A DBMS Architecture for Global Change Research*

Nabil I. Hachem
Michael A. Gennert
Matthew O. Ward
Computer Science Department
Worcester Polytechnic Institute
Worcester, MA 01609

Abstract

The goal of this research is the design and development of an integrated system for the management of very large scientific databases, cartographic/geographic information processing, and exploratory scientific data analysis for global change research.

The system will represent both spatial and temporal knowledge about natural and man-made entities on the earth's surface, following an object-oriented paradigm. A user will be able to derive, modify, and apply, procedures to perform operations on the data, including comparison, derivation, prediction, validation, and visualization. This work represents an effort to extend the database technology with an intrinsic class of operators, which is extensible and responds to the growing needs of scientific research.

Of significance is the integration of many diverse forms of data into the database, including cartography, geography, hydrography, hypsography, images, and urban planning data. Equally important is the maintenance of metadata, that is, data about the data, such as coordinate transformation parameters, map scales, and audit trails of previous processing operations.

This project will impact the fields of geographical information systems and global change research as well as the database community. It will provide an integrated database management testbed for scientific research, and a testbed for the development of analysis tools to understand and predict global change.

I. Introduction

Conventional database management systems (DBMS) are, in general, not suitable for dealing with the requirements put forth by the scientific community[1]. The diversity and size of data coupled with complicated manipulation and analysis impose quite different needs from the business management environment in which most traditional database techniques are applied. This paper deals with the design of a data management and analysis system better capable of meeting the needs of the scientific community. The key components include the provision for spatial and temporal data objects and the integration of analysis operators, both of which are extensible by the user. The object-oriented paradigm is used in conjunction with deductive database methods to provide a powerful environment for the management and exploration of complex data sets.

*This work is funded by the National Science Foundation under Contract IRI-9116988.

The particular problem domain is cartographic/geographic global change research, an increasingly important area of concern. This domain poses challenges in the use of very large databases, combined spatial and temporal data sets, data consistency across multiple scales, and integration of geographical information systems (GIS).

As man's influence on the global environment becomes increasingly substantial, there is a growing concern that the ramifications of this influence must be carefully studied[2, 3]. Global warming, the greenhouse effect, the ozone layer, acid rain, and tropical deforestation are some of the issues requiring detailed analysis. In order to better understand how all of the factors determining the environment interact, many diverse types of information must be gathered and processed. By studying the history of change in particular areas, researchers are better able to form a model of causes and effects in similar regions. Sometimes change is determined primarily by one or two factors: average monthly precipitation, population shifts, and so on. Other times the relationships are not so readily apparent, and may include the correlation of climate, hydrology, ecology, biogeochemical systems, solid earth processes, and human interactions[4]. Studies which attempt to correlate a subset of these factors include the analysis of vegetation change after a forest fire[5], changes in land use in Nepal based on deforestation and population increase[6], and estimating the effects of global warming on natural vegetation patterns[7].

Data from a wide range of sources can be brought to bear in the study of global change[4]. Spatial information can be in the form of point locations, lines, regions, grids, and rasters. The list below summarizes the key sources.

Cartographic: Digital Line Graphs (DLG), Digital Elevation Maps (DEM), Land Use and Land Cover (LULC), soils, spatial reference layers for agroclimatic zones, ecological zones, averages for rainfall, solar radiation, and surface temperature, and water balance components (runoff, soil moisture).

Satellite: vegetation indices and meteorological information (rainfall, temperature, solar radiation), acquired at many different time periods.

Point data: surface weather and hydrologic stations, soil measures, solar radiation, etc., which can be acquired instantaneously or periodically.

Gridded data: output of numeric models of ground-water flow, weather forecasting, atmospheric circulation, or from analysis of meteorological data.

As the data can appear in diverse formats, conversion between formats is a necessary component of analysis. In addition, data are available at a variety of spatial scales, thus requiring registration to a common reference point and scale. Much of the data are necessarily temporal in nature as well, since change analysis requires at least two instances of a data set separated in time. Temporal registration (adjustments for seasonal changes) may also be required. Finally, non-spatial information, such as economic data, agricultural statistics, and farm management practices, may be useful to the analysis.

