Temporal and Spatial Visualization of Epidemiological Data

Sanjaya Kumar, New York State Department of Health, Troy, NY

Geographical Information Systems (GIS) provide powerful tools for visualizing spatial data. With the increased availability of georeferenced data, there is an increasing need to visualize data temporally and spatially. As part of the CDC-funded New York State Environmental Public Health Tracking (EPHT) project, the New York State Department of Health (NYSDOH) is developing a surveillance system that links and displays environmental hazard, health outcome, and sociodemographic data. This paper presents an example of how the system can be used to regularly track ozone levels and hospitalizations and to compare air pollution and health outcomes in temporal and spatial dimensions.

Ambient air pollution and health outcome data were geographically linked in order to develop a graphical display from within a GIS application. SAS® version 9.1 was used to perform data analyses and to create charts and graphs, and MapInfo® was used for linkage and display of maps and SAS outputs. Such linkage allows users to interact with the map view and plots by controlling the analysis and display of the data. Examples of various output and screenshots from our interactive, comprehensible menu illustrate the interactive maps and SAS graphs. The graphs created by this program enable users to visualize data, spot trends, and make effective visual comparisons of ambient air pollution and health outcomes between different geographical areas and time periods.

Background:

The EPHT system is used to routinely gather, integrate, and analyze data on ambient air pollutants and health outcomes of interest, and to spatially relate these datasets using GIS. Geographical representation is a vital component in understanding disease dynamics and etiology.

Improving and protecting public health requires a detailed understanding of the relationship between the environment and health effects. Accurate and timely environmental and health surveillance data can aid public health professionals in determining disease impact, recognizing clusters, and identifying populations and areas most affected or at risk. Environmental health surveillance can also help prioritize and direct critical resources, transmit health alerts to affected populations and areas, and develop effective community based initiatives to address the interaction between human and its surrounding environment.

Materials and Methods:
We used SAS® version 9.1 to develop graphical displays of data and then linked the graphical outputs using MapInfo Professional® (MIPro) version 8.0 and MapBasic® version 7.0. The data used in this presentation is for hourly measurement of ozone and number of asthma hospital admissions for each day. These datasets were linked based on geographical location. New York State Department of Environmental Conservation (NYSDEC) provides the ozone monitoring data for New York State. Hospitalization data is provided by SPARCS. Data used for this presentation is from the publicly available datasets or from the non confidential aggregated data.

The application requires that SAS is installed on user's workstation. Users can start the application from MapInfo Professional® by selecting the "Enter Criteria.." from the dropdown menu, as shown in figure 1. Once started, the application checks for the SAS installation on user's workstation.

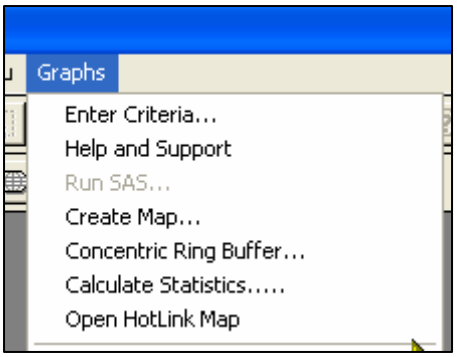

Figure 1: Graph Menu

After finding the SAS, the query application screen opens for user input. This query application, shown in Figure 2, was built using MapBasic® and provides options for setting the parameters needed to create graphical output using PROC GCHART and GPLOT in SAS. Graphical output from SAS are created using ODS and are saved as image files for later visualization in MapInfo®. The number of graphs/charts created by this application depends on the unit of the geographical area selected by the user. If the user selects "county," one output for each county is created per health outcome or pollutant. Users can also select a combination of predefined geographical scales.

The initial query application screen is divided in three sections: data and output format selection, time period for output, and geographical scale selection. The application opens with default values which users can change, as required. Each item on the application screen presents choices to select from. Users can select the time period (daily, weekly or monthly), start date, end date, pollutant, and health outcome, from a predefined menu. Users can also select and compare air pollution and health outcome data for different years simultaneously. There are additional options for creating complex SAS output, such as right vertical axis and calculation of rates for creating charts and plots. Users can also select the geographical level of display ranging from ZIP code to 36 kilometer grid across New York State.

