

Paper 087-2009

## PROC GEOCODE: Creating Map Locations from Your Data

Darrell Massengill and Ed Odom, SAS Institute Inc., Cary, NC

### ABSTRACT

How do you convert your address data into map locations? This is done through the process of geocoding. SAS/GRAPH® 9.2 now includes PROC GEOCODE to simplify this process for you. This presentation will briefly discuss the current capability of this procedure and show examples of both address geocoding and Web address (IP address) geolocating. A brief discussion of future directions will also be included.

### INTRODUCTION

Every business or organization has a lot of data that includes an address. The address data is useful only if it is transformed into location information that can be viewed on a map, used in distance calculations, or processed in other similar ways. To make this data useful, you must convert the address to a location having a latitude and longitude. This conversion process is called geocoding. If the address is an IP address, the process is usually called geolocating.

This paper will discuss both geocoding and geolocating using PROC GEOCODE. First, we will introduce the concepts needed to understand geocoding, and then we will discuss PROC GEOCODE's current and planned functionality. Finally, examples will show how to use PROC GEOCODE.

### GEOCODING BASICS

The geocoding process depends on having lookup data with the necessary information to make the conversion. This data is the key to geocoding. Factors such as age and granularity of the lookup data determine the geocoding results. Addresses routinely change with new construction, new roads being added, and postal codes being split and changed. The older your lookup data, the more likely it is that some address matches might be off.

Granularity is another important consideration. Does the location need to be the actual house location or is it okay at a ZIP Code level or even a city level? If you are viewing the addresses on a state or U.S. map, then ZIP Code or city is accurate enough.

To understand geocoding, it is important to first understand the lookup data. It is particularly important to understand the differences between ZIP Code data, ZIP+4 data, and street address data. IP address data is completely different from the other types of addresses, but it is important to understand this data, too.

### ZIP CODE DATA

A ZIP Code is a delivery route or a collection of roads traveled to deliver mail in an area. Sometimes, ZIP codes are assigned to a single building or to a post office. ZIP Code boundaries are not created by the U.S. Postal Service (USPS). ZIP Codes are not polygonal areas in the way that a county, state, or country is. Creating polygons by simply wrapping the delivery route would leave gaps between the polygons because there are large, undeveloped areas.

The area covered in a ZIP Code varies by how densely populated it is. A rural ZIP Code will be larger than one in a large city. Generally, ZIP Code address data specifies a centroid location for the ZIP Code area. The centroid is the geographic center of the area, but because the geographic area isn't real, then that location can vary between different data providers. Appendix 1 contains many frequently asked questions about ZIP Codes.

The standard ZIP Code in the United States is five digits. Figure 1 illustrates a ZIP Code area and its centroid location. All addresses in this ZIP Code would be assigned the same centroid X and Y values if you are geocoding with ZIP Code data.

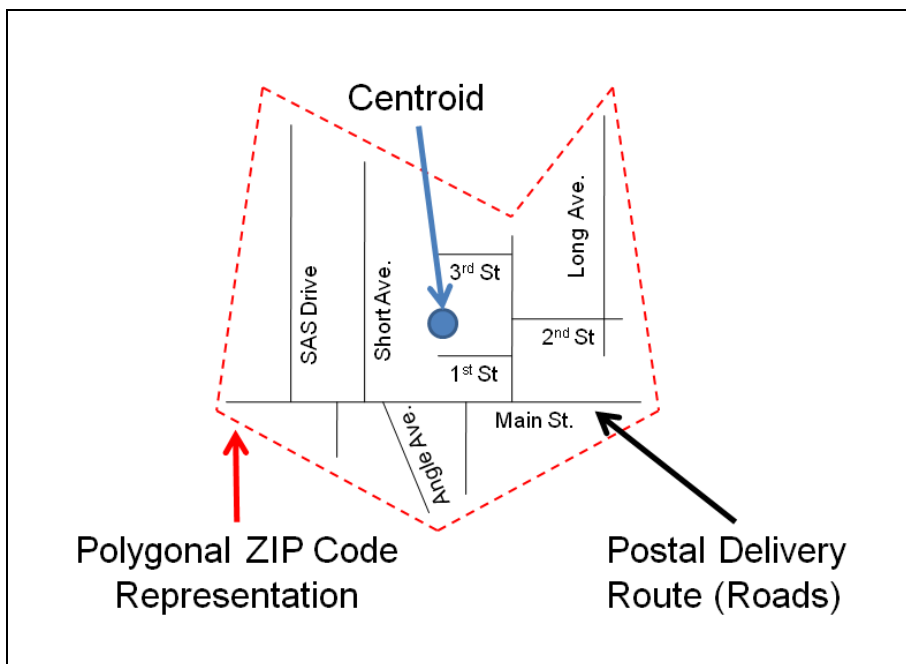


Figure 1. ZIP Code Area

A ZIP Code is further divided by ZIP+4 areas. Four additional digits are appended to the ZIP Code to specify these additional subdivisions. A ZIP+4 will likely represent a single street or a part of a street. In a high-density city area, it might represent one side of a street on a single block, or even one floor of a large building. Figure 2 illustrates the relationship between a ZIP Code and a ZIP+4. The centroid is the midpoint of those addresses in the ZIP+4 area.