A. Project Goals

The long-term goal of this research effort is to develop a spatio-temporal, object-oriented database management system (STDBMS) suitable for the expanding needs of the global change research community. The key features of the system will be:

1. *The provision of extensible data and operator objects to allow arbitrary aggregation of data objects and interactive development of new operators.* The knowledge of

the global change environment is changing on a daily basis, so it cannot be known in advance the types of operations to be needed, nor the type of information to be utilized.

2. *Integrated analysis and data management.* Systems which segregate data management from data analysis force scientists to work in a "batch mode," first extracting the data they believe will be needed, then designing analysis programs for its interpretation. This mode of operation restricts the productivity of research.

3. *Spatial and temporal analysis.* Most geographical information systems (GIS) developed to date concentrate on spatial analysis. Operations specific to the temporal nature of the data (time-series analysis, temporal registration) need to be developed as well. The spatial and temporal dimensions must be unified in a common framework.

4. *Automatic auditing of processing of the data (Metaobjects).* Data will pass through many stages of processing: registration, filtering, feature extraction, time-series analysis, and so on. Records must be kept of the source of the original data as well as all processing steps they have undergone.

5. *Network access.* Databases for global change analysis will always be distributed, due to their size and the diversity of research focus. It is imperative that a database system be aware of the existence of other databases and be capable of handling queries which involve network access. This requires a database server to be resident on each machine containing a compatible database (same format or one for which conversion routines exist), with each having a directory of the data sets available at other locations.

6. *Visual queries.* Output from the analysis of earth science data is often presented in visual form, from maps to graphs to composite raster images. Users should be allowed to refine their queries by interactively selecting portions of the output image. Spatial queries can be adjusted by isolating a region (or, to extend a search, select an area encompassing the one in question), details of objects should be available by pointing, and temporal queries can be performed using icons and graphical time scales.

7. *Update policies and transaction management.* Standard database practices are not sufficient to provide efficient update policies and transaction management of very large scientific databases[8]. This is an important research area which needs to be investigated.

We are currently generating a prototype system based on the concepts described above, using a subset of digital carto/geographic data formats (as long as samples from several distinct time periods and scales are available) to permit concentration on the object manipulation and analysis aspects of the system.

B. System Requirements

GISs are database systems that allow the manipulation, management and display of geographical data[9]. They differ from a conventional DBMS in many aspects:

1. The amount of data is very large.

2. The data represent complex objects.

3. Special spatial operators are required and are more complex than conventional numeric operators.

4. Relationships between objects are not easy to define and may have to be derived.

To the database community, GISs mainly relate to spatial or image databases. Their basic requirements are:

Data Models: While early systems were image-oriented and had limited capabilities, object orientation is gaining acceptance for the modeling of geographical data[10, 11]. Other efforts concentrated on the extension of the relational model. Examples of these efforts are ARC/INFO[12] and the systems proposed in[13, 14]. While object orientation seems very promising, this technology has not been used to its full potential in a GIS.

Query and Manipulation: GISs need a powerful query language for spatial and non-spatial data. Query optimization techniques for these languages are elaborate[15]. Most of the proposed techniques are extensions to QUEL or SQL, such as GEO-QUEL[16], PICQUERY[13], PSQL[14], and GEOQL[15]. For example in GEOQL[15], the SQL language is extended by a set of spatial operators such as *intersects*, *adjacent*, *joins*, *ends_at*, *contains*, *situated_at*, *within*, *closest*, and *furthest*. One should observe that this list is not meant to be exhaustive, and there is no consensus in the research community on its completeness[17]. Primitive (spatial) operators are not clearly defined and need a mathematical base[18]. A scientific user may (and will) need to create new operators. These operators may or may not be defined by combinations of existing spatial operators.

Storage Techniques and Spatial Access Methods: This area of research is important for a GIS. Advances in this field include a wide variety of access methods and structures for the spatial manipulation of data[19, 20]. A large variety of file and data structures is available, but a good comparative study of the performance of the different methods still needs to be performed. The results of such a study would be to provide design guidelines for spatial access methods[17].

We have synthesized two additional important requirements essential to global change studies and coupled with a GIS:

Time dimension: Encapsulation of the *temporal* component of data objects is essential and should be handled within the DBMS[21]. The technology is still maturing and issues related to temporal databases are gaining importance in the research arena[22, 23]. Some of the issues in temporal modeling for spatio-temporal databases are discussed by Qiu *et.al.*[24]. An interesting approach has been proposed[25] with the temporally oriented data model (TODM). This model is based on the *data cube*, a three-dimensionally oriented data object, where time, objects, and attributes are the dimensions of stored data. We foresee that GIS and global change research will in general require integration with a *spatio-temporal* database system.

