Distributed GIS for Monitoring and Modeling Urban Air Quality *

by

Chen-Hsiang Yeang, Joseph Ferreira, Jr., and Ayman Ismail
Massachusetts Institute of Technology

Abstract:

The progress of technology has made the measurement of air quality and the simulation of complex air pollution models both feasible and cost-effective. However, there is a long way to go in terms of facilitating widespread access to the data and models, and linking the monitoring of trace gases with specific urban activities and land use that might be controllable. As part of a NASA-funded project, we are working with scientists and engineers to design and test a distributed GIS infrastructure for studying such "urban respiration" phenomena. Measurements of trace gases within a metropolitan area (from mobile and fixed instruments) are geo-referenced, time-stamped, and stored in a relational database server (Oracle). GIS services (using ArcInfo and ArcView) are connected to the database so that subsets of the trace gas measurements can be extracted and converted on-the-fly into GIS data layers. These subsets (by location, date, and time-of-day) can be displayed and cross-referenced with other layers such as weather conditions, land use and cover, topography, hydrography, demography, and congestion levels of road networks. A web-based interface (using ArcView Internet Map Server) allows research team members at different locations to query, visualize, and process the cross-referenced data layers in order to generate surface level estimates of initial conditions for use in the air quality models.

Keywords: distributed GIS, Web GIS, air quality, urban modeling, database management

* Published in the Proceedings of the 6th International Conference in Urban Planning and Urban Management, September, 1999, Venice, Italy; and, subsequently, in Italian, in the Journal, URBANISTICA, n.114, October, 2000.

Introduction and overview

Recently anthropogenic impacts on the atmosphere at a global scale have raised considerable attention from scientists, policy makers and the general public. It is widely conceived that the increase of various chemicals released from human activities into the atmosphere is the major cause of global warming (carbon dioxide and methane) (Raynaud et al, 1993) and destruction of the ozone layer (nitrous oxides and CFCs) (Raynaud et al, 1993). At regional and local scales, studies also reveal strong linkages among the density and characteristics of human activities, air quality, and various indicators of public health. Since urban areas have higher population density and more intensive air-polluting activities (such as vehicle traffic, industries, commercial and domestic activities), they attract more attention than other places. Finding these relations between air quality and urban activities (so called urban respiration) has become both a scientific subject and public issue.

Improving our understanding of this complicated phenomenon relies mainly on three aspects: (a) improved collection technology for measuring spatial and temporal fluctuations of trace gases at finer grains; (b) sophisticated models which can better explain and predict air quality based on physical principles and available data; and (c) an information infrastructure which streamlines the management, interpretation, and presentation of the data and analyses in order to allow better and broader input, collaboration, and debate of the models and their policy implications. The purpose of this paper is to discuss how the third aspect – information management, distribution and presentation – can be used to help understanding the urban respiration phenomenon. We start by introducing the background of the urban respiration project, then we raise the issues and arguments about data access and processing in terms of a perspective on system architecture. In the third section, we describe a prototype architecture for GIS-based modeling of urban respiration. The fourth and fifth sections describe and illustrate the basic data layers that can be readily obtained and cross-referenced at scales and levels of detail that have become increasingly standardized in the U.S. In the sixth and seventh sections, we explain how the prototype architecture stores and retrieves trace gas monitoring data and we illustrate its use in developing maps and baseline modeling parameters by combining the monitoring data with information extracted from the basemaps. Finally, we will discuss the lessons learned from building this prototype system and indicate future directions we propose to develop in next few years.

1. Background of urban respiration project

The distributed GIS work reported in this paper is part of a larger multi-disciplinary effort funded by the United States National Aeronautics and Space Administration (NASA). This larger project addresses a broad set of modeling and measurement issues concerned with urban metabolism and respiration and involves researchers at several institutions as indicated in Table 1. Based on the belief that a comprehensive study about urban respiration phenomenon cannot be achieved without multi-disciplinary efforts, the urban respiration project was launched in 1997. Table 1 lists the participants and their responsibilities in this overall project.

A brief discussion of some of our collaborators' work will help in understanding the issues and motivation for our distributed GIS model. Aerodyne Research has developed a mobile van that can measure the concentration of trace gases (CO_2, CH_4, NO_x, etc.) in real-time at the rate of one measurement every one to six seconds while traveling around the city. The van acquires spatial coordinates and time stamps for the sampling points using Global Positioning System (GPS). MIT's environmental engineering group builds sophisticated models of atmospheric chemistry and fluid dynamics that can translate meteorological conditions and surface level measures of terrain, roughness, land use and trace gas emissions into volumetric simulations of trace gas chemistry and concentration. The authors of this paper are in the Planning Support System group of MIT Urban

Studies and Planning Department. We are especially interested in building distributed spatial information systems that can facilitate the sharing, use, and presentation of ‘urban respiration’ data. A key goal is to improve the linkages between the complex science and engineering models and the standardized geographic data and urban modeling tools that are used by urban planners and policy makers to understand and debate land use planning options and constraints.

Team member	**Responsibility**
NASA	Sponsor.
Aerodyne Research, Inc.	Lead contractor; focused on mobile monitoring equipment in situ trace gas measurement.
M.I.T. Dept. of Urban Studies & Planning,	Spatial information infrastructure and urban land use models.
M.I.T. Chemical Engineering Dept.	Models of atmospheric chemistry and air pollution in and above metropolitan areas
University of Washington	Measurement and modeling of meteorological conditions.
University of New Hampshire	Measurement and modeling of aerosol dynamics.

Table 1: Participants in the overall urban respiration project

The project team is especially interested in studying the urban respiration phenomenon in major metropolitan areas that face the worst air pollution. However, the scale, scope and complexity of urban activities and atmospheric chemistry in such metropolitan areas suggested that we focus our initial efforts on a metropolitan area of more tractable size and complexity. Therefore, we chose to begin with a study of Manchester, New Hampshire – a medium-sized city that was familiar to one of our research teams (UNH) and close to the research groups in the Boston metro area. (MIT and Aerodyne). Later, when we gain experience and confidence with our data collection, calibration, and modeling methods, we plan to shift our focus to Boston, Massachusetts. Boston is a large metropolitan area in Northeast United States that is familiar to the researchers and for which we have a rich assortment of land use planning data. Table 2 compares the Boston and Manchester areas with Los Angeles, California – the largest metropolitan area in the U.S. and one that generally regarded as the U.S. metro area with the most severe air pollution problems. The comparative statistics in Table 2 indicate the differences in scale and ‘complexity’ that motivate us to begin with an area like Manchester and then test what we learn in an area such as Boston.

