



**Address Verification, Correction and ZIP+4 API User's Guide**

**The Software Company, Inc.**  
<http://SoftwareCompany.com>



**NetZipCode for .NET** allows you to quickly and easily build ZIP Code lookup, address verification plus address correction and parsing into your custom applications. The delivery address is checked for proper Postal Service format and standardized if necessary. Spelling corrections are made and the address is split into USPS standard components.

**NetZipCode for .NET** not only returns a standardized and corrected address, but also provides the following additional information about each address:

- Individually Parsed Address Components
- ZIP+4 Add-On
- Preferred/Default City Name
- County Name
- Congressional District

*An array of diagnostic flags is returned with each address letting you know exactly what strategy was used in order to achieve the best match.*

### Benefits

- **Save \$\$\$ on Postage** - verify addresses before you ship or mail
- **Catch Data-Input Errors** - virtually eliminating undeliverable addresses
- **Save Keying Time** - city/state is returned from 5-digit ZIP Code
- **Unlimited Use** - no escalating charges
- **Free Database Updates** for a full year

### Features

- Addresses are standardized to USPS recommended abbreviations
- No automatic database expiration
- Proper case conversion for more attractive data presentation
- Attractive USPS database subscription pricing
- Designed for use with all .NET-compatible programming languages

**NetZipCode for .NET** follows the US Postal Service “one component failure rule” to find a correct address match. This rule permits an address match if no more than one address component mismatch exists, where a “component” is defined as a predirectional, street name, street suffix, or a postdirectional.

Consider the following address: 100 E MAIN ST N

This address has all four components. If one of the components has to be added, changed, or deleted in order to achieve a unique match, NetZipCode will correct the address and return the ZIP+4. Under USPS address matching rules, if more than one component is incorrect, no match is allowed. If adding, changing or deleting a component results in multiple matches, NetZipCode will return a list of all matching addresses.

