. **Note:** The county/parish name is only returned if you set the option `Option.OutputVerbose=Y.`
USFIPSCountyNumber	4	3-digit FIPS county code extracted from the CensusBlockID.
USFIPSStateCode	3	2-digit FIPS state code extracted from the CensusBlockID.
USFIPSStateCountyCode	6	5-digit FIPS code for state and county extracted from the CensusBlockID.

Default Output

ReverseGeocodeUSAddress always includes the following fields in the output.

Response Element	Max. Field Length with null terminator	Description
AdditionalInputData	61	This field may contain input data that was not used for matching; for example, input address information that appears after a mail stop or attention designator.
AddressLine1	104	First line of the address.
AddressLine2	104	Second line of the address.
City	29	Municipality name.
Confidence	4	Indicates the confidence in the output provided. The range is from 0 (zero) to 100, with 0 being no match and 100 being an exact match.
Country	25	The name of the country. Since ReverseAPNLookup only works for U.S. locations, this field will always contain **United States of America**.
Distance	8	The distance, in feet, from the input geocode to nearest matched address or intersection.
Elevation	11	The location's elevation in feet above or below sea level.
FirmName	41	Name of the company.
LastLine	61	Complete last address line (municipality, state, and postal code).
Latitude	11	7-digit number in degrees and calculated to 4 decimal places (in the format specified).
LocationCode	5	Indicate the accuracy (quality) of the assigned geocode. For more information, see **Address Location Codes** on page 178.

Response Element	Max. Field Length with null terminator	Description
Longitude	12	7-digit number in degrees and calculated to 4 decimal places (in the format specified).
MatchCode	5	Indicates the portions of the address that matched to the directory file. For more information, see **Match Codes** on page 170.
PBKey	13	A unique address identifier that is returned when an address match is made using the Master Location Dataset. The pbKey™ unique identifier is used as a lookup key to a GeoEnrichment dataset, in order to return attribute data for the match.
PercentGeocode	6	The percent along the street segment that matches the geocode. For example, if the returned geocode falls 1/3 along the way of the entire street segment, the percent is 33.000. **Note:** This value is always 0.0 for matches to point-level data and intersections.
PostalCode	10	9-digit ZIP Code with or without a hyphen.
PostalCode.AddOn	5	4-digit ZIP Code extension.
PostalCode.Base	6	5-digit ZIP Code.
ProcessedBy	4	The underlying software that processed the request. KGR for ReverseGeocodeUSLocation.
RRHC	2	Rural Route Highway Contract (RRHC). This field is null if the address not a RRHC.
StateProvince	3	2-character state abbreviation.

Response Element	Max. Field Length with null terminator	Description	
Status	2	Reports the success or failure of the match attempt.	
		null	Success
		F	Failure
Status.Code	23	Reason for failure: • Internal System Error • No Geocode Found • Insufficient Input Data	
Status.Description	32	Description of the problem:	
		Problem + explanation	Returned when Status.Code contains "Internal System Error"
		Geocoding Failed	Returned when Status.Code contains "No Geocode Found"
		No location returned	Returned when Status.Code contains "No Geocode Found"
StreetDataType	20	The data set that ReverseGeocodeUSLocation attempted to match against.	
		USPS	USPS
		TIGER	TIGER
		TOMTOM	TomTom Streets data set
		SANBORN POINT DATA	Sanborn Points data set
		NAVTEQ	NAVTEQ Streets data set
		TOMTOM POINT DATA	TomTom Points data set
		AUXILIARY	Auxiliary file
		CENTRUS POINT DATA	Centrus Points data set
		USER DICTIONARY	User Dictionary
		NAVTEQ POINT DATA	NAVTEQ Points data set
		MASTER LOCATION	Master Location Data

Response Element	Max. Field Length with null terminator	Description
StreetSide	2	Indicates the side of the street the range occupies.
		L The range occupies the left side of the street.
		R The range occupies the right side of the street.
		B The range occupies both sides of the street.
		U Undetermined.
USUrbanName	31	Urbanization name. Puerto Rico addresses only.

Latitude/Longitude

The latitude/longitude output fields contain the geographic coordinates of the location. To include latitude/longitude output fields in the output, set `Option.OutputRecordType = L`.

Response Element	Max. Field Length with null terminator	Description
Elevation	11	The location's elevation in feet above or below sea level.
Latitude	11	7-digit number in degrees and calculated to 4 decimal places (in the format specified).
Longitude	12	7-digit number in degrees and calculated to 4 decimal places (in the format specified).

Parsed Elements

The Parsed Elements output record type contains standard address information as individual units, such as street suffixes (AVE) and leading directionals (N and SE). To include parsed elements in the output, set `Option.OutputRecordType = E`.

Response Element	Max. Field Length with null terminator	Description
ApartmentLabel	5	Unit, such as apartment, suite, or lot.
ApartmentLabel2	5	Unit, such as apartment, suite, or lot.
ApartmentNumber	12	Unit number.
ApartmentNumber2	12	Unit number.
CrossStreetLeadingDirectional	3	Prefix for cross street.
CrossStreetName	41	Name of cross street.
CrossStreetSuffix	5	Cross street suffix.
CrossStreetTrailingDirectional	3	Postfix for cross street.
HouseNumber	12	Building number for the matched location. **Note:** This is an approximate building number based on the latitude and longitude provided. This approximate address may not exist or may not accept mail delivery.

Response Element	Max. Field Length with null terminator	Description
LeadingDirectional	3	Street directional that precedes the street name. For example, the N in 138 N Main Street.
PrivateMailbox	9	Private mailbox. Not output for multiline input.
PrivateMailbox.Designator	5	Private mailbox description. Not output for multiline input.
StreetName	41	Street name.
StreetSuffix	5	The street type of the matched location. For example, AVE for Avenue.
TrailingDirectional	3	Street directional that follows the street name. For example, the N in 456 Washington N.

Qualifiers

The qualifiers output record type contains qualification information about the match, such as the location code and the match code. To include latitude/longitude output fields in the output, set `Option.OutputRecordType = Q`.

Response Element	Max. Field Length with null terminator	Description
CountryLevel	2	The category of postal data available. Always returns **A** in ReverseGeocodeUSLocation—Validates, corrects, and provides missing postal code, city name, state/county name, street address elements, and country name.

Response Element	Max. Field Length with null terminator	Description
DatabaseVersion	15	USPS publish date, in the format Month Year.
EWSMatch	2	Indicates if ReverseGeocodeUSLocation denied a match because of Early Warning System (EWS) data.
		Y EWS denied a match.
		null EWS did not deny a match.
		For more information about EWS, see **Early Warning System (EWS)** on page 16.
ExpirationDate	11	Date the database expires, in the format MM/DD/YYYY.
GeoStanMatchScore	13	Record matching score (for multimatches only).
Intersection	2	Indicates if ReverseGeocodeUSLocation found a cross-street match.
		T True, a cross-street match was found.
		F False, a cross-street match was not found.

Response Element	Max. Field Length with null terminator	Description
IsAlias	4	ReverseGeocodeUSLocation located a matched record by an index alias. Returns 3 characters. The first is an **N** for normal street match or **A** for alias match (including buildings, aliases and firms). The next 2 characters are:
		01 Basic index (normal address match)
		02 USPS street name alias index
		03 USPS building index
		04 USPS firm name index
		05 Statewide intersection alias match (when using the `Usw.gsi` or `Use.gsi` file)
		06 Spatial data street name alias (when using, the `Us_pw.gsi`, `Us_pe.gsi`, `Us_psw.gsi`, or `Us_pse.gsi` file is required)
		07 Alternate index (when using the `Zip9.gsu`, `Zip9e.gsu`, and `Zip9w.gsu` files)
		08 LACS^{Link}
		09 Auxiliary file match
		10 Centrus Alias index (when using the `usca.gsi` file)
LACSAddress	2	Indicates if ReverseGeocodeUSLocation converted an address due to the Locatable Address Conversion System (LACS).
		L Converted
		null Not converted.
		For more information about LACS, see **Locatable Address Conversion System (LACS)** on page 15.
LocationCode.Description	128	LocationCode converted to text. Only returned when you set the configuration options to return additional descriptions (verbose).
MatchCode.Description	128	MatchCode converted to text. Only returned when you set the configuration options to return additional descriptions (verbose).

Response Element	Max. Field Length with null terminator	Description
RecordType	18	Indicates the record type: • GeneralDelivery • HighRise • FirmRecord • Normal • PostOfficeBox • RRHighwayContract
RecordType.Default	2	Indicates type of match that occurred for the record type HighRise or RRHighwayContract: **Y** Default match. **N** Exact match. **U** Not matched.
StreetDataCode	3	Indicates the data used to obtain a match. **0** USPS data in either the Centrus Enhanced, Centrus TomTom, or Centrus NAVTEQ database. **1** TIGER data in the Centrus Enhanced database. **2** TomTom data in the Centrus TomTom database. **6** NAVTEQ data in the Centrus NAVTEQ database. **7** TomTom point-level data in the Centrus TomTom Points database. **8** Point-level data from the Centrus Points database. **9** Auxiliary file data. **10** User Dictionary. **11** Point-level data from NAVTEQ Points database. **12** Master Location Data (MLD). For more information about these databases, see **Enterprise Geocoding Databases** on page 5.
StreetDataType	20	Indicates the data first used to attempt a match.

Range

The range output record type contains information about the street range, such as the high and low unit numbers. To include range data fields in the output, set `Option.OutputRecordType = R.`

Response Element	Max. Field Length with null terminator	Description
Alternate	2	USPS code that specifies whether a record is a base or alternate record.
		B Base record. Base records can represent a range of addresses or an individual address, such as a firm record.
		A Alternate record. Alternate records are individual delivery points.
HouseNumberHigh	12	House number high.
HouseNumberLow	12	House number low.
HouseNumberParity	2	Indicates if the house number range contains even or odd numbers.
		E Even
		O Odd
		B Both
PostalCodeExtensionHigh	5	4-digit ZIP Code extension high.
PostalCodeExtensionLow	5	4-digit Zip Code extension low.
UnitNumberHigh	12	Unit number high.
UnitNumberLow	12	Unit number low.

Response Element	Max. Field Length with null terminator	Description
UnitNumberParity	2	Indicates if the unit number range contains even or odd numbers.
		E Even
		O Odd
		B Both

Segment

The segment output record type contains information about the street segment identified by the data provider. To include segment data fields in the output, set `Option.OutputRecordType = S.`

Response Element	Max. Field Length with null terminator	Description
BlockLeft	16	Provides the Census FIPS Code that indicates the address is on the left side of the street.
BlockRight	16	Provides the Census FIPS Code that indicates the address is on the right side of the street.
BlockSuffixLeft	2	Current left Block suffix for Census 2010 Geography. Returns A or B. Only available in Centrus Enhanced data.
BlockSuffixRight	2	Current right Block suffix for Census 2010 Geography. Returns A or B. Only available in Centrus Enhanced data.

Response Element	Max. Field Length with null terminator	Description	
RoadClass	3	The type of road:	
		1	Major
		2	Minor
SegmentCode	11	Unique 10-digit segment ID assigned by the street network provider.	
SegmentDirection	2	Indicates the order of numbers on a segment.	
		F	Forward
		R	Reversed
		B	Both
		U	Undetermined
SegmentHouseNumberHigh	12	A high range number in the segment.	
SegmentHouseNumberLow	12	A low range number in the segment.	
SegmentLength		The length, in feet, of a block segment. *This field is deprecated.*	
SegmentParity	2	Indicates which side of the street has odd numbers.	
		L	Left side of the street
		R	Right side of the street
		B	Both sides of the street
		U	Undetermined

Response Element	Max. Field Length with null terminator	Description
SegmentPoints	1024	An array of latitude/longitude values that represent the segment coordinates to the matched segment.

The segment points data that is returned for each type of match is as follows:

- Street matches - a list of points
- Point matches - one point is returned that is the same as the matched point since a point match consists of a single segment with a single range.
- Intersection matches - the segment points are returned for the segment listed first in the intersection output. For example, Forest Ave at 6th returns the segment points for Forest Ave.

5 - ReversePBKeyLookup

ReversePBKeyLookup allows you to look up an address using a pbKey^{TM} unique identifier as input and returns all standard returns that are provided as part of address matching.

ReversePBKeyLookup is part of the Enterprise Geocoding Module. For more information about the Enterprise Geocoding Module, see **Enterprise Geocoding Module** on page 4.