1990 U.S. Census statistics (for entire metropolitan area)	**Los Angeles, CA**	**Boston, MA**	**Manchester, NH**
Population	15,608,886	5,827,654	147,809
Households	4,900,720	1,547,004	56,571
Area (sq miles)	33,966	6,450	150 (approx.)

Table 2: Comparison of U.S. metropolitan areas.

2. System architectures for accessing and processing geospatial information

2.1 Bringing GIS into the project

There are strong arguments to support bringing GIS into the scope of urban respiration project. Measurement results show the distribution of trace gases strongly depends on space and time. In order to represent the spatial variation of air quality, it is necessary to record the coordinates of sample points, draw the sample points on maps, use cartographic representations to demonstrate the variation of data values (for example, the thematic map of CO_2 concentration based on color coding), and interpolate the sampled data to estimate the values at unsampled areas. GIS becomes a potentially useful tool for addressing the spatial data processing tasks. Moreover, most factors affecting air quality – including traffic conditions, land use, vegetation cover, terrain, and the location of known point sources) - are currently available in geo-referenced form and, often, as explicit GIS data layers. It is often quite useful to display trace gas measurements and other air quality data on top of these basemaps and "context layers" at several stages of the urban modeling process. For example, early mapping of the monitoring data can suggest intuitive explanations about pollution sources, weather conditions and other parameters that help in selecting the level of filtering, space/time data aggregation and 'number crunching' needed to calibrate and run the models of atmospheric chemistry and fluid dynamics. Likewise, current GIS tools can systematize the use of simple spatial models to link the predictions of the scientific models to controllable land use factors. For example, GIS tools can readily generate distance buffers, spatial interpolations, and plume patterns that can go a long way in detecting and visualizing relevant spatial correlations, sensitivity analyses, and diffusion patterns. Nevertheless, a number of complicating factors make the exchange and use of spatially referenced data and tools far less smooth than one might imagine. For example, the sheer volume of data, the dispersed locations of the researchers and planners who analyze them, the space/time complexity of the fluid dynamics and atmospheric chemistry, and the different coordinate systems used to model spatial relations are all complicating factors.

2.2 Distributed architecture for data access and processing

We believe a distributed architecture for the spatial information infrastructure is desirable given the nature of the urban respiration project and current status of information technology. From table 1, it is clear the participants are located in different geographic areas all over the United States. It is impossible to share and access data among participants without some kind of distributed infrastructure. Even if all parties were located within the same metropolitan area, the multi-disciplinary nature of the measurement, modeling, and urban planning activities would require data sharing and coordination among physically separate clusters of research/monitoring labs and planning/analysis groups.

At the start of the project (1997), communication among the project's sub-groups was typical of current science and engineering collaborations. Phone calls and occasional meetings were supplemented by electronic mail (viz., ASCII text messages plus occasional attachments of MS-Word documents or images in GIF or TIFF format). Internet connectivity was available to all parties at speeds ranging from dialup modems to 10 Mbps ethernet links. Hence, the exchange of larger datasets (involving more than one or two megabytes) was relatively infrequent but could reliably depend upon the use of FTP sites. However, no standardized way of exchanging *geo-referenced* information was in place. For example, the mobile measurement team encoded position (using GPS data) in lat/lon, the meteorological folks used lat/lon or a Lambert conformal conical projection (to facilitate use of MM5 weather models), and the urban planning group used New Hampshire State Plane Coordinates (an Albert's projection – in NAD83 feet – that matched the land use and planning datasets supported by New Hampshire agencies and towns). Only the urban planning group had ready access to GIS tools for geo-referencing datasets and converting data among coordinate systems. But all groups had experience with PC-based LANs and most had Unix workstations for use in engineering- and/or GIS applications.

In suggesting a distributed GIS architecture for the project, we wanted to:

- improve the efficiency and effectiveness of spatial data processing for our project groups,
- test the viability of a system design that could easily be used to model other metropolitan areas, and
- select an architecture that could reasonably be replicated by other research and planning teams.

Reliance on Internet connectivity and Web-related protocols was an obvious choice since it built upon existing connections and tracked trends in coordinating intra-metropolitan information systems. However, distributed GIS technologies are new and emerging technologies that are heavily impacted by rapidly changing Internet and Web developments. We wished to avoid both the extensive development of customized software and the use of Web-based 'off-the-shelf' technologies that were likely to change rapidly during the course of the project. Nevertheless, a wide range of thin-client to thin-server options exist among the various client/server strategies that we could employ.

With these considerations in mind, we decided to build upon the existing hardware and networking capabilities and use a small set of connectivity and data conversion tools together with commercially available GIS and RDBMS software. Since data consistency and accessibility are so important, we focused on centralized strategies for storing primary data and decentralized approaches for accessing these data and for triggering the filtering, aggregating, and visualizing that are needed before downloading customized subsets of the data for local analysis. In designing the system, we tried to choose paths that would be consistent with interoperable GIS principles as outlined by the Open GIS Consortium.

While spatial referencing is a critical dimension for all participants in this project, not all of them need full-fledged GIS software. However, they should at least be able to (1) view weather and trace gas monitoring data on top of a standard set of land use, land cover, and demographic basemaps, (2) perform basic cartographic manipulations such as turning on/off specific layers, zooming in and out, and generating thematic maps for key data layers, and (3) filter key datasets interactively for selective viewing against the various basemaps. Since it is neither economical nor practical to install full-fledged GIS software on each participant's computer, we focused on using internet connectivity and Web-type protocols to give local users a minimal set of GIS functionality. The primary mechanisms for doing this is to provide local users with a graphical interface that controls the selection, geoprocessing, and mapping of datasets on a small number of GIS and DBMS servers.

3. Prototype architecture

Figure 1a provides a conceptual diagram of the system architecture for the distributed GIS approach that we have implemented. Since the Internet and World Wide Web are so pervasive, we use TCP/IP and Web browser protocols for connectivity. On the client side, individual users access the data through the Internet without requiring proprietary software. A Web browser can function as the Graphical User Interface (GUI) at client side. Users can use this interface to send requests about the characteristics of the GIS data they want to process or view. For example, they can identify the layers to be turned on, the spatial range of the map display, and the specific subsets of certain data layers that they need. The browser interface can also be used to display the resulting maps or GIS data files and/or to bundle subsets of data for downloading.