### Example

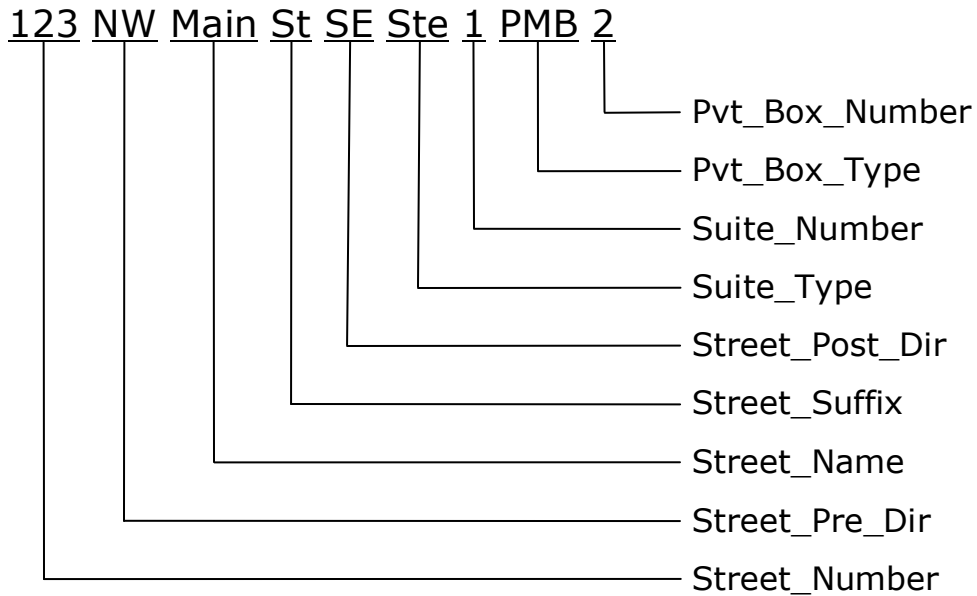
**Input Address:** 9227 2 3  
32824

**Output Address:** 9227 2<sup>nd</sup> Ave, Ste 3  
Orlando, FL 32824-8390  
County: Orange  
Preferred City: Orlando

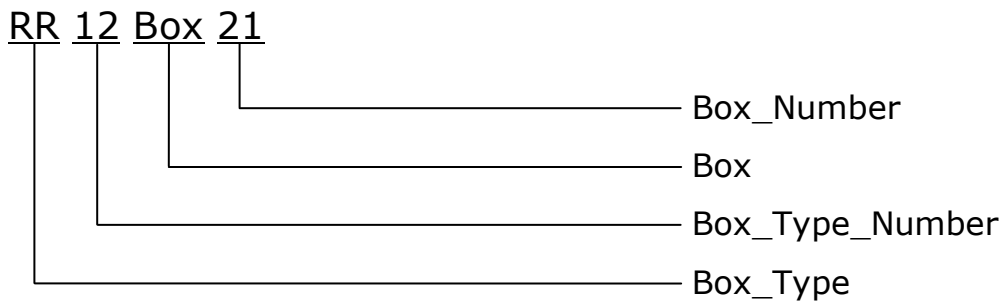
### Results:

- Street number/name verified: (deliverable address)
- Street name corrected: (2<sup>nd</sup>)
- Street suffix added: (Ave)
- Suite type added: (Ste)
- City/state added: (Orlando, FL)
- ZIP+4 added: (8390)
- County name added: (Orange)
- Preferred city added: (Orlando)
- Capitalization applied

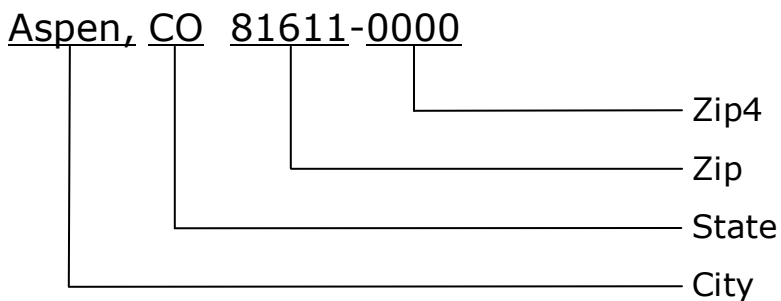
**Street Address**



**Box Address**



**City/State/Zip**



### Address\_In

**Syntax:** Address\_In = String

**Description:**

Set this property to the address string to be processed. When the “Lookup” method is invoked, the “Address\_In” and “CSZ\_In” strings are matched to the USPS National Address Database. If a match is found, the standardized and corrected address is placed into the “Address\_Out” and “CSZ\_Out” properties. In addition, each element of the “Address\_In” and “CSZ\_In” strings is placed into the appropriate address and city/state/zip component property.

---

### CSZ\_In

**Syntax:** CSZ\_In = String

**Description:**

Set this property to the city, state and/or ZIP Code string to be processed. When the “Lookup” method is invoked, the “CSZ\_In” string is matched to the USPS National Address Database. If a match is found, the standardized and corrected city/state/zip is placed into the “CSZ\_Out” property. In addition, each element of the “CSZ\_In” string is placed into the appropriate city/state/zip component property. *If this property is set to the city and state, the ZIP Code is optional. If set to the 5-digit ZIP Code, the city and state are optional.*

---

### Firm\_Name\_In (optional)

**Syntax:** Firm\_Name\_In = String

**Description:**

Set this property to the Firm or High-rise name string to be processed. This will produce the highest level of match for the most precise 4-digit ZIP+4 add-on.

---

### PR\_Urb\_In (optional)

**Syntax:** PR\_Urb\_In = String

**Description:**

Set this property to the Puerto Rican urbanization string to be processed.

### Capitalization

**Syntax:** Capitalization = StringLiteral

**Description:**

Set this property to “Upper”, “Lower” or “Mixed” to indicate your capitalization preference for the output address and its components. **Default is “Upper”.**

---

### Pvt\_Box\_Types

**Syntax:** Pvt\_Box\_Types = StringArray

**Description:**

Set this property to a list of private box types you want NetZipCode to recognize. Some common types are: “PMB”, “MAC”, etc. *See “Pvt\_Box\_Type” and “Pvt\_Box\_Number” properties.*

---

### Database\_Path

**Syntax:** Database\_Path = String

**Description:**

Set this property to the full path to the USPS National Address Database file: “NetZipCode.db”. By default, this file is located in the NetZipCode installation folder. However, you can relocate this file to any other folder as long as you set this property to the full path. **Default is first the folder of the invoking application: “AppDomain.CurrentDomain.BaseDirectory” then the NetZipCode installation folder.**

---

### Static\_Key\_Name

**Syntax:** Static\_Key\_Name = String

**Description:**

Set this property to the name portion of the static key assignment or blank.