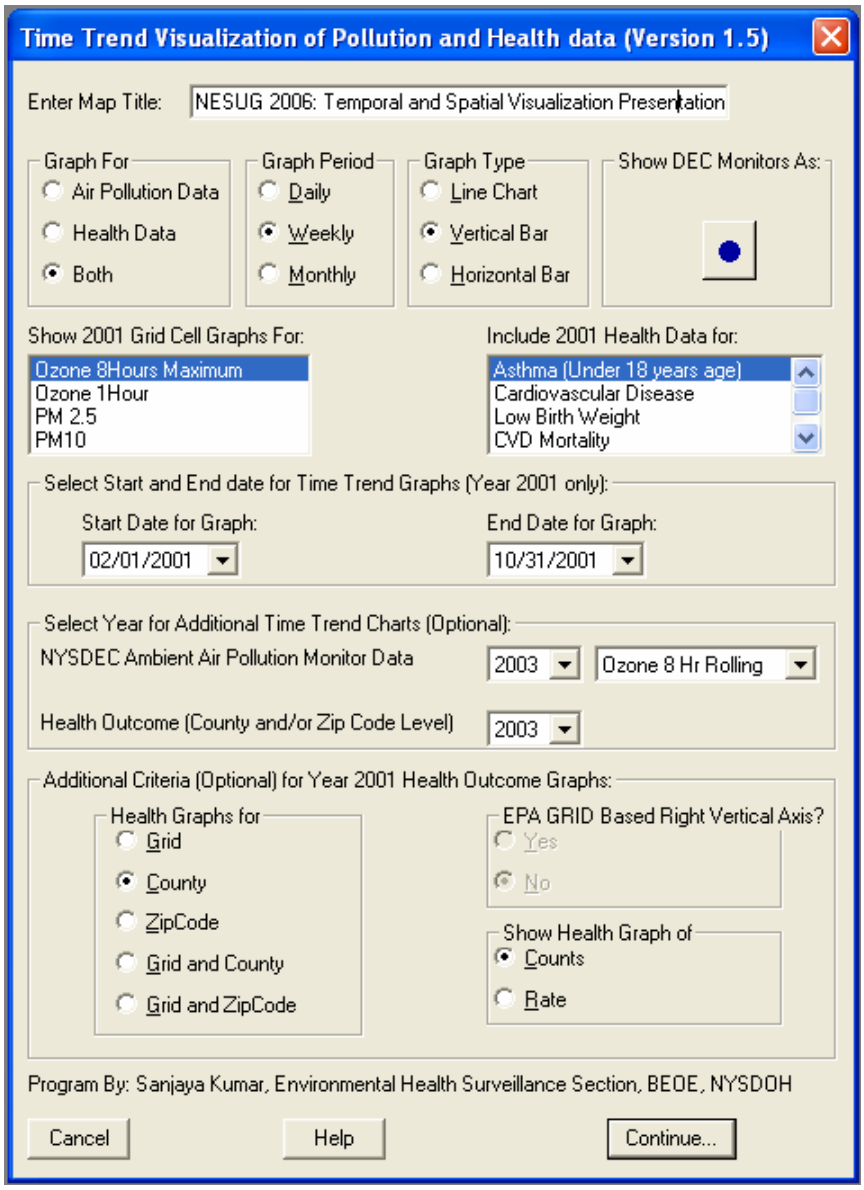

Figure 2: Criterion selection screen.

After clicking Continue, the selected parameters are saved in a temporary folder as a dbase 4 file on users' workstations. The file serves as a trigger for MapBasic to start SAS in batch mode. SAS reads this dBase 4 file to obtain the parameters, as shown in figure 3.