In this section

ReversePBKeyLookup

ReversePBKeyLookup allows you to look up an address using a pbKey™ unique identifier as input and returns all standard returns that are provided as part of address matching.

A pbKey™ unique identifier (PBKey) is returned when a match is made to the Master Location Dataset (MLD). It is a 12-character (+1 null) field, which is a persistent identifier for an address. The PBKey serves as a lookup key with Pitney Bowes GeoEnrichment databases to add attribute data for an address. Depending on the GeoEnrichment database, the attribute data can include property ownership, real estate, census, consumer expenditure, demographic, geographic, fire and flood protection, telecommunication and wireless systems information and more.

Use Cases

This section presents two example use cases for ReversePBKeyLookup:

- the first one demonstrates improved efficiency in managing and maintaining consistent and accurate data for customer address lists;
- the second one shows how to use pbKey™ unique identifiers retrieved from GeoEnrichment data products to generate an address list of customers targeted for products and services based on specific attributes associated with their address.

Address Master Data Management	To ensure the latest address information and most accurate locations are being used, businesses may regularly geocode their customer address list. There is a cost in terms of computing power to this intensive process, as well as a small chance of changes to the address match. Some businesses monitor these changes since it's integral to their business. Additionally, many businesses have multiple address databases across different business functions, and have the need for consistent representation of a single address across multiple systems and databases. ReversePBKeyLookup removes the need to re-geocode the address by using the pbKey™ unique identifier rather than the address as input. The address together with latitude/longitude coordinates are returned. The ReversePBKeyLookup process is substantially faster and therefore less costly than using the address to retrieve this information. In addition, since a pbKey™ unique identifier is persistent, there is no chance of matching to a different address.
Identifying Addresses from GeoEnrichment Data	The GeoEnrichment data products are a variety of text-based data files that contain different attributes for each address in the Master Location Dataset. You can use the attributes in one or more of these GeoEnrichment datasets to identify customers for products or services based on those specific attributes. The lookup key for these products is the pbKey™ unique identifier rather than the address. This allows you to easily link customers across multiple datasets if you need to consider attributes included in more than one GeoEnrichment dataset. For example, using Ground View Family Demographics Fabric, in conjunction with Property Attribute Fabric, you would be able to generate a list of pbKey™ unique

identifiers for records that represent young families, with 4 or more persons, in large houses, to target for specific products and services. Once records with the desired attributes have been identified, the pbKey™ unique identifiers from those records can be used to return the address and location information for those customers using ReversePBKeyLookup.

Licensing

ReversePBKeyLookup requires a special license. There are two levels of licensing for ReversePBKeyLookup:

- Standard - This license allows ReversePBKeyLookup of all of the standard MLD addresses.
- Enhanced - This license allows ReversePBKeyLookup of a portion of MLD addresses that require an additional royalty due to address sourcing constraints.

Requirements

ReversePBKeyLookup includes these requirements:

- You have licensed and installed the Master Location Dataset (MLD).
- You have licensed and installed the DVDMLDR dataset.
- The MLD and DVDMLDR datasets must be the same vintage.

ReversePBKeyLookup Search Results

When using ReversePBKeyLookup, the search results can return zero to many MLD point address variations that match the input PBKey. There will be no matches returned if the given PBKey is not found. While many PBKeys map to a single point-level address, some PBKeys map to multiple point address variations. Getting multiple point address variations from one PBKey can occur in two circumstances:

1. **Alias matches.** Some streets are known by their common name and one to many aliases. In this case, MLD may contain all variations of street names. An example of multiple alias match returns for an input PBKey (P00008BCG8WM) is shown below:

   - `AP02. Normal match (non-alias). 1206 W 600 S, FOUNTAINTOWN, IN 46130-9409`
   - `AP02. Alias match. 1206 W 1200 N, FOUNTAINTOWN, IN 46130-9409`
   - `AP02. Alias match. 1206 W COUNTY ROAD 1200 N, FOUNTAINTOWN, IN 46130-9409`
   - `AP02. Alias match. 1206 W COUNTY ROAD 600 S, FOUNTAINTOWN, IN 46130-9409`

2. **Multi-unit buildings with/without units.** In some cases, there are multi-unit addresses without individual unit address records. In this case, you may see multiple address records returned for the same input PBKey, some without unit designations and others with ranged unit designations. In the case of multi-unit addresses that have individual suite/unit number address designations, each will have their own distinct PBKey. The following example shows address results for a

PBKey that maps to a building with and without units, which share the same PBKey/location (P00003PZZOIE):

- `AP02. Normal match (non-alias). 4750 WALNUT ST, BOULDER, CO 80301-2532`
- `AP02. Normal match (non-alias). 4750 WALNUT ST STE 100-103, BOULDER, CO 80301-2532`
- `AP02. Normal match (non-alias). 4750 WALNUT ST STE 205-205, BOULDER, CO 80301-2532`
- `AP02. Normal match (non-alias). 4750 WALNUT ST, BOULDER, CO 80301-2538`

ReversePBKeyLookup Return Codes and Match Codes

The table below lists the Return Codes and Match Codes returned when using ReversePBKeyLookup.

License	Input PBKey	Point Results	Match Code
Enhanced	Found	One Enhanced	V000
Enhanced	Found	Multiple Standard and/or Enhanced	V001
Enhanced	Not Found	None	E040
Standard	Found	One Standard	V000
Standard	Found	Multiple Standard	V001
Standard	Found	One Standard, some Enhanced	V002
Standard	Found	Multiple Standard, some Enhanced	V003
Standard	Found	All Enhanced	E041
Standard	Not Found	None	E040
No license	Not applicable	Not applicable	E000

ReversePBKeyLookup Resource

JSON endpoint:

```
http://server:port/rest/ReversePBKeyLookup/results.json
```

XML endpoint:

```
http://server:port/rest/ReversePBKeyLookup/results.xml
```

Example with JSON Response

The following example requests a JSON response:

```
http://myserver:8080/rest/ReversePBKeyLookup/results.json?
Data.PBKey=P00003PZ8QKG&Option.Dataset=EGM_1
```

The JSON returned by this request would be:

```
{"output_port": [{
  Latitude=40.051418;
  Longitude=-105.265424;
  StreetSide=L;
  Distance=;
  PercentGeocode=0.0;
  FirmName=;
  AddressLine1=2440 Tamarack Ave;
  AddressLine2=;
  LastLine=Boulder, CO  80304-0987;
  AdditionalInputData=;
  City=Boulder;
  StateProvince=CO;
  PostalCode.Base=80304;
  PostalCode.AddOn=0987;
  PostalCode=80304-0987;
  USUrbanName=;
  Country=United States of America;
  LocationCode=AP02;
  MatchCode=V000;
  StreetDataType=MASTER LOCATION;
  PBKey=P00003PZ8QKG;
  Confidence=96;
  ProcessedBy=KGL;
  Status=;
  Status.Code=;
```

```
    Status.Description=;
    user_fields=System.Object[]},
]}
```

Example with XML Response

The following example requests an XML response:

```
http://myserver:8080/rest/ReversePBKeyLookup/results.xml?
Data.PBKey=P00003PZ8QKG&Option.Dataset=EGM_1
```

The XML returned by this request would be:

```
<xml.ReversePBKeyLookupResponse
xmlns="http://www.pb.com/spectrum/services/ReversePBKeyLookup">
  <output_port>
    <Address>
      <Latitude>40.051418</Latitude>
      <Longitude>-105.265424</Longitude>
      <StreetSide>L</StreetSide>
      <Distance></Distance>
      <PercentGeocode>0.0</PercentGeocode>
      <FirmName></FirmName>
      <AddressLine1>2440 Tamarack Ave</AddressLine1>
      <AddressLine2></AddressLine2>
      <LastLine>Boulder, CO  80304-0987</LastLine>
      <AdditionalInputData></AdditionalInputData>
      <City>Boulder</City>
      <StateProvince>CO</StateProvince>
      <PostalCode.Base>80304</PostalCode.Base>
      <PostalCode.AddOn>0987</PostalCode.AddOn>
      <PostalCode>80304-0987</PostalCode>
      <USUrbanName></USUrbanName>
      <Country>United States of America</Country>
      <LocationCode>AP02</LocationCode>
      <MatchCode>V000</MatchCode>
      <StreetDataType>MASTER LOCATION</StreetDataType>
      <PBKey>P00003PZ8QKG</PBKey>
      <Confidence>96</Confidence>
      <ProcessedBy>KGL</ProcessedBy>
      <Status></Status>
      <Status.Code></Status.Code>
      <Status.Description></Status.Description>
      <user_fields />
    </Address>
  </output_port>
</xml.ReversePBKeyLookupResponse>
```

Request

Input

Reverse PBKey Lookup Input Data

Reverse PBKey Lookup takes a pbKey^{TM} unique identifier as input. The table below provides information about the format and layout of the input.

Parameter	Format	Description
Data.PBKey	String [13]	12-character plus 1 null character alphanumeric pbKey^{TM} unique identifier.

Input Parameters

The following table lists the options that control ReversePBKeyLookup processing.

Note: As the Enterprise Geocoding Module transitions its administrative tasks to a web-based Management Console, labels for the options may use different wording than what you see in Enterprise Designer. There is no difference in behavior.

Parameter	Description
Option.Dataset	Specifies the database to use to look up the parcel. Use the database name specified in Management Console's Spectrum Databases page.
Option.LatLongFormat	Specifies the format for returned latitude/longitude.
	Decimal (90.000000-180.000000) Default.
	Integer (90000000-180000000)

Parameter	Description
Option.OutputCasing	Specifies the casing of the output data.

M   Returns the output in mixed case. Default. For example:

123 Main St
Mytown FL 12345

U   Returns the output in upper case. For example:

123 MAIN ST
MYTOWN FL 12345

Option.OutputVerbose	Specifies whether Reverse PBKey Lookup provides an additional description field as output. This field provides the text equivalent to a field represented by a code. For example, `LocationCode` returns a code that indicates the accuracy (quality) of the assigned geocode. `LocationCode.Description` provides the description for the code returned.

Y   Include verbose fields.

N   Do not include verbose fields. Default.

Option.OutputRecordType	Specifies optional data to include in the output. Note that Reverse PBKey Lookup always returns the default data listed in **Default Output** on page 143. The data you select here is returned with the default output data.

- `C`—Census
- `E`—Parsed Elements
- `L`—Latitude/Longitude
- `R`—Range
- `S`—Segment
- `Q`—Qualifiers

For a list of fields included in each record type, see **Response** on page 143.

If you do not want all of the fields in a group returned, do not select the group, and instead list only those fields you want returned in `Option.OutputFields`.

Option.OutputFields	Indicates the individual output fields you want returned. List fields with a pipe (`|`) between each field. You can use this field instead of the `Option.OutputRecordType` field to limit the output to the specific fields you want.

Default list: AddressLine1|LastLine|Longitude|Latitude|MatchCode|LocationCode

Response

Reverse PBKey Lookup always returns a default set of output fields that contain the address, geocode, and result indicators. For information about these fields, see **Default Output** on page 143. You can also choose to include optional categories of output data.

Default Output

Default Output Fields

Reverse PBKey Lookup always returns these fields in the response; however, the fields may not always be populated.

Response Element	Max. Field Length with null terminator	Description
AdditionalInputData	61	This field may contain input data that was not used for matching; for example, input address information that appears after a mail stop or attention designator.
AddressLine1	104	First line of the address.
AddressLine2	104	Second line of the address.
City	29	Municipality name.
Confidence	4	Indicates the confidence in the output provided. The range is from 0 (zero) to 100, with 0 being no match and 100 being an exact match.
Country	25	The name of the country. Since Reverse PBKey Lookup only works for U.S. locations, this field will always contain **United States of America**.

Response Element	Max. Field Length with null terminator	Description
Distance	8	The distance, in feet, from the input geocode to nearest matched address or intersection.
FirmName	41	Name of the company.
LastLine	61	Complete last address line (municipality, state, and postal code).
Latitude	11	Seven-digit number in degrees and calculated to 4 decimal places (in the format specified).
LocationCode	5	Indicates the accuracy (quality) of the assigned geocode. For more information, see **Address Location Codes** on page 178.
Longitude	12	Seven-digit number in degrees and calculated to 4 decimal places (in the format specified).
MatchCode	5	Indicates the portions of the address that matched to the directory file. For more information, see **Match Codes** on page 170.
PBKey	13	The PBKey that was specified in the input.
PercentGeocode	6	The percent along the street segment that matches the geocode. For example, if the returned geocode falls 1/3 along the way of the entire street segment, the percent is 33.000. **Note:** This value is always 0.0 for matches to point-level data and intersections.
PostalCode	10	Nine-digit ZIP Code with or without a hyphen.
PostalCode.AddOn	5	Four-digit ZIP Code extension.