---

### Static\_Key

**Syntax:** Static\_Key = String

**Description:**

Set this property to the key portion of the static key assignment or blank.

### **Address\_Out** (read only)

**Syntax:** String = Address\_Out

**Description:**

After invoking the “Lookup” method, this property will contain the standardized and corrected address string from the “Address\_In” property including a suite number if present. If no match was found in the USPS database, this property will be blank.

---

### **Address\_Out\_Street** (read only)

**Syntax:** String = Address\_Out\_Street

**Description:**

After invoking the “Lookup” method, this property will contain the street portion of “Address\_Out”.

---

### **Address\_Out\_Suite** (read only)

**Syntax:** String = Address\_Out\_Suite

**Description:**

After invoking the “Lookup” method, this property will contain the suite portion of “Address\_Out”.

---

### **Street\_Number** (read only)

**Syntax:** String = Street\_Number

**Description:**

After invoking the “Lookup” method, this property is set to the primary address number component of “Address\_Out” commonly referred to as house number, street number or range.

---

### **Street\_Pre\_Dir** (read only)

**Syntax:** String = Street\_Pre\_Dir

**Description:**

After invoking the “Lookup” method, this property is set to the Predirectional component of “Address\_Out”. Values will be a valid directional (N, NE, S, SE, etc.) or blank.

### **Street\_Name** (read only)

**Syntax:** String = Street\_Name

**Description:**

After invoking the “Lookup” method, this property is set to the Street Name component of “Address\_Out”. Value will be alphanumeric.

---

### **Street\_Suffix** (read only)

**Syntax:** String = Street\_Suffix

**Description:**

After invoking the “Lookup” method, this property is set to the Street Suffix component of “Address\_Out”. Values will be a valid suffix (St, Ave, Rd, etc.) or blank.

---

### **Street\_Post\_Dir** (read only)

**Syntax:** String = Street\_Post\_Dir

**Description:**

After invoking the “Lookup” method, this property is set to the Postdirectional component of “Address\_Out”. Values will be a valid directional (N, NE, S, SE, etc.) or blank.

---

### **Suite\_Type** (read only)

**Syntax:** String = Suite\_Type

**Description:**

After invoking the “Lookup” method, this property is set to the Suite Type component of “Address\_Out”. Values will be only valid suite types (Apt, Suite, Unit, etc.) or blank.

---

### **Suite\_Number** (read only)

**Syntax:** String = Suite\_Number

**Description:**

After invoking the “Lookup” method, this property is set to the Suite Number component of “Address\_Out”. Values will be alphanumeric suite number or blank.



### **Box\_Type** (read only)

**Syntax:** String = Box\_Type

**Description:**

After invoking the “Lookup” method, this property is set to the Box Type component of “Address\_Out”. Values will be only valid box types (PO Box, RR, HC, etc.) or blank.

---

### **Box\_Type\_Number** (read only)

**Syntax:** String = Box\_Type\_Number

**Description:**

After invoking the “Lookup” method, this property is set to the Box Type Number component of “Address\_Out”. Values will be alphanumeric box type number or blank.

---

### **Box** (read only)

**Syntax:** String = Box

**Description:**

After invoking the “Lookup” method, this property is set to the Box component of “Address\_Out”. Value will be “Box” or blank.

---

### **Box\_Number** (read only)

**Syntax:** String = Box\_Number

**Description:**

After invoking the “Lookup” method, this property is set to the Box Number component of “Address\_Out”. Values will be alphanumeric box number or blank.

---

### **Pvt\_Box\_Type** (read only)

**Syntax:** String = Pvt\_Box\_Type

**Description:**

After invoking the “Lookup” method, this property is set to the Private Box Type component of “Address\_Out”. Values will be valid box types as specified in the “Pvt\_Box\_Types” property.

