

Red Rover, Red Rover, Send Data Right Over: Exploring External Geographic Data Sources with SAS®

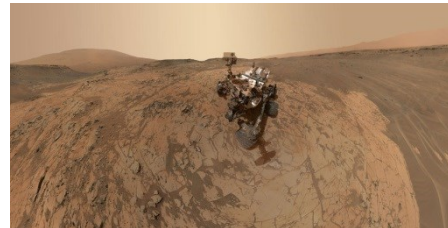
Louise S. Hadden, Abt Associates Inc., Cambridge, MA

ABSTRACT

The intrepid Mars Rovers have inspired awe and Curiosity - and dreams of mapping Mars using SAS/GRAPH®. This presentation will demonstrate how to import SHP file data (using PROC MAPIMPORT) from sources other than SAS and GfK to produce useful (and sometimes creative) maps. Examples will include mapping neighborhoods, ZCTA5 areas, postal codes and of course, Mars. Products used are Base SAS® and SAS/GRAPH®. SAS programmers of any skill level will benefit from this presentation.

INTRODUCTION

The focus of this presentation is to demonstrate how the PROC MAPIMPORT procedure can be used to extend SAS/GRAPH's mapping capabilities beyond maps commonly available through SAS, and even beyond the confines of earth.



1 Curiosity Rover

PROC MAPIMPORT

The heroes of this presentation are the Mars Rovers – of course. Running a close second is PROC MAPIMPORT. PROC MAPIMPORT has been available since SAS 9.2. This procedure enables you to import ESRI shapefiles and convert them into SAS map data sets that can be used by SAS/GRAPH. MAPIMPORT in itself does not produce any graphics output, but does aid in the production of output from PROC GMAP and other “mapping” procedures.

There are innumerable SHP files available on the internet. SHP files usually come with companion files, two of which are necessary / useful for the PROC MAPIMPORT procedure. Make sure you extract the DBF file (metadata for the “map” file), the SHX file (shape information for the “map” file), and the SHP file (combines the metadata and shape information) from any SHP file zip files into the same directory as the SHP file.

There are a couple of procedure options (CONTENTS and CREATE_ID_, new in later versions of SAS version 9) and some statements (exclude, rename, id and select) available with PROC MAPIMPORT. Make sure you look at the latest version of the SAS documentation in order to find any information on CONTENTS and CREATE_ID_. These options are great time-savers, although there are some cons associated with the CREATE_ID_ option. The CONTENTS options shows information about the SHP files including field names and types in the SAS log. CREATE_ID_ creates a map ID variable named _ID_ with a unique value for each polygon in the map. Unlike SAS map data sets, imported SHP files do not automatically have an ID variable for polygons. This means you have to use an ID statement OR the CREATE_ID_ procedure option in order to have an ID. It is possible to create and use a SAS map data set without an ID statement, but generally output looks nicer with an ID variable. In general this procedure is a little chicken and egg – you need to run it to see what the contents are, explore what ID variables might be available and in play, and then ideally rerun it with an ID statement when you have ascertained what the ID variables are. If you get extraneous lines in your image it is because either you have not used a (correct) ID statement or the image is too detailed, in which case you need to run PROC GREduce on your newly created map data set.

SYNTAX OF PROC MAPIMPORT

Several examples of using PROC MAPIMPORT are shown below. The DBF, SHX and SHP files should all be available in the same logical space.

```
PROC MAPIMPORT OUT=map-data-set DATAFILE='path-to-shapefile '
  <CONTENTS>
  <CREATE_ID_>;
  EXCLUDE 'field-identifier(s)';
  ID 'field-identifier(s)';
  RENAME 'field-identifier-1'=variable-name-1...;
  SELECT 'field-identifier(s)'
```