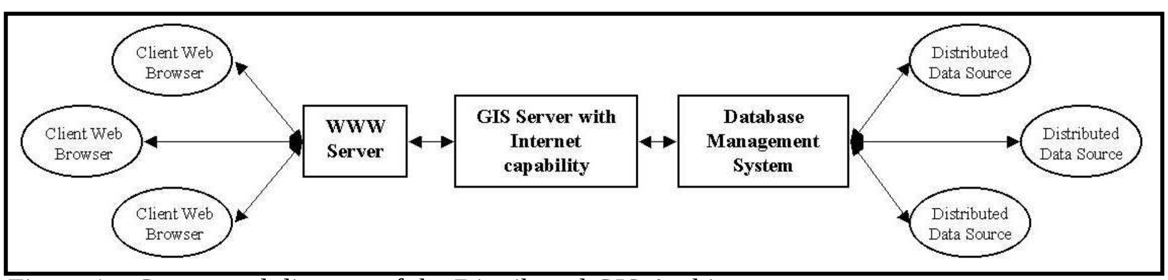


Figure 1a: Conceptual diagram of the Distributed GIS Architecture

At the server side, a Web server communicates with the client-side Web browsers. Unlike a conventional Web server, this one has an additional function of bridging to a GIS server. It translates commands from the Web browsers into the function calls identified by the GIS software. Going in the other direction, it also converts GIS maps and datasets into the formats which can be displayed on the Web.

A GIS server sits behind the Web server. It functions as conventional GIS software except with distributed computing capability. In other words, it will be able to handle multiple requests from different clients. Data consistency among multiple users is one of the major problems for distributed data sharing. If two users modify the same data simultaneously, the race conditions may lead to unexpected changes in the data. A fully-fledged database system has various protection mechanisms to avoid these conflicts (such as priority queuing and transaction-based writeback). Nevertheless, due to the nature of this project, a concrete database system for complicated data sharing purposes (such as a distributed corporate database) is not required. This is because it is practical to limit client-side processing so that the 'core' maps and datasets that are shared are viewed as 'read-only'. Most users are satisfied with this unidirectional data access and recognize the simplified data management that results. They can generate subsets of data and map the results, and the can downloading maps and filtered datasets. But they cannot use the interface to upload new data or otherwise alter the 'core' datasets

A relational database management system (RDBMS) sits behind the GIS server and stores all the tabular data from the measurement and monitoring of weather and trace gases. The major advantage of using a DBMS rather than the GIS per se to store these data is the capability to extract (and aggregate, filter, or otherwise process) specific subsets from the data in response to users' requests. For example, a single evening of mobile van readings throughout a metropolitan area might acquire tens or hundreds of thousands of observations. A researcher may wish to extract and plot a small subset of these data after, say, a smoothing operation that aggregates them in time and space to a hundred yards and/or ten seconds. These requests are readily handled as queries (and stored procedures) using the industry-standard Structured Query Language (SQL). Access to the RDBMS data is directly available to client-side users via normal distributed DBMS protocols (such as ODBC). But the system also allows the RDBMS datasets to be accessed from the GIS server in response to user-defined queries entered by the user through their GIS interface.

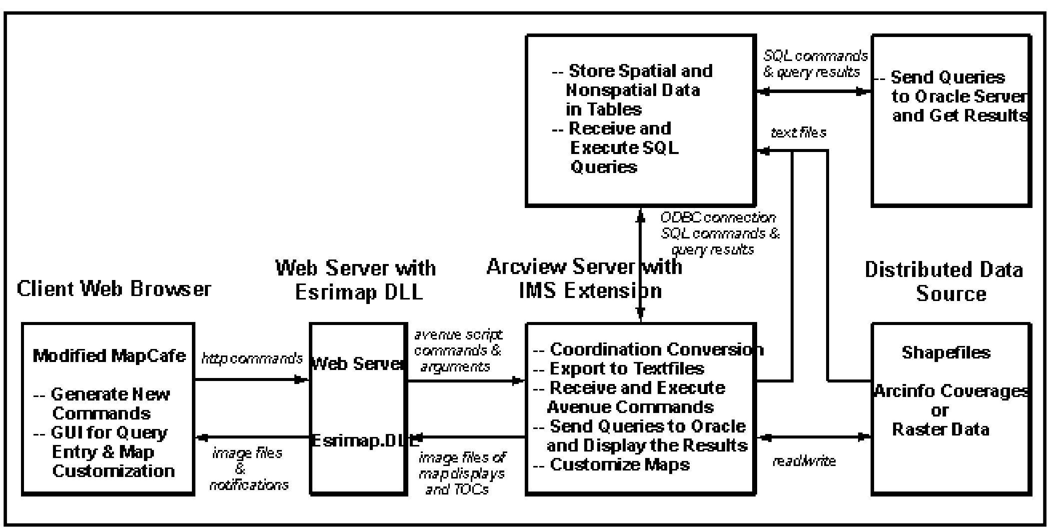


Figure 1b: Current Implementation of the Distributed GIS Architecture

Figure 1b explains the current implementation of the distributed GIS framework that was diagramed conceptually in Figure 1a. For the GIS server, we chose ArcView with the Internet Map Server (IMS) extensions (Environmental System Research Institute, ESRI). There were several reasons for the choice. The research team had substantial experience using ESRI software (both ArcView and ArcInfo) and many land use and demographic datasets of interest to the project were readily available in ArcView readable formats. In addition, ArcView has both a user-friendly interface, data formats compatible with ArcInfo, and significant market share as a standalone desktop mapping package. The former advantage makes it easier to build the graphical user interface for project participants whose expertise is not GIS. The second advantage allows us perform sophisticated GIS operations using ArcInfo and then display the resulting map layer in Arcview without the need for additional data conversion.