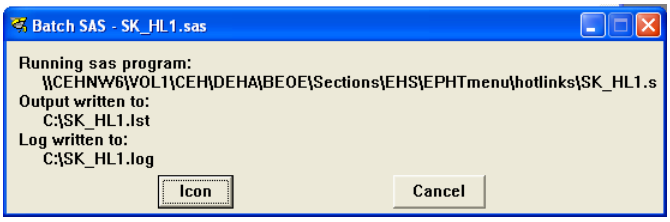

Figure 3: SAS running in Batch Mode on user workstation.

```
%let Mapdir=Z:\projects\EPHT\Demo\NESUG\;

* Find the temporary storage directory MapBasic is using;
filename finddir "&Mapdir.tempdir.dbf";

* MapBasic places its temp directory location in a dbase
file;
proc dbf db4=finddir out=finddir; run;
proc print; run;

**put the temporary storage directory in a macro variable;
data _null_;
     set finddir;
     call symput ("MBdir",trim(dirname));
run;

**get user selection data that MapBasic saved for SAS;
filename MBdata "&MBdir.MIGraph.dbf";
proc import datafile="&mbdir\MIGraph.dbf"
            out=mydata
            replace;
quit;
```

The following is the output of the user's selection:

```
Obs DIRNAME
 1  C:\DOCUME~1\sxk10.000\LOCALS~1\Temp\
-------------------------------------------------------------------------------------------------------------------

Obs   DRIVEL   I_END1   I_PERIOD1   I_CMAQ1   I_START1   M_YEAR1   H_YEAR1   I_HEALTH1   I_GTYPE1   I_GAREA1   I_RIGTHV1
 1     P:\        10          2          1          2         13        13           1          2          2           2
Obs   I_CTORRT1    I_MYD1    I_MOREDEC    UR_TITLE
 1          1           3           2      NESUG 2006: Temporal and Spatial Visualization Pre
```

Figure 4: User selected parameters read by SAS

As shown in Figure 4, the title entered during parameter selection is included in the output. After reading user input, a macro is initiated to read the data required for creating the graphs and charts:

```
%macro ud_db;
     %if &syserr ne 0 %then %goto err_dataset;
      %else %goto yes_dataset;

      %err_dataset:
      proc export data=IcantDoIt
```

```
            outfile="&MBdir\SASMIDone.DBF"
               replace;
        %goto finish;

        %yes_dataset:
       Data Environ;
         Set EData.Ozone;
        run;
---- More SAS Codes to subset the data and create graphical outputs -----
```

Once a subset is created using the user's selection criterion, a macro is initiated to select the codes for execution.

```
%LET OutLoc = &mbdir;
%MACRO AIR_Graph;
%IF &IOnlyNeed=Pollution OR &IOnlyNeed=Both %THEN %DO;
proc sort data=work.EPA_2(keep=CellID) out=work.Cells
nodupkeys;
by CellID;
run;
```

The macro ***AIR_Graph*** will execute only if the user has selected to create graphs/charts for air pollution or for a combination of air pollution and health data. This step saves times, particularly when there are multiple outputs for a selected geographical region. Users can select only one of the four air pollutants in creating graphs. Additional macros are required to further narrow the code execution process and, in turn, reduce the time required to generate output.

Once macro AIR_Graph is initiated, it checks the parameter to find the selected pollutant and executes the codes related to the user-selected air pollutant.

```
%IF &DSETNAME EQ OZONE8HR %THEN %GOTO Graph_O8H;
   %ELSE %IF &DSETNAME EQ OZONE1HR %THEN %GOTO Graph_O1H;
      %ELSE %IF &DSETNAME EQ PM25 %THEN %GOTO Graph_PM25;
          %ELSE %GOTO Graph_PM10;
%GOTO Graph_Finish;

SYMBOLGEN:  Macro variable DSETNAME resolves to OZONE8HR
MLOGIC(CMAQ_GRAPH):  %IF condition &DSETNAME EQ OZONE8HR is TRUE
MLOGIC(CMAQ_GRAPH):  %GOTO GRAPH_O8H (label resolves to GRAPH_O8H).
MLOGIC(DOITOZ8):  Beginning execution.
```

```
%GRAPH_O8H:
%macro doitOz8;
  %do i = 1 %to &hm;
    %let which = %scan(&cells,&i);
filename out "&OutLoc\&which..png";

&proc_type data=work.EPA_2(where=(CellId="&which"));
where GRAPHfor='O8';
Format Obs_date Date5. Month Monfmt.;
&plot_type;
Title1 color=CX090766 font=SwissXB height=2.0 "&TitWord
Average 8Hr Max Ozone (AIR Model)";
Title2 color=CX090766 font=Swissbi height=1.5 "Cell
ID=&which";
run;
  %end;
%mend doitoz8;
%doitOz8;
quit;

%GOTO GRAPH_FINISH;
```