Operator dimension: Global change research requires the use of existing and forthcoming *scientific methods* or *operators* which can be applied to analyze scientific data. This requirement can be generalized to scientific databases. The operators should be integrated and managed within the scientific DBMS. Two non-exclusive approaches are proposed:

1. management of these operators as object types within the database, and/or
2. development of an extensible language based on an object class *operator*.

With these two additional dimensions, we propose a 3-dimensional data cube construct oriented along the following coordinates: the time dimension, the spatial dimensions (this is a two-dimensional cooordinate), and the data type (operator object class) dimension.

French *et.al.*[1] report that there is no agreement as to where operators belong. The approach outlined here is to design an extensible data model where we attempt to include the operators as an intrinsic part of the DBMS.

C. Functional Requirements

It is critical to understand the analysis needs of global change researchers and integrate these needs into the data management environment[1]. A preliminary set of these needs is[4]:

Normalize diverse data sets to a common environment and scale.

Interpolate sparse data in space and time.

Validate data for accuracy.

Derive reference data based on multiple inputs.

Compute secondary spatial data fields from various sources and time scales, including time series analysis on data sets.

Compare multiple estimates derived via different means.

Detect anomalies in the data.

Display information in traditional and non-traditional manners.

Calibrate and check data for consistency.

Aggregate data over different scales and data types.

II. Architecture Overview

We are currently designing the prototype interfaced to POSTGRES. We will make every effort to integrate the essential components of a GIS by providing an extensible language based on user definable operators (Figure 1). The language is based on the object class *operator* (Figure 2). In our system, the subclasses will be: *selector*, *calibrator*, *predictor*, *displayer*, *deriver*, *comparer*, *checker*, *editor*, *updator*, and *converter*. Each of these subclasses include an extensible set of methods which can be applied to spatial or non-spatial objects. Each object will have an attribute list specifying to which objects classes they apply. Additional flexibility can be achieved by providing deductive and inference capabilities for these operator subclasses.