However, ArcView plus its IMS extension cannot satisfy all our data access and management needs. If we stored the monitoring and measurement data as GIS layers (in this case, as ArcView shape files), end users would have little freedom to select and display specific subsets of the entire data set. Excluding the display of unwanted subsets of the shapefile data would be too complicated a task that is not possible with the reduced GIS functionality that is available via the ArcView-IMS graphical interface. Therefore, we decided to use a database management system (DBMS) to store the air quality data, and allow users to access these data through Arcview IMS. We chose Oracle since it supports industry-standard SQL and distributed DBMS protocols such as ODBC, it is compatible with ArcView, and it is widely used to store enterprise-level datasets for environmental planning and metropolitan management. To illustrate the need for a DBMS, suppose a user wants to display the measurement samples obtained during 2pm to 4pm among all data collection trips. The user can input his or her queries using the graphical user interface at the client side. The server-side of the ArcView-IMS application passes the request (as a standard SQL query) to the DBMS, which then returns the results as an ArcView table. ArcView then converts the data table into a GIS point data layer and display it on client's map.

The client side of the implemented system is a generic web browser which runs a Java applet called MapCafe that comes with the IMS extensions to ArcView. MapCafe sends commands generated by the user interaction (in http format) to the web server, and it receives and displays image files, tables, and annotations generated by the GIS Server and packaged (in http format) for client-side viewing. The look of the GUI at client side is similar to ArcView, except more limited in its toolbars and interactivity. (Figure 5 will provide an example later in this paper). Basically, the MapCafe acts as a thin-client application providing the user with a 'keyboard extension' through the Web so that the user can run (some of) the ArcView functionality that they would have if they were sitting directly at the GIS server. Users can turn specific data layers on/off, zoom in/out, pan, get information about specific data elements, and print. The added functionality of executing (and then mapping) logical DBMS queries of air quality data sets the results required customized 'Avenue' scripts and JAVA programming.

At the Web server side, ESRI provides a CGI program (a dynamically linked library, esrimap.dll, in the case of a PC server) which converts the incoming http commands into the GIS commands (Avenue scripts) required by the GIS server (Arcview). This 'relay' code on the Web server also takes the images and texts produced by ArcView and packages them for client-side display.

The ArcView application runs on a GIS server that can be – but need not be – the same machine as the Web server (or the DBMS server). In our case we use either a PC or a Unix workstation as the GIS server. The GIS server listens for avenue script commands from the relay program, performs the requested manipulations as if the user were requesting them at the keyboard, and sends the updated display back to the relay program. For example, suppose the client-side user zooms in by dragging a rectangle across the map showing in their browser window. The relay program will send to the GIS server the commands that reset the zoom level of the map (the command will include the four vertices of the zoom rectangle expressed in viewing coordinates). ArcView performs the zoom, redraws the map on its local map display, and then sends the map – as a GIF image – back through the Web server to the client. In our scheme, ArcView also functions as a relay to database server. Through the Open Database Connection (ODBC), it can send SQL commands to one (or more) Oracle database servers and receives the query results. It then converts the resulting table into a temporary shape file, opens this shape file as a new layer of the map, and sends the map to the client as explained above.

By using ArcView-IMS, we have chosen a *thin-client* solution for our distributed GIS. No GIS 'smarts' exist on the client-side. The GIS server packages all maps as graphic images and passes them to the client-side browser. All spatial data processing – e.g., buffering, point-in-polygon computations, and the like must be done on the server side and the map 'themes' (i.e., data layers) and associated data the ArcView displays must be explicitly loaded and prepared for viewing by the manager of the GIS server. In effect, the Web is used as a long (and somewhat cumbersome) keyboard extension to allow the client to act as if they were sitting at the GIS server examining prepared ArcView 'projects'. An alternative (*thick-client/thin-server*) approach would be to use the servers only as a data repository and move the GIS processing to the client-side. A simple example would be to run ArcView on each client and use a network file server to share the raw data over the net.

Intermediate solutions are also offered by ESRI and other vendors (e.g., Intergraph, Autodesk, and MapInfo). For example, the GIS server could supply snippets of vector data (rather than finished maps in raster image format) and these snippets could be then be overlayed, cross-referenced, and processed or displayed locally. Likewise, our use of the DBMS server (Oracle) is limited to storing only the trace gas measurement data. Relational database managers with spatial data extensions (such as SDE and SDO) are becoming available that could hold all the basemaps and tabular data (as well as trace gas data) and then make customized snippets available to the user. However, at this stage, we wanted to avoid extensive coding of client applications and experiment with the simplest possible schemes for enabling the project team to access and map shared data. After describing our experiences with the initial implementation, we return to these system design issues. In the meantime, note that the current system does provide some alternative mechanisms for accessing the data. For example, users who do not need to generate maps, they can also access the DBMS server as an ODBC client to retrieve data in tabular format. They can also generate and save customized 'views' of the data that can easily be included in the set of maps and queries available through the current implementation of the distributed GIS system.

4. Spatial Data Layers and Basemaps

We have collected a series of data layers from a variety of sources (mostly online) that are representative of datasets describing land use, land cover, terrain, demographics, meteorology, and known point-sources of air pollution. Although they are not an exhaustive list of determining factors for air pollution, we believe they constitute a meaningful and reasonably comprehensive set of data that are relevant to the generation and dissemination of air pollutants. Moreover, they are data that can be assembled in a reasonably standardized fashion for most U.S. metropolitan areas in order to help us understand the urban respiration phenomenon

4.1 Land use/land coverage

The land use/land cover (LULC) data files from US Geological Survey (USGS) describe the vegetation, water, natural surface, and cultural features on the land surface. Original data sources include high-altitude aerial photographs and earlier land use maps and field surveys. They are stored in a Geographic Information Retrieval Analysis System (GIRAS) format and are often available online through State-supported Web sites that archive environmental management data. The scale of the LULC maps is 1:250,000.

4.2 EPA monitoring sites

The Aerometric Information Retrieval System (AIRS) and the AIRS Facility Subsystem (AIRS/AFS) are online services of the EPA's Envirofact database – a large database of environmental data maintained by U.S. Environmental Protection Agency. It comprises the identification information, spatial coordinates, and emission inventory of EPA-monitored sites. The chemicals monitored by EPA include CO, NO_2, particulate matters, lead, SO_2, and volatile organic compounds. The database is open to the public and can be accessed through the World Wide Web (http://www.epa.gov/enviro). Since the Envirofact data warehouse is stored in Oracle and supports SQL queries from the public using ODBC protocols, our distributed GIS architecture can include online queries of emission information for sites in the Envirofact database.

4.3 Terrain data

Terrain data for Manchester, NH, comes from the New Hampshire Resource Net, a GIS warehouse supported by the state of of New Hampshire and maintained at UNH (http://nhresnet.sr.unh.edu/). The archived data are stored in a 7.5-minute digital elevation model (DEM) format and represent elevation estimates for use with map scales of 1:24,000 or 1:25,000. The elevation estimates are 30 meters apart for an x-y grid that is based on a UTM (Universal Transverse Mercator) projection method and the 1927 North American Datum (NAD 1927).