The next step is the creation of associated files for MapInfo Professional, the GIS application, in order to process the SAS graphs as a map layer. This step is accomplished by creating a *.tab file for each graphical output file. Macro ***doitcelltab*** creates one *,tab file for each image file generated by PROC GCHART. A dataset named cells, generated by PROC SORT provides the filename based on cell ID. Each *.tab file simply includes coordinate information to display the image in MapInfo®, image file location, and image name. These parameters are required by the MapInfo® to recognize each image file as a “layer”.

```
%GRAPH_FINISH:

%MACRO MakeEPATabs;
%IF (&IOnlyNeed=Pollution OR &IOnlyNeed=Both) %THEN %DO;
PROC SORT DATA=EPA_2(keep=CellID)
out=work.Cells nodupkeys; by CellID;
run;

%macro doitCelltab;
  %do i = 1 %to &hm;
    %let which = %scan(&cells,&i);
filename out "&OutLoc\&which..tab" ;
data _null_;
```

```
file out;
fileline1="&which..png";
put '!table';
put '!version 300';
put '!charset WindowsLatin1';
put ;
put 'Definition Table';
put 'File ';
put '"' fileline1'"';
put ' Type "RASTER"';
put '  (0,1) (0,0) Label "Pt 1",';
put '  (1,1) (1,0) Label "Pt 2",';
put '  (0,0) (0,1) Label "Pt 3"';
put '  CoordSys NonEarth Units "ft"';
  put 'Units "ft"';
run;
%end;
%mend doitcelltab;
%doitcelltab;
quit;
%END;
%Mend MakeEPATabs;
%MakeEPATabs;
```

Figure 5, a screen image of a log window, shows the location of *.tab file generated for each graph produced by PROC GCHART. Each graph/chart generated by SAS is hyperlinked to the user-selected geographical unit, and MIPro opens these image files as a mapping layer.

```
MPRINT(DOITCELLTAB):   filename out "C:\DOCUME~1\sxk10.000\LOCALS~1\Temp\\0811670.tab" ;
MPRINT(DOITCELLTAB):   data _null_;
MPRINT(DOITCELLTAB):   file out;
SYMBOLGEN:  Macro variable WHICH resolves to 0811670
MPRINT(DOITCELLTAB):   fileline1="0811670.png";
MPRINT(DOITCELLTAB):   put '!table';
MPRINT(DOITCELLTAB):   put '!version 300';
MPRINT(DOITCELLTAB):   put '!charset WindowsLatin1';
MPRINT(DOITCELLTAB):   put ;
MPRINT(DOITCELLTAB):   put 'Definition Table';
MPRINT(DOITCELLTAB):   put 'File ';
MPRINT(DOITCELLTAB):   put '"' fileline1'"';
MPRINT(DOITCELLTAB):   put ' Type "RASTER"';
MPRINT(DOITCELLTAB):   put '  (0,1) (0,0) Label "Pt 1",';
MPRINT(DOITCELLTAB):   put '  (1,1) (1,0) Label "Pt 2",';
MPRINT(DOITCELLTAB):   put '  (0,0) (0,1) Label "Pt 3"';
MPRINT(DOITCELLTAB):   put '  CoordSys NonEarth Units "ft"';
MPRINT(DOITCELLTAB):   put 'Units "ft"';
MPRINT(DOITCELLTAB):   run;

NOTE: The file OUT is:
      File Name=C:\DOCUME~1\sxk10.000\LOCALS~1\Temp\\0811670.tab,
      RECFM=V,LRECL=256

NOTE: 13 records were written to the file OUT.
      The minimum record length was 0.
      The maximum record length was 30.
```

Figure 5: SAS log showing the location of *.tab file associated with each graphical output.

Using MapInfo to link and view the SAS generated graphs/charts provide additional functions such as changing size and zoom level, placing in the layout along with thematic maps, adding a title, and selecting a section of the graph for display.