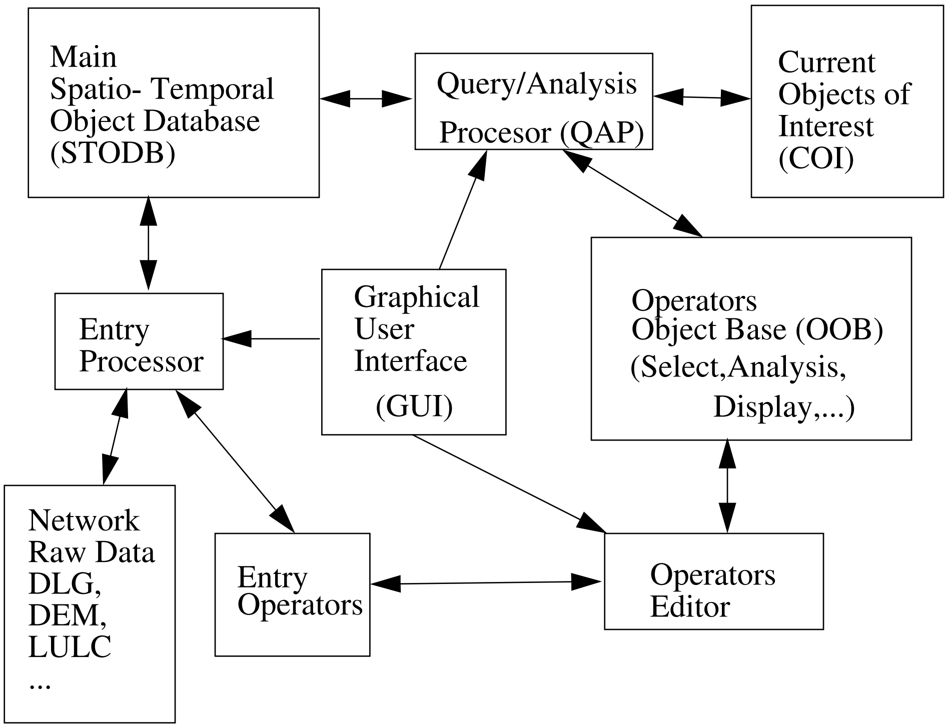

Figure 1. STDBMS System

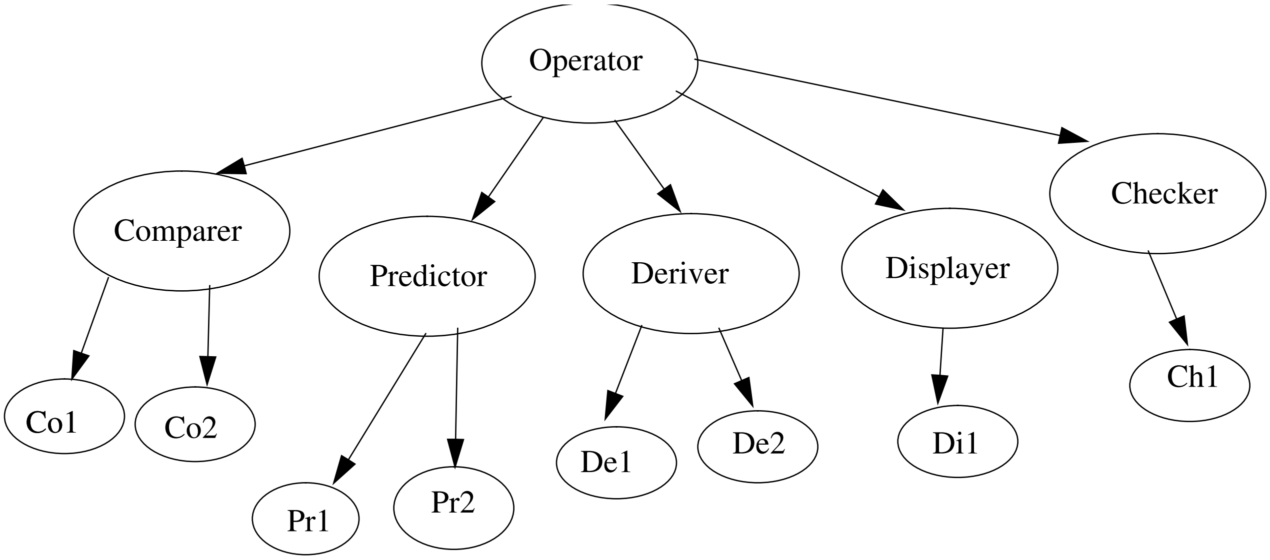

Figure 2. Class of operator objects

Figure 1 shows the main functional components and their relationships. Most components fall into one of two types: Database access and Operator access. All interaction occurs through a Graphical User Interface. The components are:

Graphical User Interface (GUI): The GUI is responsible for all interaction between the user and the system. It manages windows, responds to input events, displays help messages, and invokes programs and procedures for the user.

The main mode of operation uses a REQUEST—PERFORM—DISPLAY loop. The user makes a request of the system, which causes some action to be performed. The result of performing that action is displayed and the system is ready to respond to new requests. Subsequent requests may result in new data being displayed or may alter the display characteristics of previously displayed data.

Main Spatio-Temporal Object Database (STODB): The Main Spatio-Temporal Object Database is an very large database containing many data formats. It must be flexible enough to store data objects that cannot be specified at system design time, yet efficient enough to provide rapid access. The STODB will be layered upon the POSTGRES system[26].

Query/Analysis Processor (QAP): The Query/Analysis Processor responds to queries and requests from user via the GUI. When a request arrives, this module decides if it can be satisfied by a database access, and if so, how best to structure the access. If the request involves the application of one or more operator objects, the QAP decides which operators are most applicable. The QAP also serves the function of hiding implementation details from the user.

Current Objects of Interest (COI): The Current Objects of Interest (COI) is a subset of the main Spatio-Temporal Object Database that is being actively manipulated by the user. It acts to focus the attention of the user and the computer on those parts of the dataset that are most relevant to the current task. Conceptually, it is analogous to the current sheet from a set of topographic maps.

The COI acts as both a data cache and a short-term working storage. Data within the COI are stored in the same format as within the main STODB. Thus, the same access routines can be used for both types of data. The advantage is that queries that can be handled within the COI avoid the overhead of accessing the entire STODB.