---

### **Pvt\_Box\_Number** (read only)

**Syntax:** String = Pvt\_Box\_Number

**Description:**

After invoking the “Lookup” method, this property is set to the Private Box Number component of “Address\_Out”. Values will be alphanumeric private box number or blank.

### **CSZ\_Out** (read only)

**Syntax:** String = CSZ\_Out

**Description:**

After invoking the “Lookup” method, this property will contain the standardized city/state/zip (last line) from the “CSZ\_In” property.

---

### **City** (read only)

**Syntax:** String = City

**Description:**

After invoking the “Lookup” method, this property is set to the City component of “CSZ\_Out”. See “*Preferred\_City\_Name*” property.

---

### **State** (read only)

**Syntax:** String = State

**Description:**

After invoking the “Lookup” method, this property is set to the State component of “CSZ\_Out”. Values will be only valid USPS state abbreviations (FL, AZ, CO, etc.) or blank.

---

### **Zip** (read only)

**Syntax:** String = Zip

**Description:**

After invoking the “Lookup” method, this property is set to the 5-digit ZIP Code component of “CSZ\_Out”. Values will be a 5-digit numeric ZIP Code or blank.

---

### **Zip4** (read only)

**Syntax:** String = Zip4

**Description:**

After invoking the “Lookup” method, this property is set to the 4-digit ZIP+4 add-on component of “CSZ\_Out”. Values will be a 4-digit numeric ZIP+4 add-on code (sector/segment) or blank.

### **Preferred\_City** (read only)

**Syntax:**      String = Preferred\_City

**Description:**

After invoking the “Lookup” method, this property is set to the preferred/default city name for the 5-digit “Zip” property. See “*Preferred\_City\_Name*” and “*Preferred\_State*” properties.

---

### **Preferred\_State** (read only)

**Syntax:**      String = Preferred\_State

**Description:**

After invoking the “Lookup” method, this property is set to the preferred/default state abbreviation for the 5-digit “Zip” property. Values will be valid USPS state abbreviations (FL, AZ, CO, etc.) or blank. See “*Preferred\_City\_Name*” and “*Preferred\_City*” properties.

---

### **Match\_Count** (read only)

**Syntax:**      Integer = Match\_Count

**Description:**

After invoking the “Lookup” method, this property is set to an integer count of the number of matching addresses when the “Multiple\_Match” property is set to “True”. See “*GetFirstMatch*”, “*GetNextMatch*” methods and “*Multiple\_Match*” property.

### Address\_Type (read only)

**Syntax:** String = Address\_Type

**Description:**

After invoking the “Lookup” method, this property is set to one of the following:

<b>S</b>	Street	(1 N Main St, 2 US Highway 285, etc.)
<b>P</b>	Post Office Box	(PO Box 1)
<b>R</b>	Rural Route	(RR 1 Box 2)
<b>H</b>	Highway Contract	(HC 1 Box 2)
<b>G</b>	General Delivery	(General Delivery, Gen Del, GD, etc.)
<b>M</b>	Military	(CMR 1 Box 2, etc.)
<b>N</b>	Not a valid address	(“Address_Match” property will also be set to “False”)

---

### Zip\_Type (read only)

**Syntax:** String = Zip\_Type

**Description:**

After invoking the “Lookup” method, this property is set to one of the following:

<b>P</b>	PO Box Only	(no street addresses in this ZIP Code)
<b>M</b>	Military Zip	(state codes AA, AE or AP)
<b>U</b>	Unique Zip	(assigned to a single business or organization)
<b>S</b>	Standard Zip	(any combination of street addresses and PO Boxes)
<b>N</b>	Not a valid Zip	(“CSZ_Match” property will also be set to “False”)

### **County** (read only)

**Syntax:** String = County

**Description:**

After invoking the “Lookup” method, this property is set to the name of the county in which the address resides. Values will be alphanumeric county name or blank.

---

### **County\_FIPS** (read only)

**Syntax:** String = County\_FIPS

**Description:**

After invoking the “Lookup” method, this property is set to the Federal Information Processing Standard (FIPS) code for the county in which the address resides. Values will be a three-digit numeric string or blank.

---

### **State\_FIPS** (read only)

**Syntax:** String = State\_FIPS

**Description:**

After invoking the “Lookup” method, this property is set to the Federal Information Processing Standard (FIPS) code for the state in which the address resides. Values will be a two-digit numeric string or blank.