All addresses in this ZIP+4 area would get the same centroid if geocoded with ZIP+4 data. However, there would be multiple locations within the overall ZIP Code area for each ZIP+4.

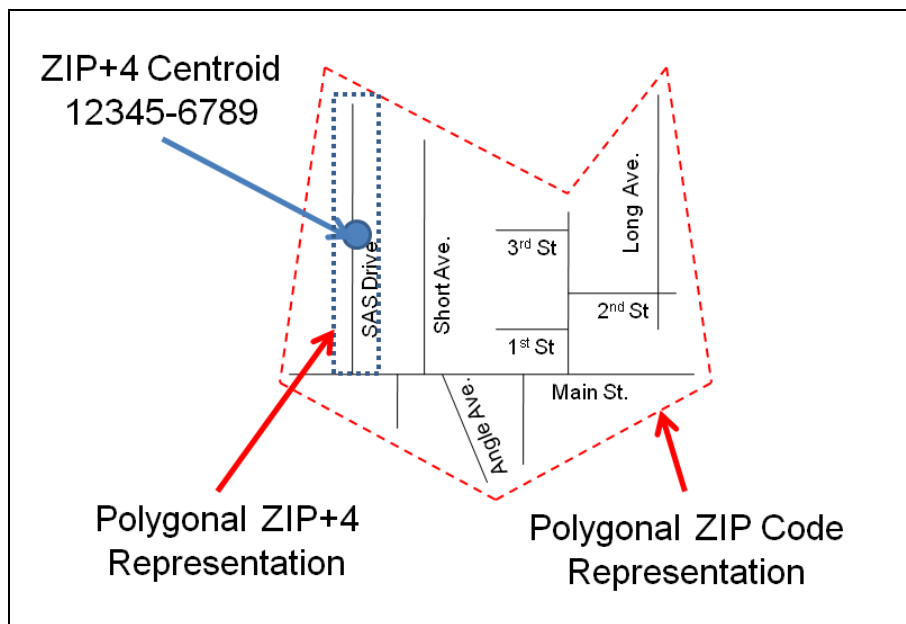


Figure 2. ZIP+4 Area

ZIP Code data will get you to the general location of the address, but not to the actual house location. ZIP+4 data will probably get you to the correct street in the address, but not to the actual house location. To geocode to the specific house location, you need street-level data.

### STREET ADDRESS DATA

Street-level address data contains information about the ZIP Code, state, city, and street. This information enables you to process the entire address and make your map location more accurate. This data does not contain the location for every house number or address. Generally, this data only approximates the position of a particular address on a street by assuming that house numbers are an equal distance apart, which might not be true. Figure 3 illustrates street-level addressing.

Anyone who has used a GPS or looked up directions on the Web knows that the location found might not be quite right. The same is true when using any geocoding mechanism.