Response Element	Max. Field Length with null terminator	Description
PostalCode.Base	6	Five-digit ZIP Code.
ProcessedBy	4	The feature code for the stage that processed the request. For Reverse PBKey Lookup, the value is `KGL`.
StateProvince	3	Two-character state abbreviation.
Status	2	Reports the success or failure of the match attempt

Status:

null	Success
F	Failure

Response Element	Max. Field Length with null terminator	Description
Status.Code	23	Reason for failure: • Internal System Error • No Address Found • Insufficient Input Data

Status.Description — 128 — Description of the problem:

Problem + explanation	Returned when Status.Code = Internal System Error.
Geocoding Failed	Returned when Status.code = No Address Found.
No location returned	Returned when Status.code = No Address Found.

Response Element	Max. Field Length with null terminator	Description	
StreetDataType	20	The database that Reverse PBKey Lookup attempted to match against.	
		USPS	USPS
		TIGER	TIGER
		TOMTOM	TomTom Streets data set
		SANBORN POINT DATA	Sanborn Points data set
		NAVTEQ	NAVTEQ Streets data set
		TOMTOM POINT DATA	TomTom Points data set
		AUXILIARY	Auxiliary file
		CENTRUS POINT DATA	Centrus Points data set
		USER DICTIONARY	User Dictionary
		NAVTEQ POINT DATA	NAVTEQ Points data set
		MASTER LOCATION	Master Location Data
StreetSide	2	Indicates the side of the street the range occupies. One of the following:	
		L	The range occupies the left side of the street.
		R	The range occupies the right side of the street.
		B	The range occupies both sides of the street.
		U	Undetermined.
USUrbanName	6	USPS[®] urbanization name. Puerto Rican addresses only.	

Census

Census output fields contain U.S. Census information about the address. To include census data in the output, set `Option.OutputRecordType` = `C`.

> **Note:** These Census output fields - `CBSADivisionName, CBSAName, CSAName` and `USCountyName` - are only returned when you set `Option.OutputVerbose=Y`.

Response Element	Max. Field Length with null terminator	Description
APN	46	The assessor's parcel number of the property. The assessor's parcel number is an ID number assigned to a property by the local property tax authority.
BlockSuffix	2	Single character block suffix for split Census blocks. Returns A or B. Only available in Centrus Enhanced data.
CBSACode	6	Indicates Core Based Statistical Area (CBSA).
CBSADivisionCode	6	Denotes a subdivision of a CBSA.
CBSADivisionName	128	The name of the Core Based Statistical Area (CBSA) division in which the address is located. A CBSA division is a metropolitan statistical area with a population of at least 2.5 million that has been subdivided to form smaller groupings of counties referred to as "metropolitan divisions." For more information, see *Metropolitan and Micropolitan Statistical Areas* section of the U.S. Census Bureau website: **www.census.gov/population/www/metroareas/metroarea.html** **Note:** The CBSA division name is only returned if you set the option `Option.OutputVerbose=Y`.
CBSAMetro	2	Metropolitan Statistical Area. Valid values include: **Y** — Metro statistical area **N** — Micro statistical area **null** — Data unavailable

Response Element	Max. Field Length with null terminator	Description
CBSAName	128	The name of the core based statistical area (CBSA) in which the address is located. A CBSA is a collective term that refers to both metropolitan and micropolitan areas. A metropolitan area has a population of more than 50,000, and a micropolitan area has a population between 10,000 and 49,999. For more information, see *Metropolitan and Micropolitan Statistical Areas* section of the U.S. Census Bureau website: **www.census.gov/population/www/metroareas/metroarea.html** **Note:** The CBSA name is only returned if you set the option `Option.OutputVerbose=Y.`
CensusBlockID	16	The ID of the Census Federal Information Processing Standard (FIPS) code.
CensusTract	7	Six digits extracted from the CensusBlockID.
CSACode	4	Denotes the code for a geographic entity that consists of 2 or more adjacent CBSAs with employment interchange measures of at least 15.
CSAName	128	The name of the combined statistical area (CSA) in which the address is located. A CSA is a combination of two or more adjacent Core Based Statistical Areas (CBSAs) with a high employment interchange measure. The employment interchange measure is the sum of the percentage of employed residents of the smaller entity who work in the larger entity and the percentage of the employment in the smaller entity that is accounted for by workers who reside in the larger entity. Pairs of CBSAs with employment interchange measures of at least 25% combine automatically. Pairs of CBSAs with employment interchange measures of at least 15%, but less than 25%, may combine if local opinion in both areas favors combination. **Note:** The CSA name is only returned if you set the option `Option.OutputVerbose=Y.`

Response Element	Max. Field Length with null terminator	Description
USCountyName	128	The name of the county or parish in which the address is located. **Note:** The county/parish name is only returned if you set the option `Option.OutputVerbose=Y`.
USFIPSCountyNumber	4	The three-digit FIPS county code of the county in which the address is located.
USFIPSStateCode	3	The two-digit FIPS state code of the state in which the address is located.
USFIPSStateCountyCode	6	Five-digit FIPS code for state and county extracted from the CensusBlockID.

Latitude/Longitude

The latitude/longitude output fields contain the geographic coordinates of the location. To include latitude/longitude output fields in the output, set `Option.OutputRecordType` = L.

Response Element	Max. Field Length with null terminator	Description
Latitude	11	7-digit number in degrees and calculated to 4 decimal places (in the format specified).
Longitude	12	7-digit number in degrees and calculated to 4 decimal places (in the format specified).

Parsed Elements

The Parsed Elements output fields contain standard address information as individual units, such as street suffixes (AVE) and leading directionals (N and SE). To include parsed elements in the output, set `Option.OutputRecordType` = `E`.

Response Element	Max. Field Length with null terminator	Description
AdditionalInputData	61	This field may contain input data that was not used for matching; for example, input address information that appears after a mail stop or attention designator.
AddressLine1	104	First line of the address.
AddressLine2	104	Second line of the address.
ApartmentLabel	5	Apartment designator (such as STE or APT), for example: 123 E Main St. APT 3
ApartmentLabel2	5	Secondary apartment designator, for example: 123 E Main St. APT 3, 4th Floor
ApartmentNumber	12	Apartment number, for example: 123 E Main St. APT 3
ApartmentNumber2	12	Secondary apartment number, for example: 123 E Main St. APT 3, 4th Floor
City	29	Municipality name.
Country	25	The name of the country. Since Reverse PBKey Lookup only works for U.S. locations, this field will always contain **United States of America**.

Response Element	Max. Field Length with null terminator	Description
CrossStreetLeadingDirectional	3	Leading directional, for example: 123 E Main St. Apt 3
CrossStreetName	41	Cross street name, for example: 123 E Main St. Apt 3 **Note:** Reverse PBKey Lookup only returns Cross street outputs if you entered an intersection as an address. For example, entering Pearl and 28th, Boulder, CO returns cross street information. Entering 2800 Pearl, Boulder, CO does NOT return cross street information.
CrossStreetSuffix	5	Cross street suffix, for example: 123 E Main St. Apt 3
CrossStreetTrailingDirectional	3	Cross street trailing directional, for example: 123 Pennsylvania Ave NW
FirmName	41	Name of the company.
HouseNumber	12	Building number, for example: 123 E Main St. Apt 3 **Note:** This is an approximate building number based on the APN, FIPS county code, and FIPS state code provided. This approximate address may not exist or may not accept mail delivery.
LastLine	61	Complete last address line (municipality, state, and postal code).
LeadingDirectional	3	Leading directional, for example: 123 E Main St. Apt 3
PostalCode	10	Nine-digit ZIP Code with or without a hyphen.
PostalCode.Base	6	Five-digit ZIP Code.
PostalCode.AddOn	5	Four-digit ZIP Code extension.

Response Element	Max. Field Length with null terminator	Description
PrivateMailbox	9	Private mailbox indicator. Not output for multiline input.
PrivateMailbox.Designator	5	The type of private mailbox. Possible values include: • Standard • Non-Standard
RRHC	2	Rural Route/Highway Contract indicator.
StateProvince	3	Two-character state abbreviation.
StreetName	41	Street name, for example: 123 E Main St. Apt 3
StreetSuffix	5	Street suffix, for example: 123 E Main St. Apt 3
TrailingDirectional	3	Trailing directional, for example: 123 Pennsylvania Ave NW
USUrbanName	6	USPS® urbanization name. Puerto Rican addresses only.

Qualifiers

The qualifiers output fields contain qualification information about the match, such as the location code and the match code. To include qualifier output fields in the output, set `Option.OutputRecordType` = Q.

Response Element	Max. Field Length with null terminator	Description
CountryLevel	2	The category of postal data available. Always returns A in Reverse PBKey Lookup—Validates, corrects, and provides missing postal code, city name, state/county name, street address elements, and country name.
DatabaseVersion	15	USPS publish date, in the format Month Year.
EWSMatch	2	Indicates if Reverse PBKey Lookup denied a match because of Early Warning System (EWS) data.
		Y EWS denied a match.
		null EWS did not deny a match.
		For more information about EWS, see **Early Warning System (EWS)** on page 16.
ExpirationDate	11	Date the database expires, in the format MM/DD/YYYY.
GeoStanMatchScore	13	Record matching score (for multimatches only).
Intersection	2	Indicates if ReverseAPNLookup found a cross-street match.
		T True, a cross-street match was found.
		F False, a cross-street match was not found.

Response Element	Max. Field Length with null terminator	Description
IsAlias	4	Reverse PBKey Lookup located a matched record by an index alias. Returns 3 characters. The first is an **N** for normal street match or **A** for alias match (including buildings, aliases, firms, etc.). The next 2 characters are:
		01 Basic index (normal address match)
		02 USPS street name alias index
		03 USPS building index
		04 USPS firm name index
		05 Statewide intersection alias match (when using the Usw.gsi or Use.gsi file)
		06 Spatial data street name alias (when using the Us_pw.gsi, Us_pe.gsi, Us_psw.gsi, or Us_pse.gsi file)
		07 Alternate index (when using Zip9.gsu, Zip9e.gsu, and Zip9w.gsu)
		08 LACS^{Link}
		09 Auxiliary file match
		10 Centrus Alias index (when using usca.gsi)
LACSAddress	2	Indicates if Reverse PBKey Lookup converted an address due to the Locatable Address Conversion System (LACS).
		L Converted
		null Not converted
		For more information about LACS, see **Locatable Address Conversion System (LACS)** on page 15.
LocationCode.Description	128	LocationCode converted to text. Only returned when you set the configuration options to return additional descriptions (verbose).
MatchCode.Description	128	MatchCode converted to text. Only returned when you set the configuration options to return additional descriptions (verbose).

Response Element	Max. Field Length with null terminator	Description
RecordType	18	Indicates the record type: • GeneralDelivery • HighRise • FirmRecord • Normal • PostOfficeBox • RRHighwayContract
RecordType.Default	2	Indicates type of match that occurred for the record type HighRise or RRHighwayContract: **Y** — Default match **N** — Exact match **U** — Not matched
StreetDataCode	3	Indicates the data used to obtain a match. **0** — USPS data in either the Centrus Enhanced, Centrus TomTom, or Centrus NAVTEQ database. **1** — TIGER data in the Centrus Enhanced database. **2** — TomTom data in the Cenrus TomTom database. **6** — NAVTEQ data in the Centrus NAVTEQ database. **7** — TomTom point-level data in the Centrus TomTom Points database. **8** — Point-level data from the Centrus Points database. **9** — Auxiliary file data **10** — User Dictionary. **11** — Point-level data from NAVTEQ Points database. **12** — Master Location Data (MLD). For more information about these databases, see **Enterprise Geocoding Databases** on page 5.

Response Element	Max. Field Length with null terminator	Description
StreetDataType	20	The database that Reverse PBKey Lookup attempted to match against.