After processing codes for each geographical region in the selected area, SAS continues to the final step: creating a trigger file for MapBasic which indicate the completion of SAS processing.

```
%Finish:
*Create a dummy data for MapBasic to know that its wakeup
time!;

proc export data=IamDone
       outfile="&MBdir\SASMIDone.DBF"
          replace;
%mend Ud_db;
%ud_db;
```

Once this file is created in the specified directory, MapBasic directs MapInfo to open SAS-generated output files and link with the preloaded New York state map using information embedded in the SAS-generated output (Figure 6).

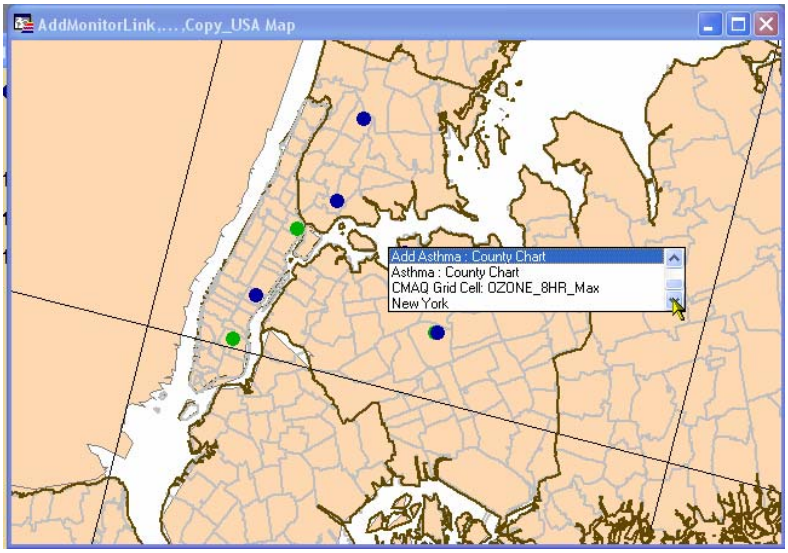

Figure 6: Map showing the hyperlinked SAS output with user selected geographical area.

Users can click on any of the geographical areas linked with SAS output to open the air pollution and/or health outcome bar charts. As shown in Figure 6, multiple SAS outputs are lined to a single geographical area. Any number

of SAS outputs can be opened in a given session, and sized as needed. Each SAS output opens as a separate layer, as shown in Figure 7.

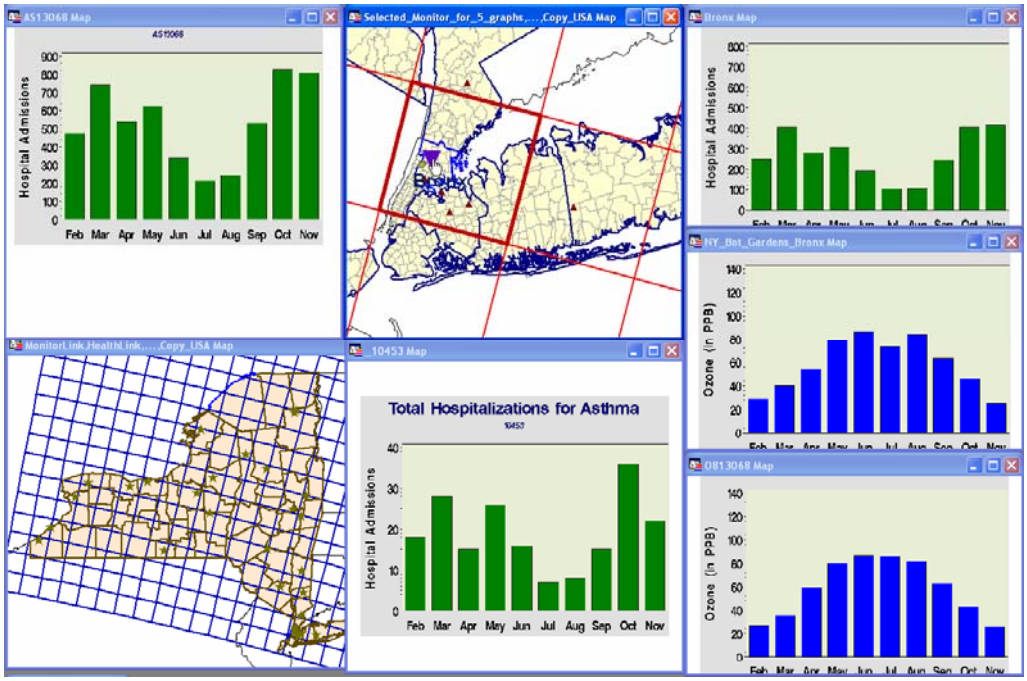

Figure 7: SAS generated graphical output for air pollution level and hospital admissions in geographical area by time.