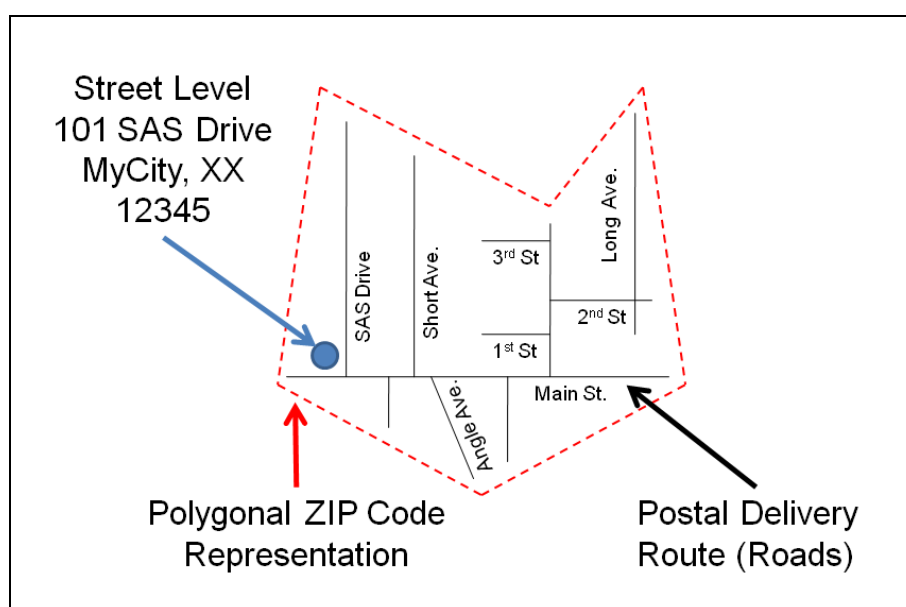


Figure 3. Street-Level location

### IP GEOLOCATING DATA

IP data is a form of range data and was not designed to be geographic, like street addresses. IP addresses are very different from ZIP Code and street addresses. Generally, these are collected from visitors to Web sites and indicate the connection the visitor used. IP address lookup data contains information that matches ranges of IP addresses to particular geographic locations. The location found will not be at the street or even ZIP Code level, but might indicate the city, state, or country where the IP address is registered.

### CHOOSING YOUR DATA

The type of geocoding you want to do will determine the type of lookup data needed. What things are important to you? How precise does the location need to be? Do you need the street-level address, or will the ZIP Code or city location be enough? How up-to-date does the data need to be? How much are you willing to pay for the data? The more up-to-date, accurate, and fine-grained the data, the more it costs to purchase and maintain it. Also, higher-resolution data requires more disk storage space and takes longer to run the geocoding process against it. There are free sources for some types of data, but these are not updated as frequently as the data you purchase.

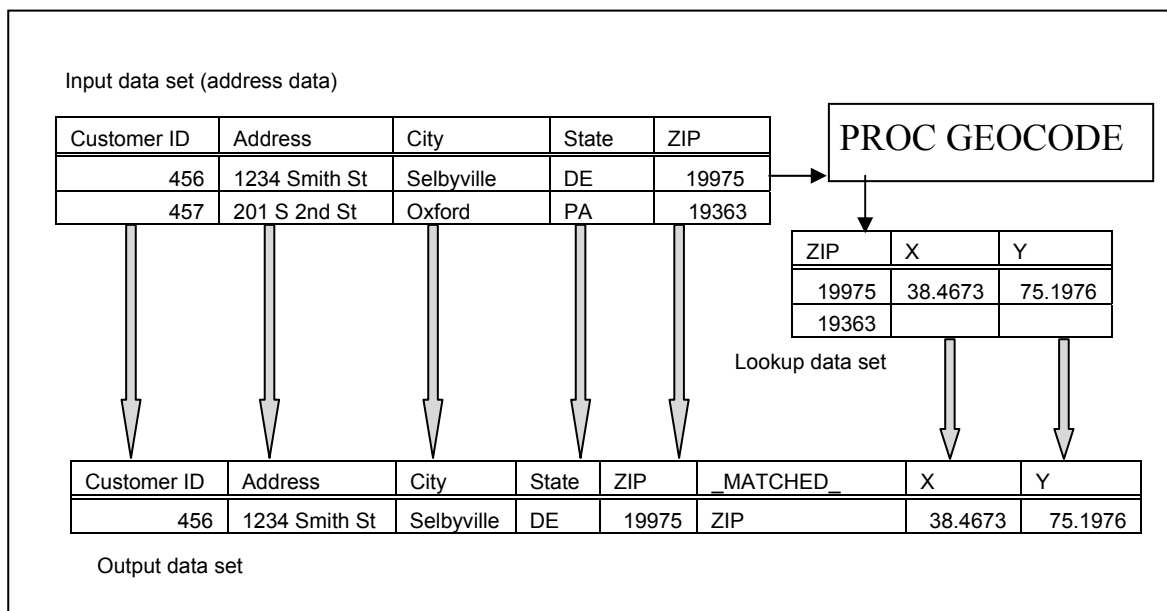
## PROC GEOCODE

The GEOCODE procedure converts address data to geographic coordinates (latitude and longitude values). These geographic coordinates can then be used on a map to calculate distances or to perform spatial analysis. Appendix 2 contains more information about what can be done with the geocoded coordinates. In addition, the procedure enables you to add attribute values to your data if they are in the lookup data. Examples would be adding census blocks or area codes to an address.

The GEOCODE procedure requires two SAS data sets:

- The input data set that you want to geocode. This data set will contain variables related to the address such as street address, city, state, and ZIP Code.
- A lookup data set containing the data to transform your address data into geographic locations. By default, SASHELP.ZIPCODE is used for ZIP Code or city geocoding. Range data (for example, IP addresses) uses two data sets.

The simplest example of how these data sets are used is with ZIP Code geocoding. Figure 4 shows that the variables from the input data are carried forward into the output data set. The ZIP variable is looked up in the lookup data set and the X and Y values (longitude and latitude) are added to the output data set. These are the geographic coordinates of that ZIP Code centroid. In addition, if the lookup data set contains other attributes such as county names or census blocks, you can specify that these additional values be moved to the output data set.