USPS	USPS
TIGER	TIGER
TOMTOM	TomTom Streets data set
SANBORN POINT DATA	Sanborn Points data set
NAVTEQ	NAVTEQ Streets data set
TOMTOM POINT DATA	TomTom Points data set
AUXILIARY	Auxiliary file
CENTRUS POINT DATA	Centrus Points data set
USER DICTIONARY	User Dictionary
NAVTEQ POINT DATA	NAVTEQ Points data set
MASTER LOCATION	Master Location Data

Range

The range output fields contain information about the street range, such as the high and low unit numbers. To include range data fields in the output, set `Option.OutputRecordType` = R.

Response Element	Max. Field Length with null terminator	Description
Alternate	2	USPS code that specifies whether a record is a base or alternate record.
		B Base record. Base records can represent a range of addresses or an individual address, such as a firm record.
		A Alternate record. Alternate records are individual delivery points.
HouseNumberHigh	12	House number high.

Response Element	Max. Field Length with null terminator	Description
HouseNumberLow	12	House number low.
HouseNumberParity	2	Indicates if the house number range contains even or odd numbers.
		E Even
		O Odd
		B Both
PostalCodeExtensionHigh	5	4-digit ZIP Code extension high.
PostalCodeExtensionLow	5	4-digit Zip Code extension low.
UnitNumberHigh	12	Unit number high.
UnitNumberLow	12	Unit number low.
UnitNumberParity	2	Indicates if the unit number range contains even or odd numbers.
		E Even
		O Odd
		B Both

Segment

The segment output fields contain information about the street segment identified by the data provider. To include segment data fields in the output, set `Option.OutputRecordType` = S.

Response Element	Max. Field Length with null terminator	Description
BlockLeft	16	Provides the Census FIPS Code that indicates the address is on the left side of the street.
BlockRight	16	Provides the Census FIPS Code that indicates the address is on the right side of the street.
BlockSuffixLeft	2	Current left Block suffix for Census 2010 Geography. Returns A or B. Only available in Centrus Enhanced data.
BlockSuffixRight	2	Current right Block suffix for Census 2010 Geography. Returns A or B. Only available in Centrus Enhanced data.
RoadClass	3	The type of road: **1** The road is a major road. **2** The road is a minor road.
SegmentCode	11	Unique 10-digit segment ID assigned by the street network provider.
SegmentDirection	2	Indicates the order of numbers on a segment. **F** Forward **R** Reversed **B** Both **U** Undetermined
SegmentHouseNumberHigh	12	A high range number in the segment.
SegmentHouseNumberLow	12	A low range number in the segment.

Response Element	Max. Field Length with null terminator	Description
SegmentLength		The length, in feet, of a block segment. *This field is deprecated.*
SegmentParity	2	Indicates which side of the street has odd numbers.
		L Left side of the street
		R Right side of the street
		B Both sides of the street
		U Undetermined

6 - Geocode US Address Auxiliary Files

In this section

Auxiliary File Overview

Use auxiliary files to match against special data that is not included in the GeocodeUSAddress database.

The GeocodeUSAddress database is updated regularly to incorporate changes made by the USPS and third-party data vendors. You may have newer information that has not yet been incorporated. Auxiliary files provide a way for you to process your input records against a file that includes these changes.

> **Note:** ReverseGeocodeUSAddress does not support auxiliary files.

There are two types of auxiliary file records:

• **Street Records**—Contains a range of one or more addresses on a street. For required fields, see **Auxiliary File Layout** on page 164. A street record must not have secondary address information mailstops, Private mail boxes (PMBs), and PO Boxes.
• **Landmark Records**—Represents a single site. For required fields, see **Auxiliary File Layout** on page 164. A landmark record must not have street type abbreviations, predirectional and postdirectional abbreviations, or low and high house numbers.

> **Note:** You cannot update the auxiliary file while GeocodeUSAddress is running. If you want to update the auxiliary file, stop GeocodeUSAddress before attempting to replace or edit the file.

Matching to Auxiliary Files

GeocodeUSAddress matches an input address to an auxiliary file as follows:

1. GeocodeUSAddress determines if there is an auxiliary file present.

   If you have an auxiliary file in the dataset directory, GeocodeUSAddress automatically loads and attempts to match to the auxiliary file. You can verify that GeocodeUSAddress found an auxiliary file by looking at the version information page in the Management Console. One of the following statuses display:

   • Loaded—An auxiliary file is loaded
   • None—An auxiliary file has not been found or loaded
   • Invalid—An auxiliary file was found, but failed to successfully load

GeocodeUSAddress only accepts one auxiliary file. If more than one auxiliary files is present, GeocodeUSAddress attempts to match against the first file. GeocodeUSAddress ignores any additional auxiliary files for matching, regardless if GeocodeUSAddress found a match to the first auxiliary file.

If a record in the auxiliary files is invalid, GeocodeUSAddress returns a invalid record message. GeocodeUSAddress continues to match input addresses with the auxiliary file, but will not match to the invalid auxiliary file record.

2. If an auxiliary file is present, GeocodeUSAddress attempts to match to the auxiliary file.

   GeocodeUSAddress assumes that the auxiliary file is the most accurate data set and attempts to find a match to the input address in the auxiliary file. If GeocodeUSAddress cannot find a match in the auxiliary file, it matches the input address with the other Enterprise Geocoding Module databases.

   > **Note:** GeocodeUSAddress only matches input address lists to auxiliary files if there is an exact match. Your input address list should be free of misspellings and incomplete addresses.

3. If GeocodeUSAddress finds an exact record match to the auxiliary file, it standardizes the match to USPS regulations and returns the output of the auxiliary file match.

   GeocodeUSAddress uses the following defaults if you do not include the values in the auxiliary file:

   • House number parity = B (both odds and evens)
   • Segment direction = A (ascending)
   • Side of street = U (unknown)

Record Type Matching Rules

When matching against an auxiliary file, GeocodeUSAddress uses the following rules:

Street record match

• The input house number must fall within or be equal to the low and high house number values of the auxiliary record.
• The input house number must agree with the parity of the auxiliary record.
• The input ZIP Code must exactly match the ZIP Code of the auxiliary record.

Landmark record match

• The input data must contain a ZIP Code and address line, and the values must exactly match the values on the auxiliary record.
• The input address cannot have any other data, such as a house number, unit number, or Private Mail Box (PMB).

Note: GeocodeUSAddress only matches the ZIP Code against the auxiliary file. GeocodeUSAddress does not verify that the ZIP Code of the input address record is correct for the city and state. Validate this information in your input lists before processing against the auxiliary file.

Unavailable Features and Functions

The following features and functions do not apply when GeocodeUSAddress makes an auxiliary file match.

• GeocodeUSAddress does not match to
• two-line addresses
• multi-line addresses
• intersection addresses
• dual addresses
• GeocodeUSAddress does not perform EWS, ZIPMove, LACSLink, or DPV processing on auxiliary matches
• You can only access the auxiliary file with processing through the Find function. You cannot access the auxiliary file through the Find First/Next or MBR functions
• You can only access the auxiliary file logic using the address code option of the Find function, not the geocode option.

Auxiliary Match Output

GeocodeUSAddress provides special data type, match codes, and location code values for auxiliary matches. When GeocodeUSAddress finds a match to an auxiliary file, the default output follows these conventions:

• GeocodeUSAddress formats the auxiliary file match as a street-style address for output. This excludes PO Boxes, Rural Routes, General Delivery, etc.
• GeocodeUSAddress follows the case setting you indicate (by default, upper case) by the casing function. GeocodeUSAddress does not maintain the casing in the auxiliary file for mixed casing values. For example, GeocodeUSAddress returns O'Donnell as ODONNELL or Odonnell depending on the setting of the casing function.

Note: GeocodeUSAddress does not change the casing for the User Data field.

• GeocodeUSAddress removes spaces at the beginning and ending of fields in the auxiliary file.

Note: GeocodeUSAddress does not remove spaces for the User Data field.

Auxiliary File Layout

You must comply with the following organizational rules when you create auxiliary file:

- Files are fixed-width text files with a `.gax` extension
- Files can contain up to 500,000 records.
- Use semicolons in the first column to indicate a row is a comment, not a data record; GeocodeUSAddress ignores rows that begin with a semicolon.
- For optimal performance, order the records within the file by descending ZIP Code, and then descending street name.
- Records must represent only one side of a street. To represent both sides of a street, create a record for each side of the street.
- Records must represent segments that are straight lines.
- House numbers must follow USPS rules documented in Publication 28.
- Numeric fields, such as ZIP Codes, must contain only numbers.
- If house numbers are present in the record, the house number range must be valid according to USPS rules documented in Publication 28, Appendix E.
- Latitude and Longitude values must be in millionths of decimal degrees.
- Records cannot contain PO Box addresses.

The following table shows auxiliary file layout.

Field	Description	Required	Required for Street Segment Match	Required for Landmark Match	Exact Match Required if Present	Length	Position
ZIP Code	5-digit ZIP Code.	X	X	X	X	5	1-5
Street name	Name of the street or landmark.	X	X	X	X	30	6-35

Field	Description	Required	Required for Street Segment Match	Required for Landmark Match	Exact Match Required if Present	Length	Position
Street type abbreviation	Street type. Also called street suffix. See the USPS Publication 28, Appendix C for a complete list of supported street types.				X	4	36-39
Predirectional	USPS street name predirectional abbreviation. Supported values are N, E, S, W, NE, NW, SE, and SW.				X	2	40-41
Postdirectional	USPS street name postdirectional abbreviations. Supported values are N, E, S, W, NE, NW, SE, and SW.				X	2	42-43
RESERVED	RESERVED					4	44-47
Low house number	Low house number of the address range.	X	X			11	48-58
High house number	High house number of the address range.	X	X			11	59-69
House number parity	Indicates the parity of the house number in the range. E - Even O - Odd B - Both					1	70

Field	Description	Required	Required for Street Segment Match	Required for Landmark Match	Exact Match Required if Present	Length	Position
Segment direction	Direction the house numbers progress along the segment: F - Forward (default) R - Reverse					1	71
RESERVED	RESERVED					1	72
FIPS state	US government FIPS state code.					2	73-74
FIPS county	US government FIPS county code.					3	75-77
Census tract	US Census tract number.					6	78-83
Census block group	US Census block group number.					1	84
Census block ID	US Census block ID number.					3	85-87
RESERVED	RESERVED					5	88-92
State abbreviation	USPS state abbreviation.					2	93-94
County name	Name of the county.					25	95-119
MCD code	Minor Civil Division code.					5	120-124

Field	Description	Required	Required for Street Segment Match	Required for Landmark Match	Exact Match Required if Present	Length	Position
MCD name	Minor Civil Division name.					40	125-164
CBSA code	Core Based Statistical Area code.					5	165-169
CBSA name	Core Based Statistical Area name.					49	170-218
RESERVED	RESERVED					5	219-223
City Name	City name. Overrides the city/state preferred city name upon a return.					40	224-263
RESERVED	RESERVED					237	264-500
User-defined data	User-defined data.					300	501-800
Record ID Number	User-defined unique record identifier.					10	801-810
Side of street	Side of the street for the address: L - Left side R - Right side B - Both sides U - Unknown side (default) This is relative to the segment endpoints and the segment direction.					1	811

Field	Description	Required	Required for Street Segment Match	Required for Landmark Match	Exact Match Required if Present	Length	Position
Beginning longitude	Beginning longitude of the street segment in millionths of degrees.	X	X	X		11	812-822
Beginning latitude	Beginning latitude of the street segment in millionths of degrees.	X	X	X		10	823-832
Ending longitude	Ending longitude of the street segment in millionths of degrees.					11	833-843
Ending latitude	Ending latitude of the street segment in millionths of degrees.					10	844-853

7 - Match and Location Codes for U.S. Geocoding

In this section

Match Codes

The geocoder returns match codes indicating the address portions that matched or did not match to the database.

If the geocoder cannot make a match, the match code begins with "E" and the remaining digits indicate why the address did not match. For the descriptions of the "E" match codes, see **Match Codes for No Match - Definitions for "Ennn" return codes** on page 176. The digits do not specifically refer to which address elements did not match, but rather why the address did not match.

Match Codes

This section covers:

- **Match Code Definitions** on page 170
- **Definitions for 1st-3rd hex digit match code values** on page 172
- **Definitions for Extended Match Code (3rd hex digit values)** on page 173
- **Definitions for the Reverse PBKey Lookup "Vhhh" return codes** on page 175
- **Match Codes for No Match - Definitions for "Ennn" return codes** on page 176

Match Code Definitions

The table below contains the match code values.