4.4 Surface water hydrography

The data of surface water hydrography also comes from the New Hampshire Resource Net. It contains the vector representations of the boundaries and/or centerlines of various surface water bodies (lakes, ponds, rivers, wetlands, reservoirs, etc.) The scale is also 1:24,000/1:25,000. It uses New Hampshire State Plane Feet projection and 1983 North American Datum (NAD 1983).

4.5 Road networks and railroads

The data about road and railroad networks also come from the same New Hampshire Resource Net and have the same projection, horizontal datum, scale, and file format as the hydrography data.

4.6 Demographics

Demographic data about population characteristics come from 1990 U.S. Census Bureau datasets (STF3a). They are georeferenced by place of residence down to the blockgroup level and are linked to vector-based maps of census tracts, block groups, and streets using the Census Bureau's TIGER files (and other street centerline datasets supplied by third-parties and derived from and/or compatible with TIGER data).

4.7 Creating Base Maps

By using the various data layers described above, we were able to create and overlay a rich collection of data relevant to the modeling and analysis of urban respiration in and around Manchester, NH. To facilitate viewing and analysis that is consistent with NH supported GIS data, all the georeferenced data were converted to New Hampshire State Plane coordinates (NAD-83, feet). While this conversion process is well understood, many GIS packages do not yet support robust enough methods of on-the-fly coordinate projection. Among the few data sources that we assembled for Manchester, NH, we encountered four different coordinate systems, 2 different NAD reference sets and both ‘meters’ and ‘feet’ accounting. Since there are good reasons why one ‘size’ does **not** fit all purposes, we regard this phenomenon as evidence that robust, on-the-fly methods for coordinate conversion are a key element of building useful distributed GIS systems for supporting multi-disciplinary modeling and analysis.

Figure 2 shows a map of the United States with southern New Hampshire circled and Manchester, NH, visible in the inset. The inset covers a 9x9 mile^2 area in and around Manchester and shows municipal boundaries, railroads, roads, water bodies and EPA monitoring sites (http://ortho.mit.edu/nasa/basemap2.gif). The 4x4 mile urban core is almost surrounded by limited access circumferential highways. A major route (I-293) cuts through the city along the Piscataquog River and another major route (I-93) runs in parallel with I-293 along the east side of the city. Most of the EPA-monitored emission sources concentrate at the east bank of the river. This is also where most urban activities are concentrated.

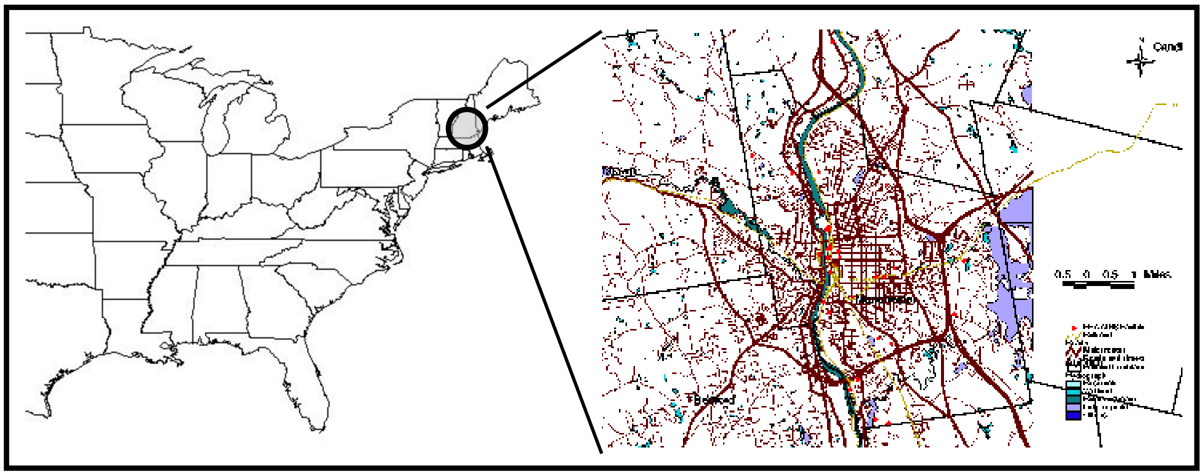


Figure 2: The 48 Coterminous United States with Manchester, NH, shown in the inset.

The land use/land cover data (not shown here but available at: http://ortho.mit.edu/nasa/lumap2.gif) indicate that most commercial and industrial areas are concentrated along the banks of the river and the south end. These land uses are surrounded by residential areas and vegetation cover comprises a large proportion of land outside the urban core.

Figure 3 shows a ‘screen shot’ of ArcView with the Manchester map from Figure 2 visible in the map window and the visible map layers (ArcView ‘themes’) checked in the legend along the left side of the map. Figure 4 shows the terrain in and around Manchester using a lattice mesh developed from the DEM data. . The North edge of the terrain is along the upper left border and the topography descends from the northwest toward the southeast corner. The terrain model highlights the river valley running North/South through town and the hilly terrain on the Northwest side of town.

Figure 3: ArcView screen-shot of Manchester.

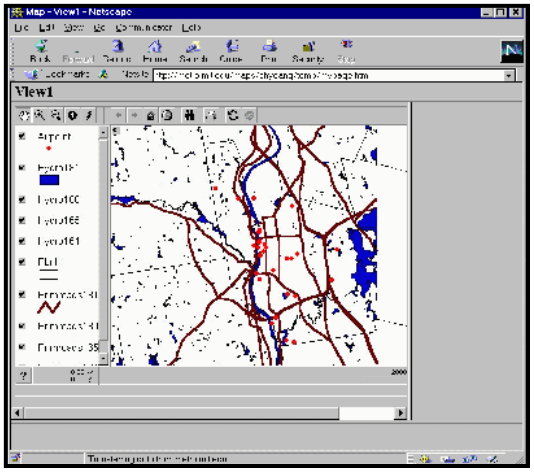
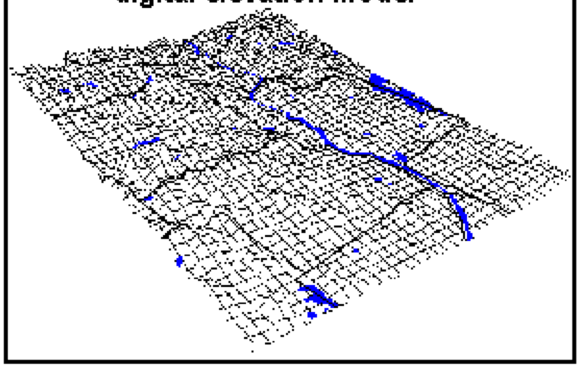


