CHAPTER
6

Batch Geocoding

Overview

Geocoding is the process of adding location information to an existing data set that contains address data. *Location information* is the X and Y coordinate data for the street addresses on the map. The geocoding facility in SAS/GIS software attempts to match each address in a SAS address data set to a location on the map. If a match is found, the X and Y coordinates of the address are added to the address data set. Other spatial information about the matched location can also be added to the address data set.

SAS/GIS software provides an interactive interface for geocoding in the GIS Geocoding window. The window is convenient for geocoding individual address data sets. However, if you have a large number of data sets or a data set with a large number of observations that you want to geocode, you may find the *batch geocoding facility* to be more convenient. Beginning with Release 6.12 of the SAS System, the batch geocoding facility in SAS/GIS allows data to be geocoded without invoking SAS/GIS, without user intervention, and with improved performance. For example, you can set up a program to run overnight to geocode address data sets without user interaction.

Addresses in Spatial Data

To use geocoding in SAS/GIS, you must set up your spatial database to contain address information. SAS/GIS uses CLASS values for composites to identify address information on the spatial database. One of the composites must be defined as **`CLASS=CITY`** to point to the city name, and one of the composites must be defined as **`CLASS=ADDRESS`** to point to the location portion of the address. Composites that are defined with other CLASS values improve accuracy. You can use the SPATIAL CONTENTS statement in PROC GIS to view the composites that are defined for your spatial database.

The following composite CLASS values identify elements of the address information:

NAME
Identifies the name component of the address feature, such as **Main** in the address **101 N Main Ave**.

TYPE
Identifies the type component of the address feature, such as **Ave** in the address, **101 N Main Ave**.

ADDRESS
Identifies the specific address of the feature, such as **101** in the address, **101 N Main Ave**.

A chain has four values to define the address range for each side:

FROMLEFT	Beginning address on the left side
TOLEFT	Ending address on the left side
FROMRIGHT	Beginning address on the right side
TORIGHT	Ending address on the right side.

DIRECTION_PREFIX
Identifies the directional prefix component of the address feature, such as **N** in the address **101 N Main Ave**.

DIRECTION_SUFFIX
Identifies the directional suffix component of the address feature, such as **W** in the address **1141 First St W**.

CITY | PLACE
Identifies the value as a city name.

STATE
Identifies the value as a state name.

ZIP
Identifies the value as a ZIP code value.

PLUS4
Identifies the value as a ZIP+4 extended postal code value.

To perform geocoding, you must set up your spatial database to contain composites for CITY and ADDRESS at the minimum. Composites that are defined with additional CLASS values will help to improve the accuracy of the geocoding.

You can use the SPATIAL CONTENTS statement of PROC GIS to determine whether your spatial database contains the minimum composites that are necessary to perform geocoding. Submit the following statements in the SAS Program Editor for the spatial entry that you wish to geocode against:

```
proc gis catalog=libref.catalog;
spatial contents spatial-entry;
run;
quit;
```

The output that is produced by the SPATIAL CONTENTS statement will include a list of all of the composites that are defined for the specified spatial entry. If the spatial database includes address information, this list will include some or all of the composites that are defined with the required CLASS values for address information.

Using Batch Geocoding

Using the batch geocoding facility is a two-step process:

1 Use the %GCBATCH macro to assign values to the macro variables that control the geocoding process. See "%GCBATCH Macro Syntax" on page 74 for more information.

2 Call the SCL program to perform batch geocoding, SASHELP.GIS.GEOCODEB.SCL. In a SAS program, you can use the DM statement to issue an AF command to execute the SCL, as follows:

```
dm 'af c=sashelp.gis.geocodeb.scl; run;';
```

Note: If you are invoking SCL from your own frame application, you must use CALL display, for example, **call display('sashelp.gis.geocodeb.scl')**, instead of the DM command. △

How Batch Geocoding Works

To achieve the most accurate geocoding, ensure that the address data set to be geocoded contains name, address, city, state, ZIP code, and ZIP+4 variables. At least the address and city variables are required.

The geocoding facility first takes the chains, nodes, and details data sets and creates new data sets for the sorted and summarized versions in the SAS data library that was specified with the GLIB macro variable. Names for the geocoding data sets are generated from the specified chains data set name. For example, if your chains data set is GMAPS.USAC and you specify GLIB=GEOLIB in the %GCBATCH macro, then the geocoding facility creates the following data sets:

GEOLIB.USAS (Sorted chains)

GEOLIB.USAM (Summarized chains)

GEOLIB.USAP (Detail points and nodes)

The geocoding facility uses these data sets to match the addresses in the address data set. As it is processing the address data set, the geocoding facility provides a progress indicator. For every 10 percent of the addresses that are geocoded, a message is written to the SAS log.