For descriptions of the hex digits for the match codes, see **Definitions for 1st-3rd hex digit match code values** on page 172.

Code	Description
Ahhh	Same as `Shhh`, but indicates match to an alias name record or an alternate record.
Chh	The street address did not match, but the geocoder located a street segment based on the input ZIP Code or city .
D00	Matched to a small town with P.O. Box or General Delivery only.

Code	Description
Ghhh	Matched to an auxiliary file.
Hhhh	The house number was changed.
Jhhh	Matched to a user-defined dictionary.
Nxx	Matched to the nearest address. Used with reverse geocoding. The following are the only values for N:

NS0	Nearest street center match (nearest street segment interpolated)
NS1	Nearest unranged street segment
NP0	Nearest point address
NX0	Nearest intersection

Code	Description
P	Successful reverse APN lookup.
Qhhh	Matched to USPS range records with unique ZIP Codes. CASS rules prohibit altering an input ZIP if it matches a unique ZIP Code value.
Rhhh	Matched to a ranged address.
Shhh	Matched to USPS data. This is considered the best address match, because it matched directly against the USPS list of addresses. S is returned for a small number of addresses when the matched address has a blank ZIP + 4.
Thhh	Matched to a street segment record.
Uhhh	Matched to USPS data but cannot resolve the ZIP + 4 code without the firm name or other information. CASS mode returns an E023 (multiple match) error code.
Vhhh	Matched to MLD and DVDMLDR using Reverse PBKey Lookup. For match code values, see **Definitions for the Reverse PBKey Lookup "Vhhh" return codes** on page 175.

Code	Description
Xhhh	Matched to an intersection of two streets, for example, "Clay St & Michigan Ave." The first hex digit refers to the last line information, the second hex digit refers to the first street in the intersection, and the third hex digit refers to the second street in the intersection. **Note:** The USPS does not allow intersections as a valid deliverable address.
Yhhh	Same as Xhhh, but an alias name record was used for one or both streets.
Z	No address given, but verified the provided ZIP Code .

Definitions for 1st-3rd hex digit match code values

The table below contains the description of the hex digits for the match code values.

Note: The third hex digit is only populated for intersection matches or as part of the Extended Match Code.

- For intersection matches, use the table below for the 3rd hex digit definitions.
- For Extended Match Code, see **Definitions for Extended Match Code (3rd hex digit values)** on page 173.

Code	In first hex position means:	In second and third hex position means:
0	No change in last line.	No change in address line.
1	ZIP Code changed.	Street type changed.
2	City changed.	Predirectional changed.
3	City and ZIP Code changed.	Street type and predirectional changed.
4	State changed.	Postdirectional changed.

Code	In first hex position means:	In second and third hex position means:
5	State and ZIP Code changed.	Street type and postdirectional changed.
6	State and City changed.	Predirectional and postdirectional changed.
7	State, City, and ZIP Code changed.	Street type, predirectional, and postdirectional changed.
8	ZIP + 4 changed.	Street name changed.
9	ZIP and ZIP + 4 changed.	Street name and street type changed.
A	City and ZIP + 4 changed.	Street name and predirectional changed.
B	City, ZIP, and ZIP + 4 changed.	Street name, street type, and predirectional changed.
C	State and ZIP + 4 changed.	Street name and postdirectional changed.
D	State, ZIP, and ZIP + 4 changed.	Street name, street type, and postdirectional changed.
E	State, City, and ZIP + 4 changed.	Street name, predirectional, and postdirectional changed.
F	State, City, ZIP, and ZIP + 4 changed.	Street name, street type, predirectional, and postdirectional changed.

Definitions for Extended Match Code (3rd hex digit values)

As mentioned in **Extended Match Codes** on page 50, when `Option.ExtendedMatchCode` is set to "`Y`", additional information is returned about any changes in the house number, unit number and unit type fields in the matched address, as well as whether there was address information that was ignored. This additional information is provided in a 3rd hex digit that is appended to match

codes for address-level matches only - A, G, H, J, Q, R, S, T or U (see **Match Code Definitions** on page 170).

"Address information ignored" is specified when any of these conditions apply:

- The output address has content in the `AdditionalInputData` field.
- The output address has a second address line (`AddressLine2`).
- The input address is a dual address (two complete addresses in the input address). For example, "4750 Walnut St. P.O Box 50".
- The input last line has extra information that is not a city, state or ZIP Code, and is ignored. For example, "Boulder, CO 80301 USA", where "USA" is ignored when matching.

The table below provides the descriptions for the Extended Match Code 3rd hex digit return values:

> **Note:** For Auxiliary file matches, the 3rd hex digit is always "0".

Code	In 3rd hex position means:
0	Matched on all address information on line, including Unit Number and Unit Type if included.
1	Matched on Unit Number and Unit Type if included. Extra information on address line ignored. Extra information on address line ignored. Extra information not considered for matching moved to `AddressLine2` or `AdditionalInputData` field.
2	Matched on Unit Number. Unit Type changed.
3	Matched on Unit Number. Unit Type changed. Extra information on address line ignored. Extra information on address line ignored. Extra information not considered for matching moved to `AddressLine2` or `AdditionalInputData` field.
4	Unit Number changed or ignored.
5	Unit Number changed or ignored. Extra information on address line ignored. Extra information on address line ignored. Extra information not considered for matching moved to `AddressLine2` or `AdditionalInputData` field.
6	Unit Number changed or ignored. Unit Type changed or ignored.
7	Unit Number changed or ignored. Unit Type changed or ignored. Extra information on address line ignored. Extra information on address line ignored. Extra information not considered for matching moved to `AddressLine2` or `AdditionalInputData` field.
8	Matched on Unit Number and Unit Type if included. House Number changed or ignored.

Code	In 3rd hex position means:
9	Matched on Unit Number and Unit Type if included. House Number changed or ignored. Extra information on address line ignored. Extra information not considered for matching moved to `AddressLine2` or `AdditionalInputData` field.
A	Matched on Unit Number. Unit Type changed. House Number changed or ignored.
B	Matched on Unit Number. Unit Type changed. House Number changed or ignored. Extra information on address line ignored. Extra information not considered for matching moved to `AddressLine2` or `AdditionalInputData` field.
C	House Number changed or ignored. Unit Number changed or ignored.
D	House Number changed or ignored. Unit Number changed or ignored. Extra information on address line ignored. Extra information on address line ignored. Extra information not considered for matching moved to `AddressLine2` or `AdditionalInputData` field.
E	House Number changed or ignored. Unit Number changed or ignored. Unit Type changed or ignored.
F	House Number changed or ignored. Unit Number changed or ignored. Unit Type changed or ignored. Extra information on address line ignored. Extra information on address line ignored. Extra information not considered for matching moved to `AddressLine2` or `AdditionalInputData` field.

Definitions for the Reverse PBKey Lookup "Vhhh" return codes

The table below lists the "Vhhh" hex digit values returned with Reverse PBKey Lookup. For more information, see **ReversePBKeyLookup** on page 135.

> **Note:** When there are one or more address variations for a Reverse PBKey Lookup, the match code returned is always "`V000`".

Match Code	Definition
V000	Match made using input pbKey. One Standard or Enhanced point address result returned depending on license.
V001	Match made using input pbKey. Multiple Standard and/or Enhanced point address variations results returned depending on license.
V002	Match made using input pbKey. One Standard, some Enhanced point address variations results returned depending on license.

Match Code	Definition
V003	Match made using input pbKey. Multiple Standard, some Enhanced point address variations results depending on license.

Match Codes for No Match - Definitions for "Ennn" return codes

The table below describes the values returned when the application cannot find a match or an error occurs.

Code	"nnn" values	Description
Ennn		Indicates an error, or no match. This can occur when the address entered does not exist in the database, or the address is badly formed and cannot be parsed correctly. The last three digits of an error code indicate which parts of an address the application could not match to the database.
	nnn = 000	No match made.
	nnn = 001	Low level error.
	nnn = 002	Could not find data file.
	nnn = 003	Incorrect GSD file signature or version ID.
	nnn = 004	GSD file out of date. Only occurs in CASS mode.
	nnn = 010	No city and state or ZIP Code found.
	nnn = 011	Input ZIP not in the directory.
	nnn = 012	Input city not in the directory.
	nnn = 013	Input city not unique in the directory.

Code	"nnn" values	Description
	nnn = 014	Out of licensed area. Only occurs if using Group 1 licensing technology.
	nnn = 015	Record count is depleted and license has expired.
	nnn = 020	No matching streets found in directory.
	nnn = 021	No matching cross streets for an intersection match.
	nnn = 022	No matching segments.
	nnn = 023	Unresolved match.
	nnn = 024	No matching segments. (Same as 022.)
	nnn = 025	Too many possible cross streets for intersection matching.
	nnn = 026	No address found when attempting a multiline match.
	nnn = 027	Invalid directional attempted.
	nnn = 028	Record also matched EWS data, therefore the application denied the match.
	nnn = 029	No matching range, single street segment found.
	nnn = 030	No matching range, multiple street segments found.
	nnn = 040	No match found using input PBKey with ReversePBKeyLookup.

Code	"nnn" values	Description
	nnn = 041	Not licensed to return Enhanced point address(es) found for input PBKey. Additional ReversePBKeyLookup license option required to return results.

Location Codes

Location codes indicate the locational accuracy of the assigned geocode. Note that an accurately placed candidate is not necessarily an ideal candidate. Examine the match codes and/or result codes in addition to location codes to best evaluate the overall quality of the candidate.

A Location Code of "E" indicates a location code is not available. This usually occurs when you have requested ZIP Code centroids of a high quality, and one is not available for that match. It can occur infrequently when the Enterprise Geocoding Module does not have a 5-digit centroid location. The Enterprise Geocoding Module can also return an "E" location code when it cannot standardize an input address and there is no input ZIP Code. In this case, do not assume the ZIP Code returned with the non-standardized address is the correct ZIP Code because the Enterprise Geocoding Module did not standardize the address; therefore, the Enterprise Geocoding Module does not return geocoding or Census Block information.

Address Location Codes

Location codes that begin with an "A" are address location codes. Address location codes indicate a geocode made directly to a street network segment (or two segments, in the case of an intersection).

An address location code has the following characters.

1st character	Always an "A" indicating an address location.

2nd character	May be one of the following:	
	C	Interpolated address point location
	G	Auxiliary file data location

	I	Application infers the correct segment from the candidate records
	P	Point-level data location
	R	Location represents a ranged address
	S	Location on a street range
	X	Location on an intersection of two streets

3rd and 4th character	Digit indicating other qualities about the location.

Table 1: Address Location Codes

Code	Description
AGn	Indicates an Auxiliary file for a geocode match where n is one of the following values:
n = 0	The geocode represents the center of a parcel or building.
n = 1	The geocode is an interpolated address along a segment.
n = 2	The geocode is an interpolated address along a segment, and the side of the street cannot be determined from the data provided in the auxiliary file record.
n = 3	The geocode is the midpoint of the street segment.

Code	Description
`APnn`	Indicates a point-level geocode match representing the center of a parcel or building, where nn is one of the following values:
`nn = 00`	User Dictionary centroid. Geocode returned by a User Dictionary.
`nn = 02`	Parcel centroid Indicates the center of an accessor's parcel (tract or lot) polygon. When the center of an irregularly shaped parcel falls outside of its polygon, the centroid is manually repositioned to fall inside the polygon as closely as possible to the actual center.
`nn = 04`	Address points Represents field-collected GPS points with field-collected address data.
`nn = 05`	Structure centroid Indicates the center of a building footprint polygon, where the building receives mail or has telephone service. Usually a residential address consists of a single building. For houses with outbuildings (detached garages, shed, barns, etc.), only the residences have a structure point. Condominiums and duplexes have multiple points for each building. Larger buildings, such as apartment complexes, typically receive mail at one address for each building and therefore individual apartments are not represented as discrete structure points. Shopping malls, industrial complexes, and academic or medical center campuses where one building accepts mail for the entire complex are represented as one point. When addresses are assigned to multiple buildings within one complex, each addressed structure is represented by a point. If the center of a structure falls outside of its polygon, the center is manually repositioned to fall inside the polygon.