---

### **Congressional\_District** (read only)

**Syntax:** String = Congressional\_District

**Description:**

After invoking the “Lookup” method, this property is set to the congressional district code for this address. Values will be alphanumeric congressional district or blank.

---

### **Firm\_Name\_Out** (read only)

**Syntax:** String = Firm\_Name\_Out

**Description:**

After invoking the “Lookup” method, this property is set to the firm or high-rise building name from the optional “Firm\_Name\_In” string. Values will be alphanumeric or blank.

**PR\_Urb\_Out** (read only)

**Syntax:** String = PR\_Urb\_Out

**Description:**

After invoking the “Lookup” method, this property is set to the Puerto Rican Urbanization name from optional the “PR\_Urb\_In” string. Values will be alphanumeric or blank.

**Address Flags** (read only)**Description:**

After invoking the “Lookup” method, the following Boolean (True/False) properties are returned indicating the status of the address lookup.

**Syntax:**

**Address\_Match = True** (CSZ\_Match is also set to “True”)

A match was found in the USPS National Address Database for the “Address\_In” and “CSZ\_In” properties. This condition also sets one or more of the following properties indicating any corrections that were made:

**Address\_Corrected = True**

“Street\_Number”, “Street\_Pre\_Dir”, “Street\_Name”, “Street\_Suffix” or “Street\_Post\_Dir” components were corrected in order to achieve a match.

**Suite\_Corrected = True**

“Suite\_Type” or “Suite\_Number” components were corrected in order to achieve a match.

**POB\_Corrected = True**

Post Office Box “Box\_Type” component was corrected in order to achieve a match.

**PR\_Urb\_Corrected = True**

Puerto Rican urbanization “PR\_Urb\_In” component was corrected in order to achieve a match.

**CSZ\_Match = True**

A match was found in the USPS National Address Database for the “CSZ\_In” property. This condition also sets one or more of the following properties indicating any corrections that were made:

**City\_Corrected = True**

City name component was added or corrected in order to achieve a match.

**State\_Corrected = True**

State component was added or corrected in order to achieve a match.

**Zip\_Corrected = True**

ZIP Code component was added or corrected in order to achieve a match.

**Zip4\_Corrected = True**

ZIP+4 component was added or corrected in order to achieve a match.

**Address\_Match = False**

A unique match could not be found in the USPS National Address Database for the “Address\_In” and “CSZ\_In” properties. This condition also sets one or more of the following properties indicating the reason a unique match could not be found:

**Multiple\_Match = True**

More than one address was matched in the USPS National Address Database.  
*See “GetFirstMatch”, “GetNextMatch” methods and “Match\_Count” property.*

**Street\_Number\_Unknown = True**

The “Street\_Number” was out-of-range for this street in the USPS National Address Database.

**Street\_Name\_Unknown = True**

The “Street\_Name” was not matched in the USPS National Address Database.

**CSZ\_Unknown = True**

City State ZIP Code mismatch in the USPS National Address Database.

**Insufficient\_Address\_Data = True**

There was not enough data to find a match in the USPS National Address Database.

**CSZ\_Match = False** (CSZ\_Unknown is also set to “True”)

No match was found in the USPS National Address Database for the “CSZ\_In” property.

---

**No\_USPS\_Delivery = True**

Address was verified, but USPS doesn’t deliver to this address.

**Preferred\_City\_Name = True**

The “City” property returned is the USPS preferred/default city name for the 5-digit ZIP Code. *See “Preferred\_City” and “Preferred\_State” properties.*

**Highrise = True**

Address is a high-rise apartment or office building.

**Firm = True**

Address is a firm (business). When "Firm\_Name\_In" property is set to a firm name, the firm name is matched to the USPS database and the "Firm" flag is set to “True” when a match occurs. This is the finest level of match and will, in some cases, return a different ZIP+4 add-on code as a result of the match.