Figure 4. The wireframe diagram of digital elevation model



5. Visualizing Mobile Measurement of Trace Gas Concentration

Figure 5 shows the CO_2 concentration data collected on November 10th and 11th 1997. The background shading represents land use data from the USGS-LULC coverage. The mobile measurements are point data that form the linear paths that crisscross the downtown area and evident in the map foreground. Please note that the printed grayscale figure is far less readable than an on-screen map in color. The narrow and lighter portions of the paths represent CO_2 readings at or near the background level. The wider and brighter portions of the paths represent successively higher readings. The trip on November 10th was conducted during the late evening (from 8pm to 1am), whereas the trip on November 11th was during the late afternoon and early evening (4pm to 9pm). The 10 hours of mobile readings produced 36,000 readings (one per second) for each of the two gases that were monitored (CO_2 and CH_4). The time stamp, GPS location, and trace gas readings for the data are stored in Oracle and converted into mappable data using ArcView.

The contrast of CO_2 concentration between the two time periods shown in Figure 5 is striking. The November 11th data contains more high concentration spots than the November 10th data. Except at the southern tip of the city (the intersection between routes 3 and 293), all other high concentration points are not overlapping. This reveals the possibility that either (1) most CO_2 sources are non-stationary and time-varying, which strongly point to motor vehicles; and/or (2) the measurement data are very sensitive to local traffic conditions, especially if the van is measuring vehicle exhaust from cars close in front of it.

The land use classification data from USGS LULC are too coarse (both spatially and in terms of land use categories) to demarcate precisely those areas that have higher CO_2 concentrations (for reasons other than car emissions). Although almost all the high CO_2 concentration spots are located within the industrial, commercial or residential areas, there are many low concentration spots which are located within these areas, too. Terrain and meteorological conditions (e.g. being upwind or downwind of emission sources) are likely to have significant impacts even if little atmospheric chemistry is occurring (at the surface).

These observations indicate that the data processing and 'reverse engineering' needed to link trace gas measurements to emission sources requires considerable data filtering, modeling and analysis.

The ArcView-generated maps in Figure 5 are displayed (via MapCafe) in a Web browser using ArcView-IMS. Hence any user on one of the research project teams can use this interface to explore and map the mobile measurement data. But the ArcView-IMS user is limited to simple zoom, pan, and query options for exploring a limited number of data layers. Complex filtering and querying of the mobile measurements is either not possible or too awkward and time consuming with the standard ArcView-IMS tools.

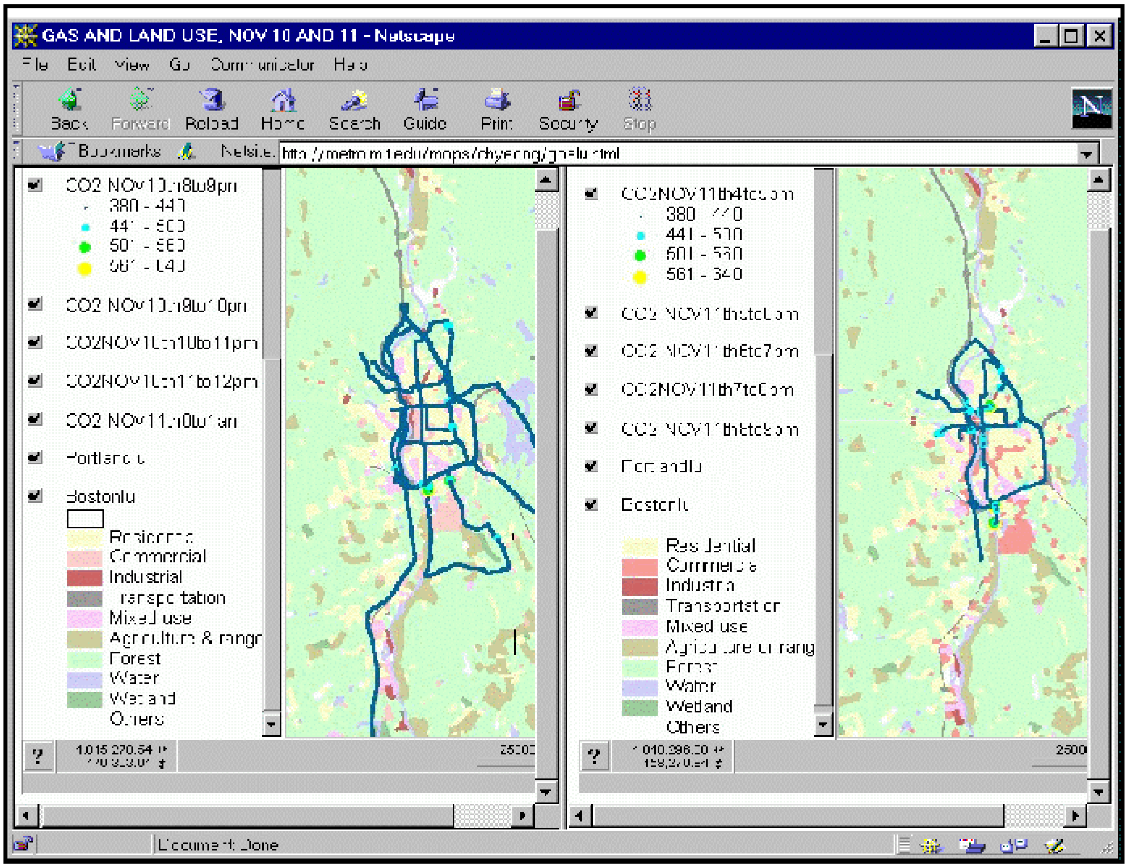


Figure 5: MapCafe Screen-Shot comparing CO_2 Measurements at Different Times of Day.