**Figure 4. ZIP Code Lookup**

In reality, geocoding is more complicated than this. By default, the procedure will give you the next larger area if the ZIP Code isn't found. If a standard five-digit ZIP Code isn't found, then it will attempt to find the city area location. If a ZIP+4 isn't found, it will move to the ZIP Code, and then to the city. If you are interested in the ZIP Code location only, you can turn off this behavior. The `_MATCHED_` variable indicates the type of successful match that was found. The value in this example means the ZIP Code matched. Other possible values are listed in the documentation.

Currently, the GEOCODE procedure supports five methods of geocoding. These include ZIP Code centroid, ZIP+4 center, city center, range geocoding (commonly used as IP address geolocating), and custom geocoding. The preceding descriptions of ZIP Code and ZIP+4 data explain the type of data returned for these methods. City geocoding is similar to the others, but a city location is found by averaging the ZIP Code centroid data for the city and state to find the mean location. Range geocoding is used when the address value in one value is in a range of values. This is commonly used for finding the locations of IP addresses, but any type of range lookup could be done. Custom geocoding enables you to provide your own data and lookup key.

Additional geocoding methods are currently being researched and developed. Street-level geocoding is currently in development for a future SAS release.

Lookup data is the key to geocoding. SAS does not provide geocoding data for all of the geocoding methods. Some data must be found on external sites or purchased. Where possible, there is data available on the SAS Maps Online Web site. The lookup data is discussed in more detail below.

## PROCEDURE SYNTAX

The syntax for the GEOCODE procedure is very flexible and enables you to specify different lookup data sets and use different variable names.

The basic syntax is like other SAS procedures:

```
PROC GEOCODE <options(s)>;
```

The data set options are the following:

DATA is the address data set. This contains the address data to be geocoded. The options in the following table that begin with ADDRESS are used to specify the names of the variables in this data set. If the variable names match the DEFAULT column in the table, the option is not required.

OUT is the output data set. This is the data set containing the geocoded results.

LOOKUP is the lookup data set. This specifies the data set containing the lookup data to convert the address data. The options in the table beginning with LOOKUP are used to specify the names of the variables in this data set. If the variable names match the DEFAULT column, then the option is not required.

There are currently five geocoding methods supported in this procedure. These methods control the type of geocoding performed and the type of lookup data required.

- ZIP – ZIP Code centroid geocoding. If no match is found for the five-digit ZIP Code, then it uses the CITY method unless the NOCITY option is specified. The default lookup data set is SASHELP.ZIPCODE.
- PLUS4 – ZIP+4 geocoding. If no match is found for the nine-digit ZIP+4, then it uses the ZIP method unless the NOZIP option is specified.
- CITY – City and state geocoding.
- RANGE – Matches are made to lookup data containing a range of values. IP addresses are a common use of this type of geocoding. This method requires specifying a second data set with the RANGEDATA= option, in addition to the LOOKUP= option.
- CUSTOM – This method enables you to use custom lookup data. Examples are internal sales territories, Metropolitan Statistical Areas (MSA), and telephone area codes.

Different options are used with the different geocoding methods. The following table indicates which methods use which option. The DEFAULT column indicates the expected variable name in the data set. If the variable matches that name, the option is not required.

Option	Argument	ZIP	PLUS4	CITY	RANGE	CUSTOM	DEFAULT
ADDRESSCITYVAR=	char-variable	Y	Y	Y			CITY
ADDRESSPLUS4VAR=	variable		Y				PLUS4
ADDRESSSTATEVAR=	char-variable	Y	Y	Y			STATE
ADDRESSVAR=	variable				Y	Y	
ADDRESSZIPVAR=	variable	Y	Y				ZIP
ATTRIBUTEVAR=	variable-list	Y	Y	Y	Y	Y	
BEGINRANGEVAR=	numeric-variable				Y		
ENDRANGEVAR=	numeric-variable				Y		
LOOKUPCITYVAR=	char-variable	Y	Y	Y			CITY
LOOKUPKEYVAR=	variable				Y		
LOOKUPPLUS4VAR=	variable		Y				PLUS4
LOOKUPSTATEVAR=	char-variable	Y	Y	Y			STATECODE
LOOKUPVAR=	variable					Y	
LOOKUPXVAR=	numeric-variable	Y	Y	Y	Y	Y	X
LOOKUPYVAR=	numeric-variable	Y	Y	Y	Y	Y	Y
LOOKUPZIPVAR=	variable	Y	Y				ZIP
NOCITY		Y	Y				
NOSTIMER		Y	Y	Y	Y	Y	
NOZIP			Y				
RANGEDATA=	data-set				Y		
RANGEDECIMAL					Y		
RANGEKEYVAR=	variable				Y		

The complete syntax is specified in SAS/GRAPH documentation for the GEOCODE procedure. This can be found in the SAS/GRAPH Reference at <http://support.sas.com/documentation/onlinedoc/graph/index.html>.