Code	Description
nn = 07	**Manually placed** Address points are manually placed to coincide with the midpoint of a parcel's street frontage at a distance from the center line.
nn = 08	**Front door point** Represents the designated primary entrance to a building. If a building has multiple entrances and there is no designated primary entrance or the primary entrance cannot readily be determined, the primary entrance is chosen based on proximity to the main access street and availability of parking.
nn = 09	**Driveway offset point** Represents a point located on the primary access road (most commonly a driveway) at a perpendicular distance of between 33-98 feet (10-30 meters) from the main roadway.
nn = 10	**Street access point** Represents the primary point of access from the street network. This address point type is located where the driveway or other access road intersects the main roadway.
nn = 21	**Base parcel point** When unable to match to an input unit number, or when the unit number is missing from an address location with multiple units, the "base" parcel information is returned, the address is not standardized to a unit number, and additional information, such as an Assessor's Parcel Number, is not returned.
nn = 22	**Backfill address point** The precise parcel centroid is unknown. The address location assigned is based on two known parcel centroids.

Code	Description
nn = 23	**Virtual address point** The precise parcel centroid is unknown. The address location assigned is relative to a known parcel centroid and a street segment end point.
nn = 24	**Interpolated address point** The precise parcel centroid is unknown. The address location assigned is based on street segment end points.
AIn	The correct segment is inferred from the candidate records at match time.
ASn	House range address geocode. This is the most accurate street interpolated geocode available.

AIn, ASn and ACnh share the same values for the 3rd character "n" as follows:

n = 0	Best location.
n = 1	Street side is unknown. The Census FIPS Block ID is assigned from the left side; however, there is no assigned offset and the point is placed directly on the street.
n = 2	Indicates one or both of the following: • The address is interpolated onto a TIGER segment that did not initially contain address ranges. • The original segment name changed to match the USPS spelling. This specifically refers to street type, predirectional, and postdirectional. **Note:** Only the second case is valid for non-TIGER data because segment range interpolation is only completed for TIGER data.

Code	Description
`n = 3`	Both 1 and 2.
`n = 7`	Placeholder. Used when starting and ending points of segments contain the same value and shape data is not available.
ACnh	Indicates a point-level geocode that is interpolated between 2 parcel centroids (points), a parcel centroid and a street segment endpoint, or 2 street segment endpoints.

The ACnh 4th character "h" values are as follows:

`h = 0`	Represents the interpolation between two points, both coming from User Dictionaries.
`h = 1`	Represents the interpolation between two points. The low boundary came from a User Dictionary and the high boundary, from a non-User Dictionary.
`h = 2`	Represents the interpolation between one point and one street segment end point, both coming from User Dictionaries.
`h = 3`	Represents the interpolation between one point (low boundary) and one street segment end point (high boundary). The low boundary came from a User Dictionary and the high boundary from a non-User Dictionary.
`h = 4`	Represents the interpolation between two points. The low boundary came from a non-User Dictionary and the high boundary from a User Dictionary.
`h = 5`	Represents the interpolation between two points, both coming from non-User Dictionaries.

Code	Description
h = 6	Represents the interpolation between one point (low boundary) and one street segment end point (high boundary). The low boundary came from a non-User Dictionary and the high boundary from a User Dictionary.
h = 7	Represents the interpolation between one point and one street segment end point and both came from non-User Dictionaries.
h = 8	Represents the interpolation between one street segment end point andone point, both coming from User Dictionaries.
h = 9	Represents the interpolation between one street segment end point (low boundary) andone point (high boundary). The low boundary came from a User Dictionary and the high boundary from a non-User Dictionary.
h = A	Represents the interpolation between two street segment end points, both coming from User Dictionaries.
h = B	Represents the interpolation between two street segment end points. The low boundary came from a User Dictionary and the high boundary from a non-User Dictionary.
h = C	Represents the interpolation between one street segment end point (low boundary) and one point (high boundary). The low boundary came from a non-User Dictionary and the high boundary from a User Dictionary.
h = D	Represents the interpolation between one street segment end point and one point, both coming from non-User Dictionary.

Code	Description
`h = E`	Represents the interpolation between two street segment end points. The low boundary came from a non-User Dictionary and the high boundary from a User Dictionary.
`h = F`	Represents the interpolation between two street segment end points, both coming from non-User Dictionaries.
`ARn`	Ranged address geocode, where "n" is one of the following:
`n = 1`	The geocode is placed along a single street segment, midway between the interpolated location of the first and second input house numbers in the range.
`n = 2`	The geocode is placed along a single street segment, midway between the interpolated location of the first and second input house numbers in the range, and the side of the street is unknown. The Census FIPS Block ID is assigned from the left side; however, there is no assigned offset and the point is placed directly on the street.
`n = 4`	The input range spans multiple USPS segments. The geocode is placed on the endpoint of the segment which corresponds to the first input house number, closest to the end nearest the second input house number.
`n = 7`	Placeholder. Used when the starting and ending points of the matched segment contain the same value and shape data is not available.
`AXn`	Intersection geocode, where n is one of the following:
`n = 3`	Standard single-point intersection computed from the center lines of street segments.

Code	Description
n = 8	Interpolated (divided-road) intersection geocode. Attempts to return a centroid for the intersection.

Street Centroid Location Codes

Location codes that begin with "C" are street centroid location codes. Street centroid location codes indicate the Census ID accuracy and the position of the geocode on the returned street segment. Street centroids may be returned if the street centroid fallback option is enabled and an address-level geocode could not be determined.

A street centroid location code has the following characters.

1st character	Always "C" indicating a location derived from a street segment.
2nd character	Census ID accuracy based on the search area used to obtain matching Street Segment.
3rd character	Location of geocode on the returned street segment.

The table below contains the values and descriptions for the location codes.

Character position	Code	Description
2nd Character		
	B	Block Group accuracy (most accurate). Based on input ZIP Code.
	T	Census Tract accuracy. Based on input ZIP Code.

Character position	Code	Description
	C	Unclassified Census accuracy. Normally accurate to at least the County level. Based on input ZIP Code.
	F	Unknown Census accuracy. Based on Finance area.
	P	Unknown Census accuracy. Based on input City.
3rd Character		
	C	Segment Centroid.
	L	Segment low-range end point.
	H	Segment high-range end point.

ZIP + 4 Centroid Location Codes

Location codes that begin with a "Z" are ZIP + 4 centroid location codes. ZIP + 4 centroids indicate a geocode could not be determined for the address, so the location of the center of the address's ZIP + 4 was returned instead. ZIP + 4 centroid location codes indicate the quality of two location attributes: Census ID accuracy and positional accuracy.

A ZIP + 4 centroid location code has the following characters.

1st character	Always "Z" indicating a location derived from a ZIP centroid.
2nd character	Census ID accuracy.

3rd character	Location type.
4th character	How the location and Census ID was defined. Provided for completeness, but may not be useful for most applications.

Table 2: ZIP + 4 Centroid Location Codes

Character Position	Code	Description
2nd Character		
	B	Block Group accuracy (most accurate).
	T	Census Tract accuracy.
	C	Unclassified Census accuracy. Normally accurate to at least the County level.
3rd Character		
	5	Location of the Post Office that delivers mail to the address, a 5-digit ZIP Code centroid, or a location based upon locale (city). See the 4th character for a precise indication of locational accuracy.
	7	Location based upon a ZIP + 2 centroid. These locations can represent a multiple block area in urban locations, or a slightly larger area in rural settings.

Character Position	Code	Description
	9	Location based upon a ZIP + 4 centroid. These are the most accurate centroids and normally place the location on the correct block face. For a small number of records, the location may be the middle of the entire street on which the ZIP + 4 falls. See the 4th character for a precise indication of locational accuracy.
4th Character		
	A	Address matched to a single segment. Location assigned in the middle of the matched street segment, offset to the proper side of the street.
	a	Address matched to a single segment, but the correct side of the street is unknown. Location assigned in the middle of the matched street segment, offset to the left side of the street, as address ranges increase.
	B	Address matched to multiple segments, all segments have the same Block Group. Location assigned to the middle of the matched street segment with the most house number ranges within this ZIP + 4. Location offset to the proper side of the street.
	b	Same as methodology B except the correct side of the street is unknown. Location assigned in the middle of the matched street segment, offset to the left side of the street, as address ranges increase.

Character Position	Code	Description
	C	Address matched to multiple segments, with all segments having the same Census Tract. Returns the Block Group representing the most households in this ZIP + 4. Location assigned to t he middle of the matched street segment with the most house number ranges within this ZIP + 4. Location offset to the proper side of the street.
	c	Same as methodology C except the correct side of the street is unknown. Location assigned in the middle of the matched street segment, offset to the left side of the street, as address ranges increase.
	D	Address matched to multiple segments, with all segments having the same County. Returns the Block Group representing the most households in this ZIP + 4. Location assigned to the middle of the matched street segment with the most house number ranges within this ZIP + 4. Location offset to the proper side of the street.
	d	Same as methodology D except the correct side of the street is unknown. Location assigned in the middle of the matched street segment, offset to the left side of the street, as address ranges increase.
	E	Street name matched; no house ranges available. All matched segments have the same Block Group. Location placed on the segment closest to the center of the matched segments. In most cases, this is on the mid-point of the entire street.

Character Position	Code	Description
	F	Street name matched; no house ranges available. All matched segments have the same Census Tract. Location placed on the segment closest to the center of the matched segments. In most cases, this is on the mid-point of the entire street.
	G	Street name matched (no house ranges available). All matched segments have the same County. Location placed on the segment closest to the center of the matched segments. In most cases, this is on the mid-point of the entire street.
	H	Same as methodology G, but some segments are not in the same County. Used for less than .05% of the centroids.
	I	Created ZIP + 2 cluster centroid as defined by methodologies A, a, B, and b. All centroids in this ZIP + 2 cluster have the same Block Group. Location assigned to the ZIP + 2 centroid.
	J	Created ZIP + 2 cluster centroid as defined by methodologies A, a, B, b, C, and c. All centroids in this ZIP + 2 cluster have the same Census Tract. Location assigned to the ZIP + 2 centroid.
	K	Created ZIP + 2 cluster centroid as defined by methodologies A, a, B, b, C, c, D, and d. Location assigned to the ZIP + 2 centroid.

Character Position	Code	Description
	L	Created ZIP + 2 cluster centroid as defined by methodology E. All centroids in this ZIP + 2 cluster have the same Block Group. Location assigned to the ZIP + 2 centroid.
	M	Created ZIP+2 cluster centroid as defined by methodology E and F. All centroids in this ZIP + 2 cluster have the same Census Tract. Location assigned to the ZIP + 2 centroid.
	N	Created ZIP + 2 cluster centroid as defined by methodology E, F, G, and H. Location assigned to the ZIP + 2 centroid.
	O	ZIP Code is obsolete and not currently used by the USPS. Historic location assigned.
	V	Over 95% of addresses in this ZIP Code are in a single Census Tract. Location assigned to the ZIP Code centroid.
	W	Over 80% of addresses in this ZIP Code are in a single Census Tract. Reasonable Census Tract accuracy. Location assigned to the ZIP Code centroid.
	X	Less than 80% of addresses in this ZIP Code are in a single Census Tract. Census ID is uncertain. Location assigned to the ZIP Code centroid.
	Y	Rural or sparsely populated area. Census code is uncertain. Location based upon the USGS places file.

Character Position	Code	Description
	Z	P.O. Box or General Delivery addresses. Census code is uncertain. Location based upon the Post Office location that delivers the mail to that address.

Geographic Centroid Location Codes

Location codes that begin with "G" are geographic centroid location codes. Geographic centroids may be returned if the street centroid fallback option is enabled and an address-level geocode could not be determined. Geographic centroid location codes indicate the quality of a city, county, or state centroid.

A geographic centroid location code has the following characters.

1st character	Always "G" indicating a location derived from a geographic centroid.

2nd character	Geographic area type. One of the following:
	M Municipality (for example, a city)
	C County
	S State

8 - Encountering False Positives

In this section

What is a False-Positive?

To prevent the generation of address lists, the DPV and LACS^{Link} databases include false-positive records. False-positive records are artificially manufactured addresses that reside in a false-positive table. For each negative response that occurs in a DPV or LACS^{Link} query, a query is made to the false-positive table. A match to this table (called a false-positive match) disables your DPV or LACS^{Link} key. In batch processing the job that contains the violation will complete successfully but you will not be able to run any subsequent jobs that use DPV or LACS^{Link} until you report the violation and obtain a key to reactivate DPV or LACS^{Link}.

> **Note:** The term "seed record violation" is also used to refer to encountering false positive records. The two terms mean the same thing.

Reporting DPV False-Positive Violations

Spectrum^{TM} Technology Platform indicates a false-positive match via messages in the server log.