To address some of these issues, we added a customized query capability to ArcView-IMS so that end users can access the Oracle database directly in order to query and map selected subsets of the monitoring data. Figure 6 shows the graphical user interface for these queries of the Oracle database. The user must specify a username in order to distinguish (and hide) their customized ArcView 'themes' from those that other users create. Users also need to specify the name of the table to be queried and the text label they wish to use when mapping their customized datasets. These text labels are color-coded for added clarity.

Users can use this dialogue box to select the subsets of trace gas readings that they wish to extract and map. They can select by type of gas, time period (including day and time of day), and measurement levels. The five selection boxes and text fields below the check boxes allow users to specify a range of conditions. In this example, only one condition is specified: the CO_2 level must be between 550 ppm and 600 ppm. Query results are the intersection of all specified conditions. After these fields are filled, users can either click the SUBMIT button to submit the query command to the Arcview server, or click the QUIT command to close the GUI window. Hitting the 'quit' button after entering only one's user name will load all previously generated user-defined themes into the viewing window without generating a new theme. If a new query is specified, ArcView passes the query (in SQL format) to Oracle and automatically generates and maps a new shapefile comprising the selected datapoints.

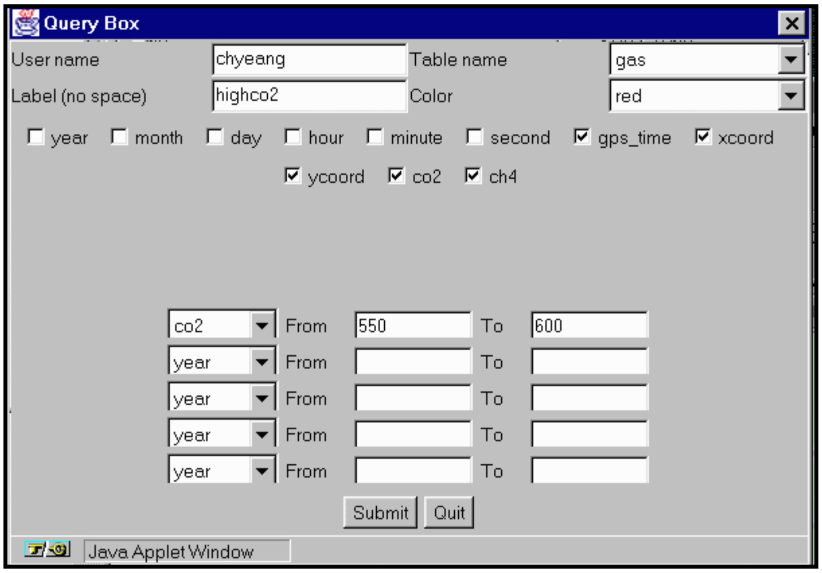


Figure 6: Query Box for Customized Querying and Mapping of Trace Gas Measurements

The look of the modified system is very similar to the original IMS user interface (Figure 5) except that a new button is added to the topmost button list. New layers of user-selected CO_2 and CH_4 observations show up as additional themes in the viewing legend. Since these layers are created from individual user queries, different users can view and map different subsets of the entire trace gas measurement database.

6. Geo-Processing Examples

The previous example (Figures 5 and 6) illustrated the use of distributed GIS tools to facilitate the visualization, filtering and aggregation of trace gas observations. The same approach can be used to handle a range of mobile (and fixed) measurement data (for meteorological data and a suite of trace gases). But such processing and mapping is only the first step in the modeling and analysis process. The broader goals are (a) to use the measurement data to calibrate the surface conditions for running (and improving) the volumetric models of atmospheric chemistry and fluid dynamics, and (eventually) (b) to 'reverse-engineer' the model predictions in order to identify the locations and types of emissions that are most influential in producing adverse air pollution conditions.

The distributed GIS architecture can also be helpful in supporting these goals. We illustrate one such use involving the estimation of certain baseline surface conditions for calibrating the model. Vehicular traffic is a key source of trace gases (such as CO_2). The location and level of these emissions is, of course, dependent upon road location, traffic congestion, cold-start effects, vehicular emission controls, and the like. The GIS basemaps described earlier provide a rich source of data that can be used with the GIS tools to build useful models of the spatial pattern of vehicle emissions. For example, Figure 7 shows the road network in and around Manchester in gray and solid lines. The buffers surrounding the roads are shaded lighter and lighter to the extent that they are closer and closer to more (or more major) roads. (A simple inverse distance model is used with weightings based on road class and number of lanes). Further adjustment could be done to reflect time of day traffic congestion and/or estimated differences in vehicle mix and emission levels depending upon the proximity of the roads to residential neighborhoods with different demographic profiles.