When a match is found, the coordinates of the address location are added to the address data set, along with any other composite values for the specified address. For example, if the spatial data have a composite named TRACT that contains census tract numbers, you can use the geocoding process to add a TRACT variable to your address data set. The resulting geocoded address data set can be used as attribute data for the map, or it can be imported to add point data to the map by using a generic import.

If an address cannot be matched to the spatial data but the address includes a ZIP code, the X and Y coordinates of the center of the ZIP code centroid for that zone are returned instead of the exact coordinates of the address.

For matching purposes, the geocoding process converts the address components to uppercase and attempts to convert direction and street type values to standard forms. The standardized versions of the address components are also added to the address data set. The M_ADDR, M_CITY, M_STATE, M_ZIP, M_ZIP4 variables that are added to the address data set reflect the address values that were actually matched during the geocoding process.

The geocoding process also adds _SCORE_ and _STATUS_ variables to the address data set. The _SCORE_ variable's value indicates the reliability of the address match. The score is calculated by adding points for matching various components of the address, as follows:

Matching Characteristic	Points
Street number	40
Street name	20
Street type	5
Street direction	5
City	5
State	5
ZIP code	15 (or 5 if only the first three digits match)
ZIP+4 code	5

A score of 100 indicates that a match was found for all of the components of the address. A score of 100 is possible only if the address in the data set includes values for all components and the spatial database has composites for all components. For example, if the address in the data set does not have a ZIP+4 value or if the spatial database does not have a composite of class PLUS4, then the highest possible score is 95.

The _STATUS_ variable can contain values such as the following:

□ Found

□ City not found

□ Street number found

□ Street name not found

□ No street number

□ Number range not found.

%GCBATCH Macro Syntax

The %GCBATCH macro accepts the following information:

□ The name of the address data set to geocode

□ The variable names in the address data set

□ The name of the map entry

□ The libref in which to store geocoding data sets, and whether they should be recreated

□ The name of the ZIP code centroids data set

□ The names of any additional polygonal composites to add to the address data set.

Note: The CH, NODE, DETNODE, FNAME, FTYPE, PRE, SUF, FX, FY, TX, TY, COUNTY, BLOCK, PLACE, STATE, TRACT, ZIP, PLUS4, FRADD, and TOADD arguments are not used in SAS/GIS software beginning with Version 8. These arguments are ignored if you specify them. △

The %GCBATCH macro has the following general form:

%GCBATCH(

<GLIB=*geocoding-library*>,

<ZIPD=*ZIP-centroids-data-set*>,

GEOD=*address-data-set*,
<NV=*name-var*,>
AV=*address-var*,
CV=*city-var*,
<SV=*state-var*,>
<ZV=*ZIP-var*,>
<P4V=*ZIP+4-var*,>
MNAME=*map-entry*,
<PV=*area-composite-list*>,
<NEWDATA=*new-data-value*>);

where

AV=*address-var*
Specifies the name of the variable that stores the street address in the address data set that you want to geocode. This parameter is required.

CV=*city-var*
Specifies the name of the variable that stores the city name portion of the address in the address data set that you want to geocode. This parameter is required.

GEOD=*address-data-set*
Specifies the address data set that you want to geocode. The *address-data-set* argument should use the form *libref.data-set-name*. This parameter is required.

GLIB=*geocoding-library*
Specifies the libref for the SAS data library where all of the sorted and summarized chains, nodes, and details data sets that are created for the geocoding process are stored.

Note: The SAS data library that you specify for the *geocoding-library* argument should be on a volume that has a large amount of free space because the geocoding data sets can be quite large. Also, to take full advantage of the geocoding facility, you should specify a permanent SAS data library. The default for this variable is WORK, but data sets in the WORK library are deleted when the SAS session is terminated, so the geocoding data sets will be lost. If geocoding data sets already exist in the specified library at the start of the geocoding process, the geocoding facility checks their creation dates against the creation date of the chains data set. The geocoding data sets will be recreated only if the chains data set has a more recent creation date. The first time that you geocode with a particular chains data set the process will take considerably longer because these geocoding data sets are being created, sorted, and indexed. Subsequent geocoding times, however, will be much faster as long as the chains data set has not been modified. This parameter is optional. △

MNAME=*map-entry*
Specifies the name of the GISMAP entry for the SAS/GIS spatial database that you are using for geocoding. The geocoding process uses the projection information in the map entry to ensure that the X and Y coordinates that are returned for the address will be in the same coordinate system as the spatial data for the map. The *map-entry* argument should use the form *libref.catalog-name.entry-name*. This parameter is required.