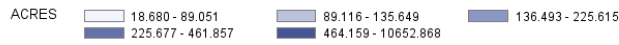
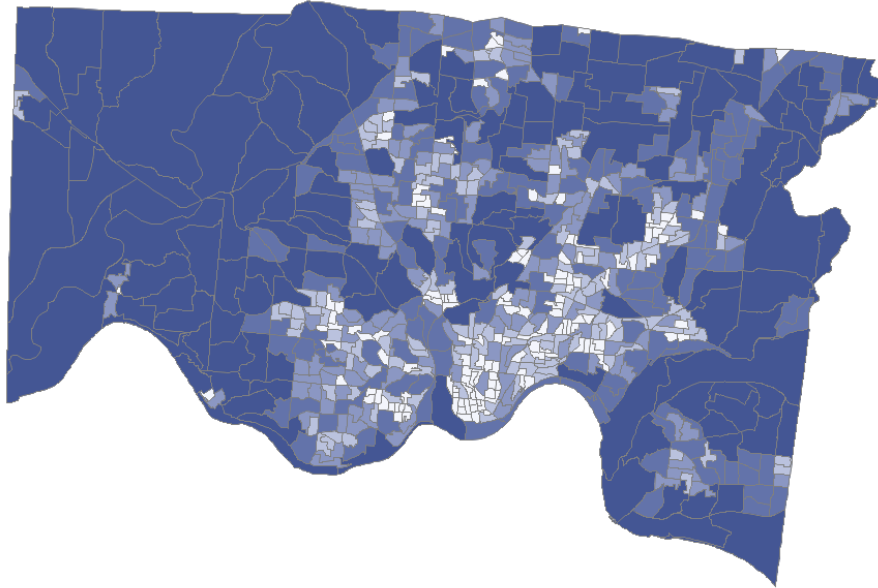
EXAMPLES

CENSUS BLOCK GROUPS IN CINCINNATI

The source of data used for this example is the Cincinnati Area Geographic Information System (CAGIS), which has a lot of useful SHP files and Cincinnati area information. SHP files often come with “mappable” data included: in this case, I chose acres as the response variable and 5 levels in my PROC GMAP code. I opted to use the CREATE_ID_ procedure option since I knew nothing about the SHP file in advance, and consulted the CONTENTS listing in the log (and looked at a test print).

```
PROC MAPIMPORT DATAFILE='.\cenbgr2010.shp' OUT=cenbgr2010 CONTENTS
  CREATE_ID_;
RUN;
PROC PRINT DATA=cenbgr2010 (OBS=5) NOOBS;
RUN;;
ODS HTML FILE='test_cenbgr2010.html';
ODS PDF FILE='test_cenbgr2010.pdf' NOTOC;
GOPTIONS RESET=all DEVICE=png HSIZE=9in VSIZE=7.5in;
TITLE1 "Census Block Groups in Cincinnati";
PROC GMAP MAP=cenbgr2010 DATA=cenbgr2010 ALL;
  ID _id_;
  CHORO acres / COUTLINE=gray LEVELS=5 NAME='cenblkgr';
RUN;
QUIT;
ODS PDF CLOSE;
```

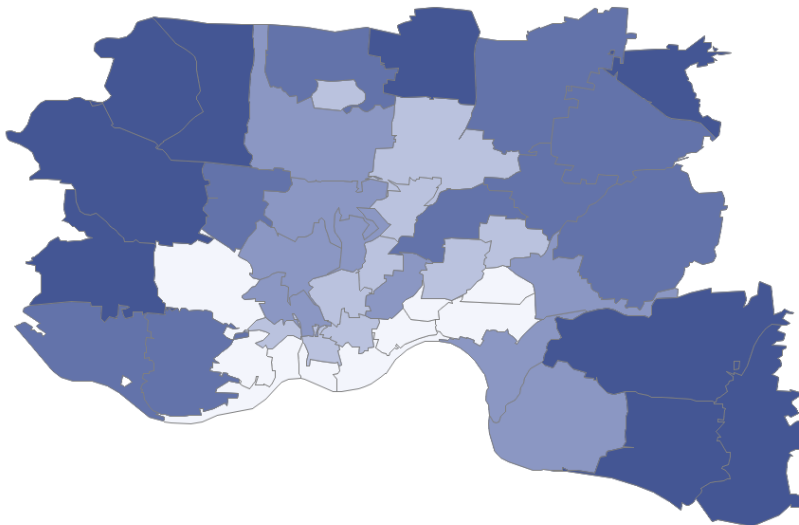
Census Block Groups in Cincinnati



ZIP CODE TABULATION AREAS IN CINCINNATI

The source of data used for this example is Bureau of the Census, which has SHP files for ZCTAs (Zip Code Tabulation Areas.) In this case, I used Zip Code (the ID variable) as the response variable and 5 levels in my PROC GMAP code. The ZCTA SHP file contained information from all Zip Codes in the US, so I used the SASHELP.ZIPCODE file to trim down the map, keeping only Zip Codes from Cincinnati.