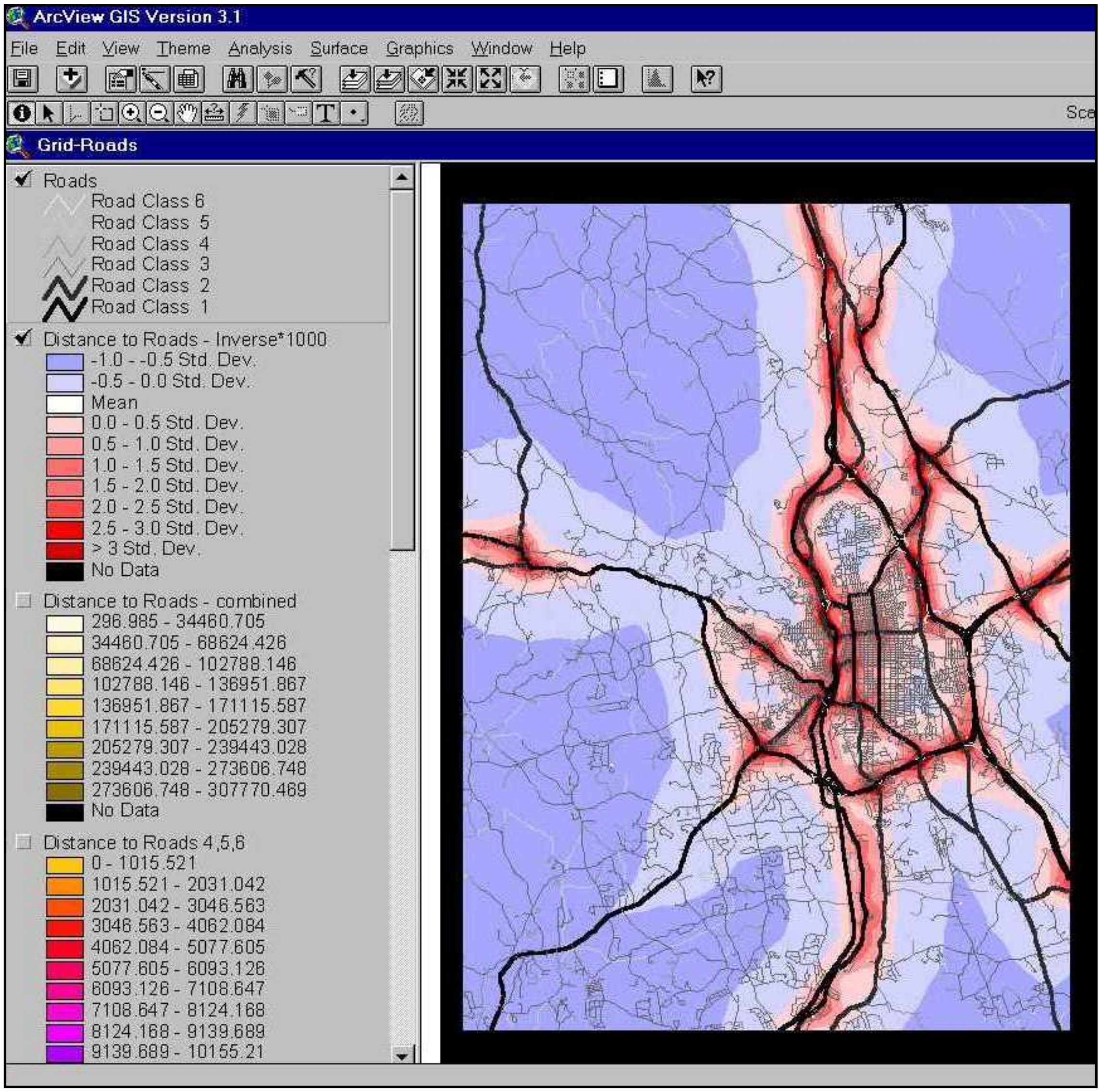


Figure 7: Proximity-to-Road Model for Estimating the Spatial Distribution of Vehicle Emissions

Such models can be used to create contour maps for the expected (surface) levels of trace gas concentration due to emissions from cars. And, trace gas measurements from, say, morning rush hour periods could be used to calibrate the parameters of these surface-level emission models. The calibrated models could then be used in estimating the road contribution to trace-gas emissions throughout the metropolitan area. They could also be used to extrapolate estimates of aggregate emission levels (from vehicles) that are generated within each grid cell across the metropolitan area. Surface level 'initial conditions' such as these are needed to calibrate and seed the air pollution models. Similar analyses and spatial data processing could be done with land use and terrain data (e.g., to estimate surface 'roughness'), and with demographic data.

GIS tools such as ArcView have the buffering, rasterization, and map algebra capabilities needed to make these estimation, spatial interpolation, and spatial aggregation steps reasonably automatic.

A limited amount of coding – similar to the oracle queries discussed earlier – can add such functionality to the distributed GIS capabilities of our system. We are currently experimenting with these approaches.

7. Conclusions and Future Work

This paper describes the first round of prototyping and using a distributed GIS system for modeling, analyzing, and monitoring urban respiration. A thin-client approach was used to distribute limited access to spatial data sets and monitoring data using Internet & Web-based protocols. Off-the shelf GIS and RDMBS tools were used to provide Internet-accessible spatial data processing and querying services with a minimal amount of customized programming. Initial experience with these tools indicates that:

- It is relatively easy to assemble and standardize key datasets of spatially referenced data that can serve as ‘basemaps’ for visualizing and analyzing trace gas monitoring data.
- It is relatively easy to store and cross-reference the GPS-referenced trace gas measurements so that they can be overlaid on the basemap layers.
- It is useful to provide (via ArcView-IMS) a minimal level of desktop mapping capability, with some consistency and user flexibility, to the various research teams.
- It is still difficult (within the limitations of a thin-client approach like ArcView-IMS) to provide sufficient flexibility and analytic capability to avoid the need for ftp exchange of raw datasets among the research teams.
- Standardized, industrial strength RDBMS services are needed to store manage and query the measurement and monitoring data with sufficient flexibility and power.
- The performance issues, reliability, and Java code requirements of a tool like ArcView-IMS are sufficiently complex and non-standard to warrant continued reliance on simpler map distribution strategies (such as static web pages and PDF-formatted maps) for some project purposes.
- More complex (and customizable) strategies are warranted for supporting (a) the filtering, interpolation, aggregation, and visualization of the meteorological and trace gas measurements [Figures 5 and 6], and (b) the spatial data processing needed to estimate surface level emissions, roughness, and other air pollution model parameters [Figure 7].

Currently, our distributed GIS work involves:

- Expanding the suite of analysis and visualization services (such as in Figures 5-7) to allow more sophisticated queries of the measurement data, to incorporate demographic analysis, and to support user-controlled rasterization of interpolated data in order to generate grid cell estimates of surface roughness and emission levels for use in the air pollution models.
- Including meteorological data in the RDBMS and in the spatial data analysis and surface level emission modeling.
- Exploring alternative Web-GIS components and services that offer improved levels of interoperability and efficiency.
- Experimenting with different approaches to facilitating user-downloads of selected maps and analyses.
- Experimenting with JAVA-based process management tools that can help visualize, record, and replay the many querying and processing steps involved in exploring and aggregating the datasets.

Bibliography

Jones, Glyn. Planning and the Reduction of Transport Emissions. The Planner. July 1993.

Lyons, T. J., J. R. Kenworthy and P. W. G. Newman. Urban Structure and Air Pollution. Atmospheric Environment. Vol. 24B No. 1 1990.

McRae, Gregory J., Willian R. Goodin and John H. Seinfeld. Development of a Second-Generation Mathematical Model for Urban Air Pollution -- II. Model Formulation. Atmospheric Environment. Vol. 16 No. 4 1982.

McRae, Gregory J. and John H. Seinfeld. Development of a Second-Generation Mathematical Model for Urban Air Pollution -- II-Evaluation of Model Performance. Atmospheric Environment. Vol. 17 No. 3 1983.

Raynaud, D., J. Jousel, J. M. Barnola, J. Chappellaz, R. J. Delmas and C. Lorius. The Ice Record of Greenhouse Gases. Science. Vol. 259 12 February 1993.