## LOOKUP DATA AND TOOLS

The GEOCODE procedure does not ship with all the lookup data you might require. In some cases, you must download the data or possibly purchase it. Macro code is shipped to import some third-party data. These macros can be modified to import data from additional sources. Each type of lookup data is discussed below.

- **ZIP Code**

- Uses SASHELP.ZIPCODE, which ships with SAS and supports U.S. ZIP Codes.
- Quarterly updates are available from the SAS Maps Online Web site at <http://support.sas.com/rnd/datavisualization/maponline/html/misc.html>.
- The LOOKUP= option can be used to point to alternate data sources.
- Monthly updates can be purchased directly and imported from [www.zipcodedownload.com](http://www.zipcodedownload.com).
- Non-U.S. postal codes and other sources of data can be imported and used.

- **ZIP+4**

- Data values derived from the TIGER ZIP+4 files set are available from the SAS Maps Online Web site at <http://support.sas.com/rnd/datavisualization/maponline/html/geocode.html>. Instructions are in the ReadMe.txt file. The data was developed using TIGER 2006, Second Edition. That is the

latest TIGER release that will include ZIP+4 values until the 2010 Census, or possibly later according to Census documentation.

- You can purchase Melissa Data Geo\*Data ZIP+4 product ([www.melissadata.com](http://www.melissadata.com)) at <http://w2.melissadata.com/geocoder/geodata.htm>.
  - SAS 9.2 includes an autocall macro (GCDMEL9.SAS), which imports Geo\*Data files into SAS data sets for use with PROC GEOCODE. You can modify this program to import other sources of data.
- **CITY**
    - This mechanism uses the same data as the ZIP Code method.
  - **RANGE (IP Address)**
    - One source of IP address lookup data is available from MaxMind ([www.maxmind.com](http://www.maxmind.com)).
    - GeoLite City and GeoLite Country are free products available from <http://www.maxmind.com/app/geolitecity> and <http://www.maxmind.com/app/geolitecountry>.
    - GeoIP City and GeoIP Country are the more accurate fee-based products at <http://www.maxmind.com/app/city> and <http://www.maxmind.com/app/country>.
    - SAS 9.2 includes an autocall macro (MAXMIND.SAS), which imports the CSV MaxMind files (binary format not supported). You can modify this program to import other sources of data.
  - **Custom**
    - Any data can be used as a custom lookup data set. The only requirement is that you have at least three variables that represent the X and Y values and a key variable to look up. The following example for the CUSTOM geocoding method provides code where a custom data set is created.

## EXAMPLES

The complete examples are available for download from <http://support.sas.com/rnd/papers>. Search for the title of this paper. In some cases, the lookup data must be purchased or downloaded and processed before using the examples. These examples demonstrate each of the current geocoding methods available in SAS 9.2.

## ZIP CODE GEOCODING

This example uses the ZIP geocoding method. Sample address data to be geocoded is provided in the example, but you will usually provide this. The lookup data is SASHELP.ZIPCODE. The example will attempt to provide the location of the ZIP Code centroid for each address. If the ZIP Code cannot be found, then it will return the location of the city. In addition, the variables MSA and AREACODE will be added to the output data for each match.

Here is a sample of the address data for ZIP Code, ZIP+4, and city geocoding.

Customer	Address	ZIP	Plus4	City	State	Revenue
J. Cheever Loophole	136 E. Water St	19901	3630	Dover	DE	985.33
Cuthbert J. Twillie	760 Moose Lodge Road	19934	2220	Camden	DE	2533.25
Kaspar Gutman	4701 Limestone Road	19808	1927	Wilmington		33.44

The procedure syntax is:

```
proc geocode
  zip                /* Geocoding method      */
  data=work.customers /* Input address data    */
  out=work.geocoded  /* Output data set       */
  attribute_var=(msa areacode); /* Added output variables */
run;
quit;
```

See the file ZIP\_GEOCODE.SAS for the complete example.

The WORK.GEOCODED data set will contain all of the variables in the address data set, plus a variable for X, Y, MSA, and AREACODE. In addition, a `_MATCHED_` variable is created to give you the type of match done. Most of these matches should be ZIP, but some might be None, City, or City mean. These indicate that the ZIP Code specified was not found, or that a lower-level location was found instead.

Figure 5 shows a map that displays the geocoded points from this example. The examples contain a program (MAPDEL.SAS) that can be used to display the points on the state of Delaware.