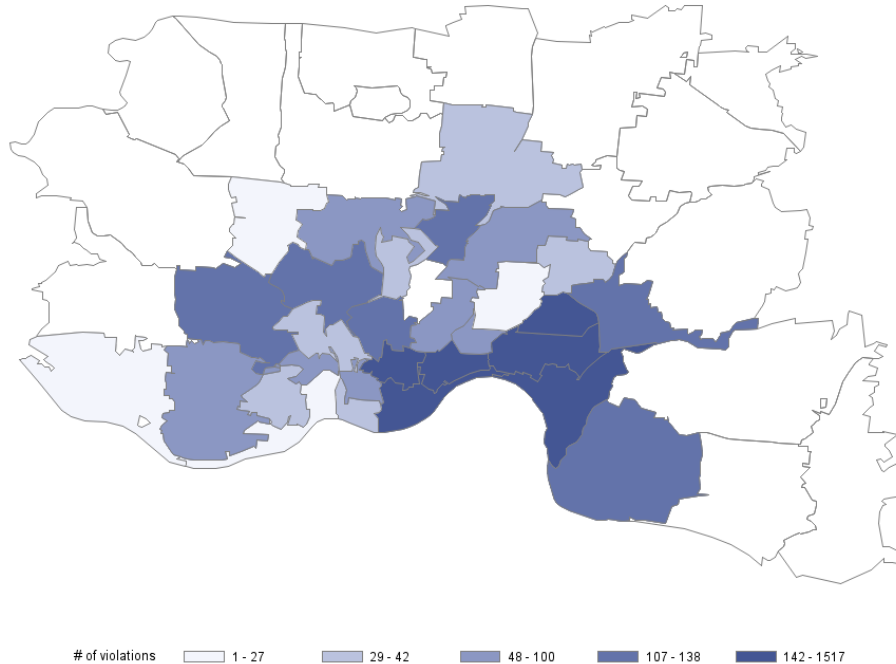
Zips in Cincinnati



FOOD VIOLATIONS BY ZIP CODE IN CINCINNATI (MARCH 2015)

The map above looks nice enough, but it really doesn't convey any information. The sources of data used for this example are Bureau of the Census (for the ZCTA map) and CAGIS (for the food violations data). In this case, I used the sum of food violations per zip code / ZCTA as the response variable and 5 levels in my PROC GMAP code. It is clear from the map that the food violation data does not cover all zip codes, so more research is needed. Sometimes that is reason enough to create a map!