You can report the violation and obtain a restart key by following these steps.

1. In your browser, go to http://<yourserver>:<port>/<product code>/dpv.jsp. For example, http://localhost:8080/unc/dpv.jsp for the Universal Addressing Module and http://localhost:8080/geostan/dpv.jsp for the Enterprise Geocoding Module.
2. Enter the mailer's information into each field. The number in parentheses after each field name indicates the maximum length of the field.
3. Click **Submit** when you're done. A **File Download** dialog will appear.
4. Click **Save** to save the file to your computer. A **Save As** dialog will appear.
5. Specify a file name and location on your local hard drive (for example `c:\DPVSeedFile.txt`) and click **Save**.
6. Go to **www.g1.com/support** and log in.
7. Click **DPV & LACS^{Link} False Positive**.
8. Follow the on-screen instructions to attach your seed file and obtain a restart key.

DPV False Positive Header File Layout

The USPS® has determined the required layout of the DPV false-positive header file, which is currently defined as a fixed-length file containing two or more 180-byte records. The first record must always be the header record, whose layout is shown below.

Table 3: DPV False-Positive Header Record Layout

Position	Length	Description	Format
1-40	40	Mailer's company name	Alphanumeric
41-98	58	Mailer's address line	Alphanumeric
99-126	28	Mailer's city name	Alphanumeric
127-128	2	Mailer's state abbreviation	Alphabetic
129-137	9	Mailer's 9-digit ZIP Code	Numeric
138-146	9	Total Records Processed	Numeric
147-155	9	Total Records DPV Matched	Numeric
156-164	9	Percent Match Rate to DSF	Numeric
165-173	9	Percent Match Rate to ZIP + 4®	Numeric
174-178	5	Number of ZIP Codes on file	Numeric
179-180	2	Number of False-Positives	Numeric

The trailer record contains information regarding the DPV false-positive match. There must be one trailer record added to the false-positive file for every DPV false-positive match. The layout is shown below.

Table 4: DPV False-Positive Trailer Record Layout

Position	Length	Description	Format
1-2	2	Street predirectional	Alphanumeric
3-30	28	Street name	Alphanumeric
31-34	4	Street suffix abbreviation	Alphanumeric
35-36	2	Street postdirectional	Alphanumeric
37-46	10	Address primary number	Alphanumeric
47-50	4	Address secondary abbreviation	Alphanumeric
51-58	8	Address secondary number	Numeric
59-63	5	Matched ZIP Code	Numeric
64-67	4	Matched ZIP + 4®	Numeric
68-180	113	Filler	Spaces

Reporting LACS/Link False-Positive Violations

Spectrum™ Technology Platform indicates a false-positive match via messages in the server log. Batch jobs will fail if a false-positive match occurs and client/server calls will throw an exception.

> **Note:** The term "seed record violation" is also used to refer to encountering false positive records. The two terms mean the same thing.

1. In your browser, go to http://<ServerName>:<port>/<product code>/lacslink.jsp. For example, http://localhost:8080/unc/lacslink.jsp for the Universal Addressing Module and http://localhost:8080/geostan/lacslink.jsp for the Enterprise Geocoding Module.
2. Enter the mailer's information into each field. The number in parentheses after the field name indicates the maximum length of the field. Click **Submit** when you're done. A **File Download** dialog will appear.
3. Click **Save** to save the file to your computer. A **Save As** dialog will appear.
4. Specify a file name and location on your local hard drive (for example `c:\lacslink.txt`) and click **Save**.
5. Go to **www.g1.com/support** and log in.
6. Click **DPV & LACS**^{Link} **False Positive**.
7. Follow the on-screen instructions to attach your seed file and obtain a restart key.

Appendix

In this section

A - User Dictionary Writer

In this section

User Dictionary Writer for US Enterprise Geocoding Module

The User Dictionary Writer is a tool for Windows and a script for Unix and Linux, which are provided with your US Enterprise Geocoding Module installation. It creates a User Dictionary that can be used as a database resource when address matching.

User Dictionary Writer Installation Directory

The location of the User Dictionary Writer installation folder is:

- On Windows - *SpectrumDirectory*`\server\modules\geostan\bin\udwriter`
- On Unix and Linux - *SpectrumDirectory*`/server/modules/geostan/bin/udwriter`

where *SpectrumDirectory* refers to the directory where you have installed the Spectrum™ Technology Platform server.

Requirements

Prior to running the User Dictionary Writer script to generate your user dictionary, the following steps must be performed:

- Verify you have a 64-bit JVM installed.
- Set the `JAVA_HOME` environment variable to point to the jdk installation directory.
- Create the input `.TAB` file that contains your address record data using MapInfo Pro.
- For Unix and Linux only: Edit the `udCreation.properties` file.
- Edit the `USA_DataManagerSettings.properties` file.

The following sections cover these steps in more detail.

Setting the JAVA_HOME environment variable

Prior to using the User Dictionary Writer, you will need to set the path to the JAVA_HOME environment variable.

On Windows

1. Locate the Java installation directory.
2. Go to **Start** > **Control Panel** > **System** > **Advanced System Settings**
3. Select the **Advanced** tab, then click the **Environment Variables** button.
4. Under **System Variables**, click **New**.
5. In the **New System Variable** window, under **Variable name**, enter *JAVA_HOME*.

   Under **Variable value**, enter where Java is installed. For example, `C:\Program Files\Java\jdk1.7.0_07`
6. Click **OK** from the **New System Variable** window, as well as on the **Advanced** tab dialog.

On Unix and Linux

1. Locate the `udbuild.sh` file in the User Dictionary installation directory.
2. Use a text editor to open the `udbuild.sh` file.
3. In the `JAVA_HOME` field, add the file path to the JDK folder.
4. Save the file and exit.

Creating the source data

This chapter includes information about creating the source data including the required fields and input files needed to create a User Dictionary.

Understanding User Dictionary capabilities and requirements

> **Note:** User Dictionaries are not for use with CASS geocoding.

The capabilities of User Dictionaries and the basic requirements for creating them are as follows

- All fields supported by normal street geocoding can be included in User Dictionaries.
- Landmarks and place names are supported in User Dictionaries. Postal or geographic centroid geocoding are not supported in User Dictionaries.
- User Dictionaries support address browsing using partial street names or landmarks and place names.
- GSDs are necessary to create the User Dictionary. This is because the GSDs have some internal structure that must be available when creating a User Dictionary.

The results from a User Dictionary are similar to that from the GSD. For address matches where the first letter of the match code would be 'S', a User Dictionary match has the letter 'J'. For example: SE9 is a match code for a match that comes from a GSD, while JE9 is for a match that comes from a User Dictionary. For more information, see the match code descriptions in **Match Codes for U.S. Geocoding**.

The output field, `StreetDataType`, returned in the Default Output group, indicates if a User Dictionary was used to geocode the address by returning `User Dictionary`.

Source data requirements

The source data for User Dictionaries includes street data but can also include place names and intersections.

To create a User Dictionary, your source data must conform to these requirements:

• Source records must include required fields, and these fields are mapped during the User Dictionary creation process. If a value of a required field is empty for a particular record, then that record will not be imported into the User Dictionary. The MapInfo table must contain specific fields. These input fields are described in **Required input fields** on page 203.
• Source records must be in a MapInfo table (`.TAB` file). For more information, see **MapInfo Pro files** on page 203.
• Segments must have two or more defined end points to be loaded into a User Dictionary. Segments without end points are ignored.
• Segments that make up intersections must have one or more end points in the intersection for the US geocoder to recognize it as an intersection. Source records can be either point objects or segments.
• Each row in the table is equivalent to a street segment.

MapInfo Pro files

MapInfo Pro allows you to use address record data that was created in other file formats and converts this information into the `.TAB` file format. The conversion process also generates other files that need to be placed in the same directory as the `.TAB` file:

• *filename*`.DAT` - Contains the attribute data for each record.
• *filename*`.MAP` - Contains geometry information.
• *filename*`.ID` - Contains the spatial index.
• *filename*`.IND` - This file is created if an index is created from one of the attribute fields.

For more information about MapInfo Pro, see the *MapInfo Pro User Guide* on **http://www.pbinsight.com/support/product-documentation/details/mapinfo-pro**.

Required input fields

You must specify the field names in the MapInfo table (TAB file) in order for the table to be translated into a User Dictionary. Certain fields are required and must be present in the MapInfo table. Other fields are optional, but are strongly recommended because there may be negative consequences

if they are omitted. This is described in **Optional (recommended) input fields** on page 204. If any of the required fields are missing, a missing field error code is returned.

The following table describes the required input fields.

Required fields	Description	Maximum field length
Left start address	Start of address range on left side of street.	10
Right start address	Start of address range on right side of street.	10
Left end address	End of address range on left side of street.	10
Right end address	End of address range on right side of street.	10
Street name	Name of street.	30
State abbreviation	Two-character state abbreviation.	2
Left ZIP Code	Zip Code for left side of street.	5
Right ZIP Code	Zip Code for right side of street.	5

Optional (recommended) input fields

The `Left and Right Odd/Even Indicator` fields are used to specify whether the sides of the street segment contain odd or even address ranges. Although these indicators are not required for creating a User Dictionary, it is important to use the `Odd/Even Indicators` when your data contains odd/even address numbers.

When the `Odd/Even Indicator` is specified, but is inconsistent with address numbers, the indicator is set to `Both`.

When the `Odd/Even Indicator` is not specified and both `Start Address` and `End Address` have values, the indicator is set to `Both`, unless the start and end address numbers are the same number. In that case, the indicator is set to `Odd` if the address numbers are odd, and set to `Even` if the address numbers are even.

When the `Odd/Even Indicator` is not specified and both `Start Address` and `End Address` have values, the indicator is set to `Both` (odd and even).

> **Note:** If your table contains odd/even indicator information, we strongly recommend that you use the `Odd/Even Indicator` fields. These fields ensure that your geocoded addresses are located on the correct side of the street. Omitting the fields when your data contains odd/even information may produce incorrect results.

The following table describes the optional input fields.

Optional fields	Description		Maximum field length
Left Odd/Even Indicator*	Left side of the street contains only odd or even address ranges.		1
	O	Odd	
	E	Even	
	B	Both	
Right Odd/Even Indicator*	Right side of the street contains only odd or even address ranges.		1
	O	Odd	
	E	Even	
	B	Both	
City*	City name		28
Left ZIP+4 Code	4-digit ZIP+4 add-on for left side of street.		4
Right ZIP+4 Code	4-digit ZIP+4 add-on for right side of street.		4
Left Census Block	Census Block ID for left side of street.		15
Right Census Block	Census Block ID for right side of street.		15
Place Name	Place name		40

* These fields are highly recommended.

User Dictionary file names and formats

The following list specifies file name and formatting requirements:

- Each User Dictionary has a base name of eight characters or fewer.
- Each User Dictionary resides in its own directory.
- The maximum length of a path to a User Dictionary is 1024 characters.
- The ZIP Code range in the MapInfo table for a User Dictionary is unlimited.

Because each User Dictionary resides in its own directory, User Dictionaries may share the same name. However, it is generally good practice to use a unique name for each User Dictionary

Some of the output files are tied to the base name. The other output files have constant names. For example, the output files for a dictionary called `ud1` are the following:

```
postinfo.jdr
postinfo.jdx
lastline.jdr
post2sac.mmj
geo2sac.mmj
sac2fn_ud.mmj
ud1.jdr
ud1.jdx
ud1.bdx
```

If your data includes place names, the dictionary would contain the following files:

```
ud1.pdx
ud1.pbx
```

The dictionary would also contain these log files:

```
ud1.log
ud1.err
```

Additional User Dictionary considerations

See the following topics for more information when working with User Dictionaries.

Data Access License

You must still have a valid access license to the data contained in the GSD when you are geocoding against your User Dictionary. For example, if you create a dictionary of New York streets and addresses, you must purchase the New York or entire U.S. GSD.

Use without GSD data files

To utilize a User Dictionary without the use of GSDs, the files listed below are required:

- `ctyst.dir`—The USPS City-State table
- `parse.dir`—Address elements and abbreviations dictionary
- `finmbr.dat`—Spatial query file

To perform postal centroid geocoding, in addition to a GSD or a User Dictionary and the files listed above, the files listed below are necessary::

- `us.z9`—Postal centroid information
- `cbsac.dir`—Required only if county names or CBSA/CSA data are needed

CASS standards

You cannot geocode to CASS standards using a User Dictionary. This also means that the ParcelPrecision Dictionary cannot be used during CASS geocoding.