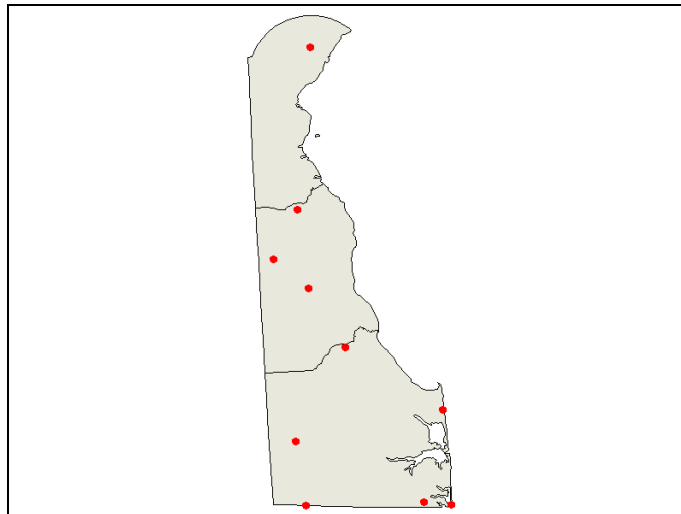


Figure 5. ZIP Code Locations Found for Delaware

## ZIP+4 GEOCODING

This example uses the ZIP+4 (PLUS4) geocoding method. Sample address data to be geocoded is provided in the example, but you will usually provide this. The address data for this example is very similar to the data for ZIP Code geocoding in the previous example. SAS does not ship lookup data for this example. You must either download the data from the SAS Maps Online Web site, or purchase your own data. The example will attempt to provide the location of the ZIP+4 center for each address. If the ZIP+4 is not found, then it will attempt to process the ZIP Code centroid. If the ZIP Code cannot be found, then it will return the location of the city by default, but this example will not process the city because of the NOCITY option. In addition, the attribute variables TRACT, BLOCK, FIPS, FENAME, and FETYPE will be added to the output data for each match.

The procedure syntax is:

```
proc geocode
  plus4                /* Geocoding method */
  lookup=lookup.zip4   /* Lookup data set */
  data=work.customers  /* Input address data */
  out=work.geocoded    /* Output data set */
  attributevar=(tract block fips fename fetype) /* Added output variables */
  nocity;              /* Only do ZIP4 or ZIP, not CITY */
run;
quit;
```

## CITY GEOCODING

This example uses the CITY geocoding method. Sample address data to be geocoded is provided in the example, but you will usually provide this. The address data for this example is very similar to the data for ZIP Code geocoding. This example uses the SASHELP.ZIPCODE file to create mean city center, based on the ZIP Codes in that city. In addition, the variables MSA and AREACODE will be added to the output data for each match.

The procedure syntax is:

```
proc geocode
  city                /* city method */
  data=work.customers /* address data to geocode*/
  out=work.geocoded  /* output data set */
  lookup=sashelp.zipcode /* lookup data set to use */
```



```

        attribute_var=(msa areacode); /* Added attribute variables*/
run;
quit;

```

## CUSTOM GEOCODING

Custom geocoding is a flexible lookup that enables you to apply your own type of lookup data to your address data. In this example, the address data is a list of customers that have only an area code from their phone number. The lookup data is created by extracting the necessary area code data from the SASHELP.ZIPCODE data set. Because this is a custom file, you must specify the name of the lookup data set, the name of the address variable, and the name of the lookup variable.

The following code creates the custom area code lookup file:

```

proc sort data=sashelp.zipcode out=temp;
  by areacode;
run;
proc means data=temp noprint;
  by areacode;
  label x='Mean Longitude of Area Code'
        y='Mean Latitude of Area Code'
        areacode='Area Code';
  var x y;
  output mean= out=work.areacodes (keep=x y areacode label='Area Code Centers');
run;

```

The procedure syntax is:

```

proc geocode
  custom                /* Custom Geocoding method      */
  data                  = customers          /* Input data set to geocode    */
  out                   = geocoded          /* Geocoded output data set     */
  lookup                = work.areacodes    /* Lookup data set              */
  lookupvar             = areacode          /* Geocode var in lookup data   */
  addressvar            = areacode;         /* Customer var to geocode      */
run;
quit;

```

## IP GEOCODING

IP geocoding uses the RANGE geocoding method and is the most common use of this method. IP addresses are generally referred to as being geolocated, rather than geocoded. Usually, a range of IP addresses belongs to a company or an Internet provider, so the lookup information is in ranges. There are no default variable names with this method, so all data sets and variable names must be specified. There are two files used for lookup. The range data set provides the ranges. The lookup data provides the latitude and longitude information. A KEY variable links the two data sets. Internally, the proper range is found, and then the key value is used to access the lookup data set to find the latitude and longitude for that key.