of Food Violations by zip code in Cincinnati March 2015



POSTAL CODES IN AUSTRALIA

PROC GEOCODE was introduced in SAS 9.2. It is the process by which you can convert address information into map locations. It is a part of the SAS/GRAPH product. It was originally for U.S. data points, but you can now import free postal code (lookup) data (and map shp files to present geocoded data points on if desired) from Great Britain and Australia as well. The procedure (and the data behind it) has been updated as of SAS 9.3M2. In 9.3M2, you can use PROC GEOCODE with international cities, and in SAS 9.4, Canadian street-level geocoding is available. Geocoding is available on a number of levels, From City to Street to IP Address. To import a custom map data set to use with PROC GEOCODE, such as postal codes in Australia, I used PROC MAPIMPORT.

```
PROC FORMAT;
  VALUE $ p2state '2'='New South Wales'
                 '0'='Northern Territory'
                 '4'='Queensland'
                 '5'='South Australia'
                 '7'='Tasmania'
                 '3'='Victoria'
                 '6'='Western Australia';
  VALUE $ p2stab '2'='NSW'
                '0'='NT'
                '4'='QLD'
                '5'='SA'
```

```

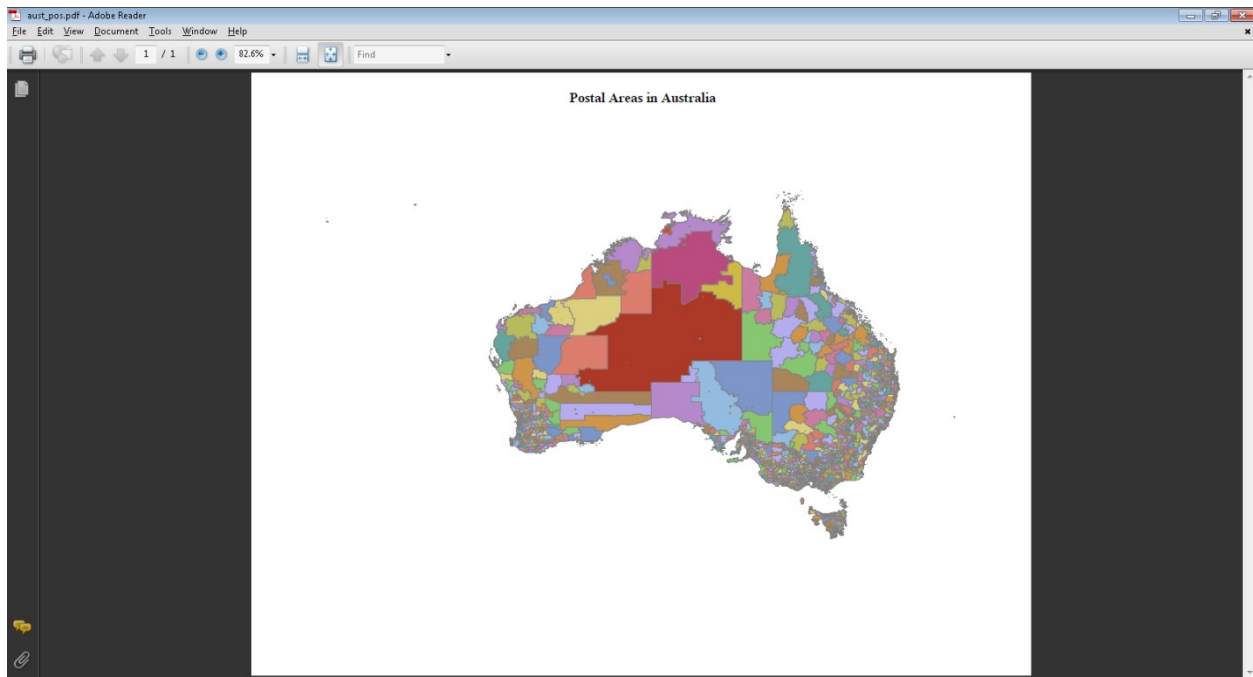
        '7'='TAS'
        '3'='VIC'
        '6'='WA';
RUN;

PROC MAPIMPORT DATAFILE='.\poa_2011_aust.shp' OUT=dd.pos_2011_aust_map;
    ID poa_code;      /* new in 9.4 - add contents, add create_id statements */
RUN;

DATA dd.pos_2011_aust_map;
    LENGTH state_name $ 30 st_abbr $ 4;
    SET dd.pos_2011_aust_map;
    state_name=PUT(SUBSTR(LEFT(poa_code),1,1),$p2state.);
    st_abbr=PUT(SUBSTR(LEFT(poa_code),1,1),$p2stab.);
    IF poa_code='9191' THEN state_name='Unclassified SAls';
    IF poa_code='9494' THEN state_name='No Usual Address SAls';
    IF poa_code='9797' THEN state_name='Migratory/Off-Shore/Shipping SAls';
run;

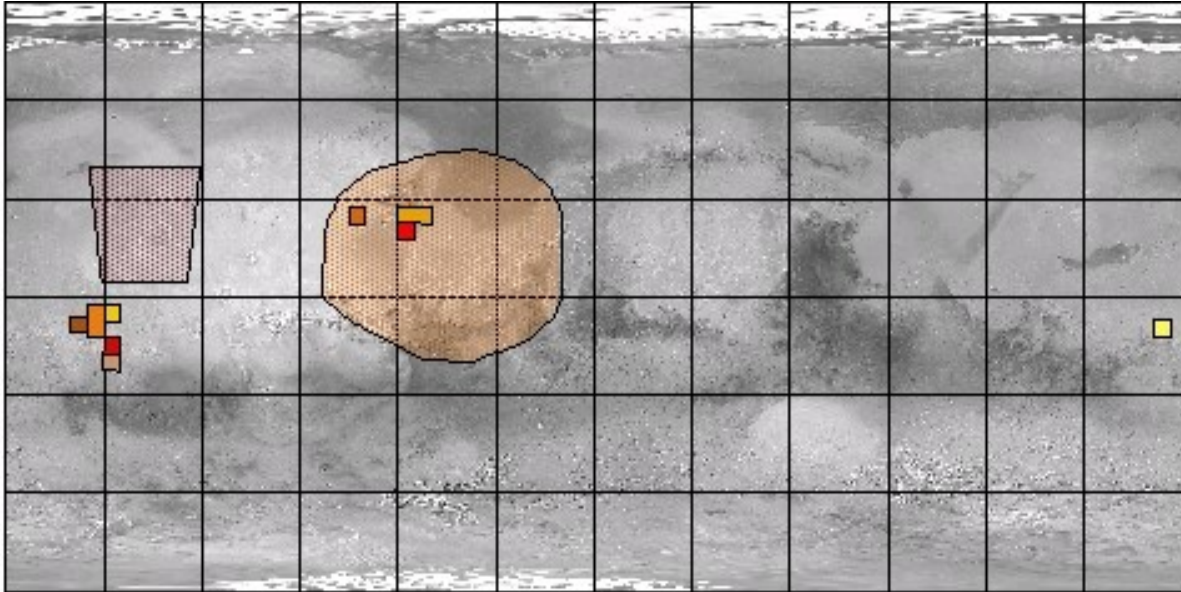
ODS PDF FILE='aust_pos.pdf' NOTOC;
GOPTIONS RESET=ALL DEVICE=png HSIZE=9in VSIZE=7.5in;
TITLE1 "Postal Areas in Australia";
PROC GMAP MAP=dd.pos_2011_aust_map
    DATA=dd.pos_2011_aust_map all;
    ID poa_code;
    CHORO poa_code / COUTLINE=gray LEVELS=10;
RUN;
QUIT;
ODS PDF CLOSE;