Address Range order

The US geocoder determines the order of the address range based on a comparison of the start and end addresses. The comparison produces these results:

• If the end is greater than the start, the range is ascending.
• If the start is greater than the end, the range is descending.
• If the start is equal to the end, the range is ascending.

Street intersections and User Dictionaries

When geocoding to street intersections with a User Dictionary, the US geocoder cannot recognize the intersections if one or more of the segments that make up the intersection does not have an end point at the intersection. This can happen when you create the User Dictionary from a customized street table in which some segments that terminate at intersections do not have end points.

The figure below shows an intersection in a User Dictionary that does not have end points for all segments; this will not be recognized by the US geocoder as an intersection.
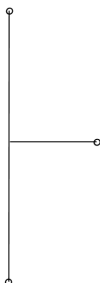
The figure below shows an intersection in a `TIGER`-based GSD which includes end points for all segments. The US geocoder geocodes to this intersection.
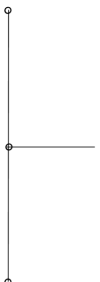
Using User Dictionaries with address point interpolation

An important part of the process of creating a User Dictionary is to specify a mapping of fields from your source data. There are two main categories of data fields: required and optional.

Of the optional fields, there are two that have an impact on the address point interpolation feature. These are the `Left Odd/Even Indicator` and `Right Odd/Even Indicator` fields. If these are not populated, the results from address point interpolation is less accurate.

Be aware that the aforementioned fields are not populated by source data obtained via MapInfo Pro. You must modify the source `TAB` file by adding the `Left Odd/Even Indicator` and `Right Odd/Even Indicator` fields, and create queries to populate them. Source data obtained from other products, or your own data, may have similar issues.

To add the `Left Odd/Even Indicator` and `Right Odd/Even Indicator` fields to a source `TAB` file, you must add them, then run a series of SQL update queries to populate them. The fields should be filled in with `O` (odd), `E` (even), or `B` (both). Below are the steps for adding these fields:

1. Add two 1-char columns to your `TAB` file. Naming each column, for example, `Ind_Right` and `Ind_Left`.
2. Perform the following updates to populate these fields:

   - Update `<tablename>`. Set `Ind_Left="E"`, `Ind_Right="O"`. Where `From_Left mod 2=0` AND `To_Left mod 2=0`
   - Update `<tablename>`. Set `Ind_Left="O"`, `Ind_Right="E"`. Where `From_Left mod 2=1` AND `To_Left mod 2=1`
   - Update `<tablename>`. Set `Ind_Left="B"`, `Ind_Right="B"`. Where `From_Left=""` AND `To_Left=""`

   Note: These example queries are simplified for illustrative purposes. Your actual queries may need to be more complex.

Editing the udCreation.properties file (Unix and Linux systems only)

The `udCreation.properties` contains file path and address variables that need to be updated prior to running the User Dictionary Writer script on Unix and Linux systems.

1. Locate the `udCreation.properties` file in the `/udwriter/config` folder.
2. Open the file using a text editor.
3. Modify these fields as needed:

startRow, endRow	Set each to 0 to do all rows in the table. Alternatively set to do a subset of the table. For example, startRow = 12 and endRow = 20. This will build the UD only using rows 12-20.
street_inputTable	The full path with the name of the input Mapinfo table.

street_outputFile	The base name for the User Dictionary files.
street_outputFilePath geoInfo_outputFilePath postInfo_outputFilePath post2sac_outputFilePath lastLine_outputFilePath geo2sac_outputFilePath	The output path to the destination directory of the User Dictionary to be built.
statusLog	The output path to the destination directory of the User Dictionary to be built with the log file name. For example, `/dev/opt/udout/testud.log`
errorLog	The output path to the destination directory of the User Dictionary to be built with the error file name. For example, `/dev/opt/udout/testud.err`

The fields below are required for a User Dictionary with Point addresses:

street_mainAddress	Street address without house number or unit information.
street_toLeft	House number.
street_fromLeft	The same value used for `street_toLeft`.
street_leftAreaName1	State abbreviation.
street_leftPostCode1	5-digit ZIP Code.

The fields below are optional for a User Dictionary with Point addresses:

street_leftAreaName3	City or town name.
street_leftCensusBlock	Census Block ID.
street_leftRangeStatus	Odd/Even status.
street_leftHighZipPlus4	ZIP Code+4
street_leftLowZipPlus4	This should be the same as `street_leftHighZipPlus4`.
street_leftPlaceName	Firm name.

The fields below are constant key-value pairs:

```
pointAddressRanges=TRUE
street_leftHighZipPlus4=NONE
geoInfo_outputFile=geoinfo
postInfo_outputFile=postinfo
post2sac_create=NO
postInfo_create=NO
logFiles=NO
COUNTRY=USA
street_create=YES
geo2sac_create=NO
geoInfo_create=NO
useConsole=NO
```

The fields below are required for a User Dictionary with address ranges:

street_mainAddress	Street address without house number or unit information.
street_fromLeft	Left starting house number.
street_toLeft	Left end house number.
street_fromRight	Right starting house number.
street_toRight	Right end house number.
street_leftAreaName1	State abbreviation.
street_rightAreaName1	State abbreviation.
street_leftPostCode1	5-digit ZIP Code.
street_rightPostCode1	5-digit ZIP Code.

The fields below are optional for a User Dictionary with address ranges:

street_leftAreaName3	City or town name.
street_rightAreaName3	City or town name (same as `street_leftAreaName3`).
street_leftCensusBlock	Census Block ID.

street_rightCensusBlock	Census Block ID.
street_leftRangeStatus	Odd/Even status.
street_rightRangeStatus	Odd/Even status.
street_leftLowZipPlus4	4-digit ZIP+4 add-on for left side of street.
street_rightLowZipPlus4	4-digit ZIP+4 add-on for right side of street.
street_leftHighZipPlus4	Same as `street_leftLowZipPlus4.`
street_rightHighZipPlus4	Same as `street_rightLowZipPlus4.`
street_leftPlaceName	Firm name
street_rightPlaceName	Firm name (same as `street_leftPlaceName`)

The fields below are constant key-value pairs for Streets:

```
geoInfo_outputFile=geoinfo
postInfo_outputFile=postinfo
post2sac_create=NO
postInfo_create=NO
logFiles=NO
COUNTRY=USA
street_create=YES
geo2sac_create=NO
geoInfo_create=NO
useConsole=NO
```

4. Save the file.

Editing the USA_DataManagerSettings.properties file

The USA_DataManagerSettings.properties file contains file path and User Dictionary variables that need to be updated prior to using the User Dictionary Writer.

1. Locate the USA_DataManagerSettings.properties file in the /udwriter/config folder.

2. Open the file using a text editor.

   By default the USA_DataManagerSettings.properties file appears as follows:

```
# Properties used by MapMarker UDCreator GUI
#
# Optional - The number of dictionaries to be loaded.  DEFAULT=1
DICTIONARY_COUNT=1

# Required - The path to the highest ranking dictionary (GeoStan data
 files)
# Note that DICTIONARY_PATH is required from 1 to DICTIONARY_COUNT.

# This information must be updated for your installation.  It specifies
 the location of the GeoStan data (using forward slashes).

DICTIONARY_PATH1=/home/g1ntest/udwriter/DVDGDT

# If us.z9 file is in a different folder than DICTIONARY_PATH1 then
uncomment the following line and update to its path (using forward
slashes).
# In addition, change DICTIONARY_COUNT above to 2.
#DICTIONARY_PATH2=C:/Program Files (x86)/Centrus/z9data


# Enter license file path (using forward slashes) and password
LICENSE_FILE=/home/g1ntest/udwriter/all_12345678_eap.lic
PASSWORD=12345678

#######################################
#
#  These items should be left unchanged
#
LIB_PATH=dlls
#
# Option to return all GeoStan data elements in
USA_UserCandidateAddress class
RETURN_EXTENDED_DATA=yes
#
# This optional property controls the number of GeoStan instances
# used to process requests in a multi-threaded environment.
#
GEOSTAN_INSTANCE_COUNT=1
```

3. In the DICTIONARY_PATH1 field, add the path to the location of the US geocoder data files.

4. In the `LICENSE_FILE` and `PASSWORD` fields, add the paths to the location of your license file and password.
5. Review the optional fields and modify as needed.
6. Save the file.

Running the User Dictionary Writer

After you have edited the `USA_DataManagerSettings.properties` file and prepared your source data to meet the requirements for a User Dictionary, you can use the UD Writer to automate the User Dictionary creation process.

On Windows

1. Open a command line window.
2. Change the directory to the UD Writer installation folder.
    SpectrumDirectory`\server\modules\geostan\bin\udwriter`

    where *SpectrumDirectory* refers to the directory where you have installed the Spectrum™"> Technology Platform server.
3. At the command line prompt, enter: `run.bat`
    The UD Writer install wizard launches and will guide you through the installation process.
4. When complete, the **UD Creation Complete** window opens, showing a log file that summarizes processing and lists all the created files. If any errors occurred, these will be indicated in the Error File area of the dialog. Confirm the wizard ran without errors, then click **Exit**.

After successfully creating the User Dictionary, the specified directory contains the files that comprise the User Dictionary (with .mmj, .jdr, .jdx, .sdx, and .bdx file extensions). The log file and error files are also stored in the same directory.

On Unix and Linux

1. Locate the `udbuild.sh` shell script in the UD Writer installation folder:
    SpectrumDirectory`/server/modules/geostan/bin/udwriter`

    Where *SpectrumDirectory* refers to the directory where you have installed the Spectrum™"> Technology Platform server.
2. Run the shell script: `sh udbuild.sh ./config/udCreation.Properties`
    The script will execute for several minutes. When complete, the command line prompt appears.
3. Check the error and log files in the User Dictionary library to verify the User Dictionary files were successfully generated.

    The User Dictionary libary contains the files that comprise the User Dictionary (with .mmj, .jdr, .jdx, .sdx, and .bdx file extensions). The log file and error files are also stored in the same directory.

Adding the User Dictionary as a Database Resource

The User Dictionary can be added as a database resource using the same procedure as other reference files for the US Enterprise Geocoding Module. For more information, see **Adding an Enterprise Geocoding Module U.S. Database Resource** on page 10.

Notices

USPS® Notices

Pitney Bowes Inc. holds a non-exclusive license to publish and sell ZIP + 4® databases on optical and magnetic media. The following trademarks are owned by the United States Postal Service: CASS, CASS Certified, DPV, eLOT, FASTforward, First-Class Mail, Intelligent Mail, LACS^{Link}, NCOA^{Link}, PAVE, PLANET Code, Postal Service, POSTNET, Post Office, RDI, Suite^{Link} , United States Postal Service, Standard Mail, United States Post Office, USPS, ZIP Code, and ZIP + 4. This list is not exhaustive of the trademarks belonging to the Postal Service.

Pitney Bowes Inc. is a non-exclusive licensee of USPS® for NCOA^{Link}® processing.

Prices for Pitney Bowes Software's products, options, and services are not established, controlled, or approved by USPS® or United States Government. When utilizing RDI™ data to determine parcel-shipping costs, the business decision on which parcel delivery company to use is not made by the USPS® or United States Government.

Data Provider and Related Notices

Data Products contained on this media and used within Pitney Bowes Software applications are protected by various trademarks and by one or more of the following copyrights:

© Copyright United States Postal Service. All rights reserved.

© 2014 TomTom. All rights reserved. TomTom and the TomTom logo are registered trademarks of TomTom N.V.

© 2016 HERE

Fuente: INEGI (Instituto Nacional de Estadística y Geografía)

Based upon electronic data © National Land Survey Sweden.

© Copyright United States Census Bureau

© Copyright Nova Marketing Group, Inc.

Portions of this program are © Copyright 1993-2007 by Nova Marketing Group Inc. All Rights Reserved

© Copyright Second Decimal, LLC

© Copyright Canada Post Corporation

This CD-ROM contains data from a compilation in which Canada Post Corporation is the copyright owner.

© 2007 Claritas, Inc.

The Geocode Address World data set contains data licensed from the GeoNames Project (**www.geonames.org**) provided under the Creative Commons Attribution License ("Attribution

pitney bowes

3001 Summer Street
Stamford CT 06926-0700
USA

www.pitneybowes.com