The procedure syntax is:

```

proc geocode
  range                /* Geocoding method          */
  out=work.IPlocations /* Geolocated output data   */

  data=weblog          /* IP address input data     */
  addressvar=IPaddress /* Variable in input data   */

  rangedata=lookup.cityblocks /* Range data                */
  beginrangevar=startipnum /* Variable in rangedata - begin range */
  endrangevar=endipnum /* Variable in rangedata - end range */
  rangekeyvar=locid /* Key variable that ties to lookup data*/

  lookup=lookup.citylocation /* Lookup data                */
  lookupkeyvar=locid /* Key variable linking to range data */
  lookupyvar=latitude /* X variable name           */
  lookupxvar=longitude /* Y variable name           */
  attributevar=(city,country); /* Additional vars wanted */
run;
quit;

```

## CONCLUSION

PROC GEOCODE converts your address data to map locations. The procedure is both simple to use and extremely flexible, enabling you to provide your own lookup data. If you need the most current data, non-U.S. postal codes, or any specialized data, you can purchase this data and easily use it with the procedure.

## REFERENCES

SAS/GRAPH 9.2 Documentation: <http://support.sas.com/documentation/onlinedoc/graph/index.html>

## RESOURCES

“PROC GEOCODE: Creating Map Locations from Your Data.” SAS Global Forum 2009 paper and example source code download. SAS Institute Inc. <http://support.sas.com/rnd/papers/>.

“Tips and Tricks IV: More SAS/Graph Map Secrets.” SAS Global Forum 2009 paper and example source code download. SAS Institute Inc. <http://support.sas.com/rnd/papers/>.

“SAS Mapping: Technologies, Techniques, Tips and Tricks.” SUGI 28 SAS Presents handout and example source code download. SAS Institute Inc. <http://support.sas.com/rnd/papers/>.

“Tips and Tricks II: Getting the most from your SAS/GRAPH maps.” SUGI 29 SAS Presents handout and example source code download. SAS Institute Inc. <http://support.sas.com/rnd/papers/>.

“Tips and Tricks III: More Unique SAS/GRAPH Maps.” SUGI 30 SAS Presents handout and example source code download. SAS Institute Inc. <http://support.sas.com/rnd/papers/>.

SAS Global Forum Proceedings: <http://support.sas.com/events/sasglobalforum/previous/index.html>

SAS Customer Support: <http://support.sas.com/>.

SAS Online Documentation: <http://support.sas.com/documentation/index.html>

SAS Maps Online Web Site: <http://support.sas.com/maponline>

## MAPS ONCONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the authors at:

Darrell Massengill  
SAS Institute Inc.  
SAS Campus Drive  
Cary, NC 27513  
[Darrell.Massengill@sas.com](mailto:Darrell.Massengill@sas.com)

Ed Odom  
SAS Institute Inc.  
SAS Campus Drive  
Cary, NC 27513  
[Ed.Odom@sas.com](mailto:Ed.Odom@sas.com)

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.

## APPENDIX 1

### ZIP CODE FREQUENTLY ASKED QUESTIONS

**What is a ZIP Code?**

It is a five-digit number identifying a U.S. Postal Service delivery area. ZIP Codes were developed in the 1960s to speed mail sorting.

**What does a ZIP Code represent?**

Most ZIP Codes are usually a group of linear delivery routes (streets). However, a ZIP Code can also be a single location, such as a post office, a large commercial building, a university, a military base, or even a navy ship.

**Do ZIP Codes change?**

Yes. ZIP Codes change as the numbers of customers on mail carrier routes increase with population changes.

**Do ZIP Codes cross county or state lines or Census blocks or tracts?**

Yes.

**Does SAS provide ZIP Code data?**

Yes. The SASHELP.ZIPCODE data set lists each ZIP with its centroid (latitude and longitude), city, state, county, telephone area code, Metropolitan Statistical Area (MSA), and time zone.

**How often is SASHELP.ZIPCODE updated?**

The most current version is included in each SAS release. Normally, only one or two new ZIP Codes are added per update.

**Can I get more frequent updates of SASHELP.ZIPCODE?**

Yes. The data set is updated quarterly and available from the SAS Maps Online Web site at <http://support.sas.com/rnd/datavisualization/maponline/html/misc.html>.

**Can I get older ZIP Code data for a historical analysis?**

Yes. You can download previous versions of SASHELP.ZIPCODE from the SAS Maps Online Web site.

**Does SAS provide ZIP Code boundary maps?**

No. ZIP Codes represent general delivery areas, not polygons. Creating seamless polygons to enclose these is very time-consuming and labor-intensive. Because ZIP Codes also change, they have to be redone on a regular basis.

**Does the Postal Service provide ZIP Code boundary maps?**

No.

**Who does provide ZIP Code boundary maps?**

Various third-party spatial data vendors do. They can easily be located with an Internet search. Because the Postal Service does not provide these maps, there are no official ZIP Code boundaries. Because each data vendor generates these using its own methods, there will likely be some differences among them.