```



MARS!

The source of the data for the MARS examples is U.S.G.S. Planetary GIS Web Server (PIGWAD: Planetary Interactive G.I.S.-on-the-Web Analyzable Database.) The image below highlights areas that have SHP file data. For this paper and presentation, the two largest images were mapped.



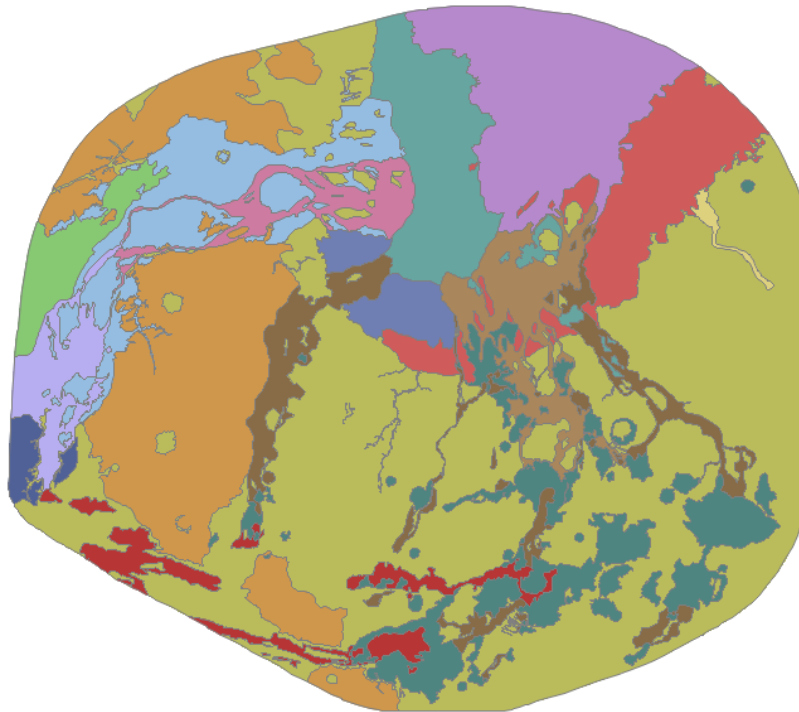
The first MARS image mapped was the Chryse area. Since I know nothing about this data, I used both CONTENTS and CREATE_ID_, and reviewed the log and the list. I used the SAS created _ID_ variable as the response variable and the name to identify the shapes.

















```
PROC MAPIMPORT DATAFILE='.\chryse_dd.shp' OUT=chryse CONTENTS CREATE_ID_ ;  
RUN;
```

```
PROC PRINT DATA=chryse (OBS=5) NOOBS;  
RUN;
```

```
ODS HTML FILE='test_chryse.html';  
ODS PDF FILE='test_chryse.pdf' NOTOC;  
GOPTIONS RESET=all DEVICE=png HSIZE=9in VSIZE=7.5in;  
TITLE1 "Chryse";  
PROC GMAP MAP=chryse DATA=chryse all;  
  ID _id_;  
  CHORO name / COUTLINE=gray NAME='Chryse';  
RUN;  
QUIT;  
ODS PDF CLOSE;
```