NEWDATA=YES | NO
Specifies whether the geocoding data sets are recreated. The default is NEWDATA=NO. If you set NEWDATA=NO, the geocoding facility searches the SAS data library that you specified with the GLIB macro variable for geocode data

sets that were created for the spatial entry. The geocoding facility checks the creation date of existing geocode data sets against the creation date of the spatial entry. If the creation date of the geocode data sets is more recent than the creation dates of the spatial entry, the geocoding facility uses the geocode data sets. Otherwise it creates new geocode data sets.

Use NEWDATA=YES to force the geocoding facility to build new versions of the geocoding data sets. You need to specify NEWDATA=YES if the existing geocoding data sets were created with an earlier version of SAS/GIS software. This parameter is optional.

NV=*name-var*

Specifies the name of the variable that stores the name portion of the address in the address data set that you want to geocode.

This parameter is optional.

PV=*area-composite-list*

Specifies the list of polygonal (area) composite values that you want added as variables to the address data set along with the X and Y coordinates of the address. By default, no other variables are added. Use spaces to separate composite names in the list. For example, the following specification adds the county and census tract and block values along with the address coordinates:

```
pv=county tract block,
```

This parameter is optional.

P4V=*ZIP+4-var*

Specifies the name of the variable that stores ZIP+4 postal codes in the address data set that you want to geocode.

This parameter is not required, but the accuracy of the geocoding process may be reduced if you omit it.

SV=*state-var*

Specifies the name of the variable that stores the state name portion of the address in the address data set you want to geocode.

This parameter is not required, but the accuracy of the geocoding process may be reduced if you omit it.

ZIPD=*ZIP-centroids-data-set*

Specifies a data set that contains the coordinates of the centers of ZIP code zones. (If an address includes a ZIP code and the street address cannot be matched, the geocoding facility supplies the ZIP code centroid coordinate instead of the address coordinate.) The default is ZIPD=SASHELP.ZIPCODE, which specifies the SASHELP.ZIPCODE data set that is supplied with SAS/GIS software. This parameter is optional.

ZV=*ZIP-var*

Specifies the name of the variable that stores the ZIP code portion of the address in the address data set that you want to geocode.

This parameter is not required, but the accuracy of the geocoding process may be reduced if you omit it.

Batch Geocoding Example

The following example shows a typical set of variable assignments for the %GCBATCH macro:

```
%gcbatch(glib=geoperm,
         geod=geoperm.dcaddr,
         nv=name,
         av=addr,
         cv=city,
         sv=state,
         zv=zip,
         mname=dcmap.dcmap.dcmap,
         pv=block county tract);
```

After you submit the %GCBATCH macro, issue the following command from any SAS System command line:

```
af c=sashelp.gis.geocodeb.scl
```

The results are written to the address data set, GEOPERM.DCADDR. If a WRITE failure occurred, then the results are in the temporary work data set, WORK._GEOCODE.

Hints and Tips

1. To ensure good quality and accurate gecoding results, you must use accurate data. If your map's address data are incomplete or out of date, your geocoding will not deliver the results you want.
2. You can use the generic point import to import the points onto the map. However, before you import the points, you must make sure that your address data set contains a variable that is named ID that has a unique value for each point.
3. Notes on the Address Data Set

   The address data set is a SAS data set that contains the addresses that you want to geocode. It should contain variables for the street address, city, state, and ZIP code (optionally the ZIP+4 code) of the addresses to be matched. The address data set can also contain a name that is associated with the address, but the name is not used in the address matching.

   In order for the geocoding facility to accurately parse the addresses, follow these guidelines:

   - Use only street addresses. P.O. boxes, rural routes, grid addresses, and addresses with alphanumeric characters cannot be geocoded. An address containing a post office box or a rural route address in addition to a street address should not cause a problem.
   - The street number portion of the street address should not contain nonnumeric characters. For example, an address such as **501-B Kent St** will be matched to **501 Kent St.**, not to the full address containing the nonnumeric character. Apartment numbers should be stored in separate variables rather than appended to the street number.
   - Use the following values for directional prefixes and suffixes, with no punctuation or spaces between letters:

     N S E W NE NW SE SW
   - Avoid using abbreviations that conflict with street name abbreviations. For example, do not use **St John St**. Use **Saint John St** instead. Spelling out **Saint** reduces chances for confusion.

   Note: The results from the geocoding are written back to this data set, so you must have WRITE access to it or make a copy you can write to. △

The correct bibliographic citation for this manual is as follows: SAS Institute Inc., *Working with Spatial Data Using SAS/GIS® Software, Version 8*, Cary, NC: SAS Institute Inc., 1999.

Working with Spatial Data Using SAS/GIS® Software, Version 8

ISBN 1–58025–519–1

SAS Institute Inc., SAS Campus Drive, Cary, North Carolina 27513.

1st printing, October 1999

The Institute is a private company devoted to the support and further development of its software and related services.