Final output in a MapInfo layout window is as shown in figure 8.

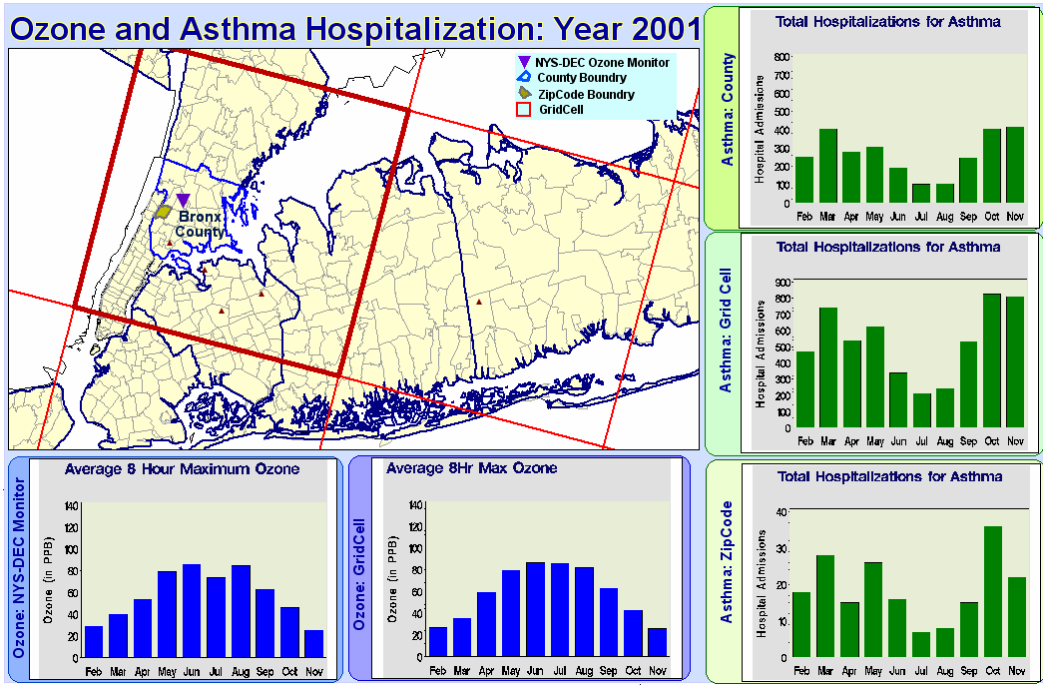

Figure 8: Final layout window showing location map and SAS output.

CONCLUSION:

This presentation introduces a process by which epidemiological data can be visualized temporally and spatially using SAS® ODS and Graph functions in combination with GIS applications. The interactive programming of software requires a fair amount of initial planning and programming. Here we provide a preliminary step for a beginner/intermediate level SAS programmer to explore the benefits of the combined use of SAS and additional GIS software. This process enables users to spot trends and make effective visual comparisons of ambient air pollution and health outcomes among different geographical areas and time periods. Such observations can be used by public health officials to develop and drive public health initiatives and interventions. While the techniques used in the above examples are basic, they provide the foundation for developing more complex and query specific graphs and charts.

ACKNOWLEDGEMENTS:

This work was supported in part by US Department of Health and Human Services, Centers for Disease Control and Prevention grant U50/CCU422440-03 for Environmental Public Health Tracking.

TRADEMARK:

CONTACT INFORMATION:

Your comments and questions are valued and welcome. Please contact the author at:

New York State Department of Health
547 River Street, Room 200
Troy, NY 12180
Sxk10@health.state.ny.us