**Why are ZIP Code boundary maps so expensive?**

See the previous answer explaining why SAS does not provide these maps.

**What is ZIP+4?**

ZIP+4 is an optional four-digit add-on to the standard ZIP Code. It basically subdivides a larger ZIP Code delivery area into smaller carrier routes. It is used primarily by businesses or large commercial mailers to pre-sort mail for a bulk discount.

**Does SAS provide ZIP+4 data?**

Yes. The ZIP4 data set was generated from Census Bureau 2006 Second Edition TIGER files. It includes ZIP Code, ZIP+4, latitude, longitude, street name, Census tract and block, city, state/county FIPS code, and Census Feature Class Code (CFCC) values. Because this data set is too large to ship with SAS, you can download it from the SAS Maps Online Web site. It is in SAS transport format and includes a SAS CIMPORT program to convert it into a

SAS data set for your platform. This data set can be used with the PLUS4 option in PROC GEOCODE for ZIP+4 geocoding.

**Who else provides ZIP+4 data?**

Various third-party data vendors do. For example, the Melissa Data GEO\*Data product contains ZIP+4 centroids (<http://www.melissadata.com/>). SAS provides the GCDMEL9.sas autocall macro to import these files into SAS data sets for use by PROC GEOCODE. This macro program can be modified to import other vendor's products as well.

**How often is the SAS ZIP4 data set updated?**

The last TIGER data release from the Census Bureau that contained ZIP+4 values was the 2006 Second Edition. For the most recent 2008 TIGER release, the Census Bureau changed to the shapefile (.shp) format. That release did not include ZIP+4 values. The Census Bureau stated that ZIP+4 values might reappear sometime after the 2010 Census. We will update the ZIP4 data set whenever the Census Bureau TIGER file updates include those values.

**Does SAS provide ZIP+4 boundary maps?**

No. See the previous answer explaining why SAS does not provide ZIP Code maps.

**What are ZCTAs?**

ZIP Code Tabulation Areas (ZCTA) are generalized ZIP Code maps created by the Census Bureau. It says that a ZCTA is "a statistical geographic entity that approximates the delivery area for a U.S. Postal Service five-digit or three-digit ZIP Code." ZCTAs are closed polygons. There is not a ZCTA for each ZIP Code.

**Where can I get ZCTA data?**

It is available by state from the U.S. Census Bureau at <http://www.census.gov/geo/www/cob/z52000.html>. PROC MAPIMPORT will import the shapefile (.shp) format into SAS/GRAPH map data sets. SAS/GIS will import the ARC/INFO Export (.e00) format into SAS/GIS spatial data sets.

**How often are ZCTAs updated?**

ZCTAs were originally released using 2000 Census data and have not been updated since. According to the Census Bureau, the 2010 Census will include new ZCTA files.

**Where can I get additional ZCTA information?**

The Census Bureau maintains a ZCTA FAQ at <http://www.census.gov/geo/ZCTA/zctafaq.html>.

**Does SAS provide non-U.S. Postal Codes?**

No. Various third-party spatial data vendors do. They can be located with an Internet search.

## APPENDIX 2

### SPATIAL CAPABILITY SUMMARY

#### DISPLAY ON MAP

Geocoding assigns latitude and longitude values to your addresses. Those geocoded locations can then be displayed on a map:

- Annotate them onto a SAS/GRAPH map.
- Import them as a point layer into a SAS/GIS map.
- Import them into a third-party mapping system, for example, ESRI ArcView or Google Maps.

#### BASE SAS FUNCTIONS

SAS address- or coordinate-related functions:

- FIPNAME - Convert two-digit FIPS code to uppercase state name
- FIPNAMEL - Convert two-digit FIPS code to mixed case state name
- FIPSTATE - Convert two-digit FIPS code to two-character state postal code
- GEODIST - Calculate geodetic distance between latitude and longitude coordinates
- STFIPS - Convert state postal code to FIPS state code
- STNAME - Convert state postal code to uppercase state name
- STNAMEL - Convert state postal code to mixed case state name
- ZIPCITY - Return city name and two-character postal code for a ZIP Code
- ZIPCITYDISTANCE - Calculate geodetic distance between ZIP Codes
- ZIPFIPS - Convert ZIP Code to two-digit FIPS code
- ZIPNAME - Convert ZIP Code to uppercase state name
- ZIPNAMEL - Convert ZIP Code to mixed case state name
- ZIPSTATE - Convert ZIP Code to two-character state postal code

#### SPATIAL ANALYSIS

Spatial analysis involves determining the effect of position on response data. Geocoded X and Y locations can be used as the basis for spatial analyses. Several SAS procedures and products perform spatial analysis of attribute values associated with X and Y locations and compute predicted values at other positions:

- PROC VARIOGRAM
- PROC KRIGE2D
- JMP