### **Return\_Code** (read only)

**Syntax:** String = Return\_Code

#### **Description:**

After invoking the “Lookup” method, this property is set to blank upon successful completion. Most exceptions occur on the first invocation. The most common ones are listed below. *This property should be examined on each return from NetZipCode.*

#### **Common Return Codes:**

<b>Z01 – Z16</b>	Database initialization failed
<b>Z17</b>	“NetZipCode.db” version does not match “NetZipCode.dll” version
<b>Z35</b>	“NetZipCode.db” file not found ( <i>see “Database_Path” property</i> )
<b>L00</b>	Evaluation period expired
<b>L01</b>	Static key validation failed ( <i>see “Static_Key” property</i> )
<b>L50</b>	Evaluation license error

### Clear

**Syntax:**     NetZipCode.Clear

**Description:**

When this method is invoked, all properties are set to null with the exception of “Static\_Key”, “Static\_Key\_Name” and “Database\_Path”.

---

### Lookup

**Syntax:**     NetZipCode.Lookup

**Description:**

When this method is invoked, the “Address\_In” and “CSZ\_In” properties are compared to the USPS National Address Database. When a match is found, the address is standardized and corrected then placed into the “Address\_Out” and “CSZ\_Out” properties. In addition, each standardized element of the “Address\_Out” and “CSZ\_Out” properties is placed into the appropriate address component property. Next, the Address Flags are set to indicate the status of the address lookup as well as how complete and correct the address is. The “Return\_Code” property is also set and should be checked after each invocation of the “Lookup” method. *See “Return\_Code” property.*

---

### GetFirstMatch

**Syntax:**     NetZipCode.GetFirstMatch

**Description:**

This method can only be invoked when the “Multiple\_Match” flag is set to “True”. After invoking the “GetFirstMatch” method, the first address from the current group of matching addresses is returned and the “Address\_Match” flag is set to “True”. *See “Multiple\_Match” and “Match\_Count” properties.*

---

### GetNextMatch

**Syntax:**     NetZipCode.GetNextMatch

**Description:**

This method can only be invoked *after* the “GetFirstMatch” method has been invoked and when the “Multiple\_Match” flag is set to “True”. After invoking the “GetNextMatch” method, the next address from the current group of matching addresses is returned and the “Address\_Match” flag is set to “True”. When no more addresses remain, the “Address\_Match” property is set to “False”. *See “Multiple\_Match” and “Match\_Count” properties.*

### GetFirstCityInZip

**Syntax:** NetZipCode.GetFirstCityInZip

**Description:**

When this method is invoked, the 5-digit “CSZ\_In” property is compared to the USPS National Address Database. When a match is found, the first city name (alphabetically) in the list of USPS-approved mailing names for this ZIP Code is returned in the “City” property and the “CSZ\_Match” property is set to “True”. If this is the USPS preferred/default city name for this ZIP Code, then the “Preferred\_City\_Name” property is set to “True” and the “Preferred\_City” property is set to the city name. See “Preferred\_City\_Name”, “Preferred\_City” and “Preferred\_State” properties.

---

### GetNextCityInZip

**Syntax:** NetZipCode.GetNextCityInZip

**Description:**

This method can only be invoked *after* the “GetFirstCityInZip” method has been invoked and a match was found (CSZ\_Match = True). After invoking the “GetNextCityInZip” method, the next city name (alphabetically) in the list of USPS-approved mailing names for this ZIP Code is returned in the City property and the “CSZ\_Match” property is set to “True”. If this is the USPS preferred/default city name for this ZIP Code, then the “Preferred\_City\_Name” property is set to “True” and the “Preferred\_City” property is set to the city name. See “Preferred\_City\_Name”, “Preferred\_City” and “Preferred\_State” properties.

### Deploying Your Applications

Be sure to include the following in your deployment package:

**NetZipCode.dll** – usually placed in the application folder or Global Assembly Cache (GAC)

**NetZipCode.db** – usually placed in the application folder\*

\* “NetZipCode.db” database file can be placed anywhere on the target machine as long as the full path to it is specified in the “Database\_Path” property.

---

In addition to the above, there is a common runtime that can be placed in the application folder or the Global Assembly Cache (GAC) of the target machine.

**Fujitsu.COBOL.dll**

---

### Evaluation License

The evaluation license is valid for a period of 30 days or up to 1,000 calls.

[Sales@SoftwareCompany.com](mailto:Sales@SoftwareCompany.com)

[Support@SoftwareCompany.com](mailto:Support@SoftwareCompany.com)