Chryse



NAME			
		Chryse ridged plains material	
		Chryse unit 3	
		Kasei unit 1	
		Member 4 of Tharsis Formation	
		Syria Planum Formation	
		Xanthe outflow channels	
			
			
			
			

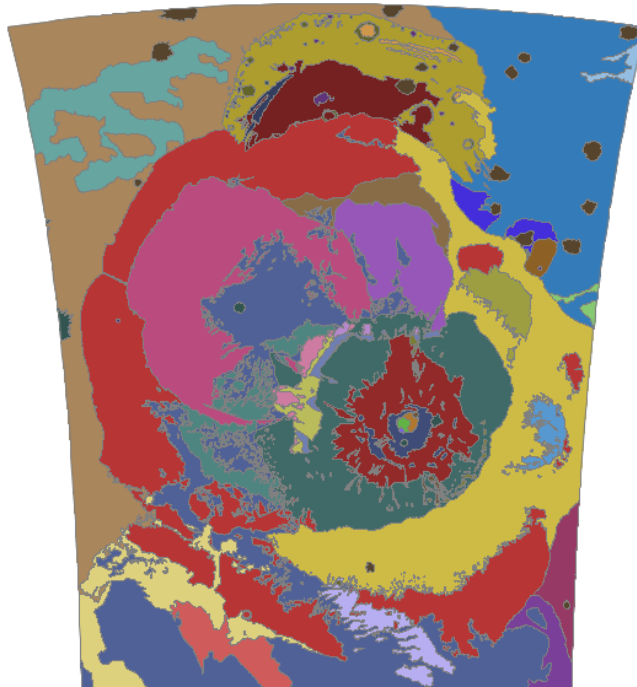
The second MARS image mapped was the I2327g area. The code is identical to that used for the Chryse area, except for the file names.

```
PROC MAPIMPORT DATAFILE='.\i-2327_g_dd.shp' OUT=i2327g CONTENTS CREATE_ID_ ;
RUN;
```

```
PROC PRINT DATA=i2327g (OBS=5) NOOBS;
RUN;
```

```
ODS HTML FILE='test_i2327g.html';
ODS PDF FILE='test_i2327g.pdf' NOTOC;
GOPTIONS RESET=all DEVICE=png HSIZE=9in VSIZE=7.5in;
TITLE1 "i2327g";
PROC GMAP MAP=i2327g
    DATA=i2327g ALL;
    ID _id_;
    CHORO uname / COUTLINE=gray NAME='i2327g';
RUN;
QUIT;
ODS PDF CLOSE;
```

i2327g



UNAME	AHosc	AHtm	Aa1	Aa3	Aab	Aah	Aam
Aar	Aas	Acf	Amp	Amr	Ams	Aoal	
Aoama	Aoamb	Aoamc	Aoamd	Aoame	Aoau	Aoc1	
Aoc2	Aoc3	Aop	Aos1	Aos2	Aos3	Aos4	
Atm	HNosc	Hal	Hf	Hhf	Hhh	Hhp	
Naf	Nam	Nap	ca	cb	cc	cd	

CONCLUSION

PROC MAPIMPORT is a very handy SAS tool which allows graphic displays of areas without SAS-supplied maps, such as zip code (ZCTAs), census block groups, health service areas, etc. Even the planets (Mars!) can be mapped using SAS in conjunction with this procedure.

REFERENCES

Bureau of the Census. Accessed September 2016. https://www.census.gov/geo/maps-data/data/cbf/cbf_zcta.html

Cincinnati Area Geographic Information System (CAGIS.) Accessed September 2016. <http://cagismaps.hamilton-co.org/cagisportal/mapdata>

U.S.G.S. Planetary GIS Web Server (PIGWAD: Planetary Interactive G.I.S.-on-the-Web Analyzable Database.) Accessed September 2016. <http://webgis.wr.usgs.gov/pigwad/maps/mars.htm>

ACKNOWLEDGMENTS

The author would like to acknowledge the many contributions of Darrell Massengill, Robert Allison, Mike Zdeb and Liz Simons, who champion mapping with SAS and never fail to answer my many questions.

CONTACT INFORMATION

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

Louise S. Hadden
Abt Associates Inc.
617-349-2385
Louise_hadden@abtassoc.com
abtassociates.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.