Objects within the COI have one attribute that main STODB objects do not: they are tagged as being permanent or ephemeral. Ephemeral objects are not generally written to the main STODB, whereas permanent ones are.

Entry Processor: The Entry Processor converts input data from diverse formats into the STODB. Acceptable formats include Digital Line Graphs, Digital Elevation Models, Land Use Land Cover databases, census information, raster data such as imagery, and others. The input data may be found either locally or through a networked data server. The Entry Processor tags all data with source and format for future audit trails.

Entry Operators: Entry Operators describe the transformations that must be applied to input data in order to convert them into STODB-compatible data. Entry Operators may be maintained in the Operators Object Base for ease of access and editing by other components.

Operators Object Base (OOB): The Operators Object Base stores operator objects. Access to the OOB is through the Operator Editor and the Query Analysis Processor (QAP). In order for the QAP to be able to efficiently locate relevant operators, operators in the OOB must provide certain standard information, including: descriptive text information for on-line help, operator class to which the operator belongs, number and types of arguments, number and types of outputs, and default parameters (if any). The operators are described in more detail below.

Operators Editor: The Operators Editor is responsible for creating and modify operator objects in the Operators Object Base. One of the features it provides is version control—editing an existing operator will generally produce a new operator.

1. Spatio-Temporal Object Database

The Spatio-Temporal Object Database (STODB) is an extensible database of data objects. Because it is impossible to determine in advance exactly which data types will be needed, any type of data objects will be potentially supported. An important set of objects will be implemented initially, with provisions made to extend the STODB to handle new types. The initial data object types will include points, lines, areas, grids, rasters, and aggregate of any of these types.

Data objects are arranged as a set of fields, where each field consists of a field name and associated data value. There is no limit on the number of fields allowed a data object. Typical fields include: data type, audit trail, neighboring data objects, pixel values (for imagery), larger object of which this is a part, smaller objects that this object contains, date and time of acquisition.

2. Operator Object Base

The Operator Object Base (OOB) is an extensible library of operators (functions) designed to modify, analyze, and display the data objects in the STODB. Many classes of object operators are useful for carto/geographic data and global change analysis. A preliminary set of these operator classes is:

Update: There are three forms of updates which must be supported on the STODB. The first is converting a standard carto/geographic database into its STODB format. This involves adding time stamps and empty Link fields (as initially the data objects are isolated in time). The second is the merging of two databases, which mainly consists of linking the associated entities across the time dimension. The third is to edit an STODB to eliminate errors in either the information itself or in the links created during merging.

Select: Selection is the mechanism by which the user or operator objects extract data objects from the STODB. Selection can be based on objects, classes or subclasses of objects, or attributes (explicit or derived) of objects (e.g. location, shape characteristics). Selection may be made on exact matches or ranges of values. It is critical in change analysis to be able to select objects based on some similarity metric, thus Select may invoke the Compare operator. Results of inquiries can be used as input to any other operator. The objects selected may be aggregated into a higher level object.

Predict: To predict how a data object will change over time involves the selection of an object or set of objects, the desired final time, and a Predictor. A Predictor uses information about the changes of a data object and perhaps other objects within a given locale (or data about the locale itself, e.g. temperature trends) or similar objects at different locations to estimate the value, position, or shape of the object at a different time. Predictors may also incorporate process models derived theoretically or experimentally.

Derive: Quantitative analysis of selected data is a crucial component in all aspects of the system. Derivers, like Predictors, can vary significantly in complexity and type of information used. Standard statistical computations, shape measures, and other processes which extract some value or description from the data fall into this operator class.

Compare: Comparisons can be done in a variety of modes, including comparing two or more sets of data objects(for similar characteristics) and comparing the results of two or more Predict or Derive operators. Various metrics for comparison are needed, including fractal analysis, image morphology, transforms, principle components, etc.

Display: Visualization has long been an integral component to data analysis, and nowhere is this more true than in the cartography/geography discipline. The output format of most queries to a GIS is a map, which can be composed of vector or raster data and use a wide range of symbology and graphical entities and attributes to convey the information. Research to date at developing visualization methods for spatio-temporal data has mostly concentrated on either tracking points over time with a traditional line graph or using animation or time selection to view the changing data[27, 28]. In our system, each type of object will be provided with a range of visualization operators, and other operators will be developed to allow compositing of multiple objects. One goal is to extend the current methods to use time as a spatial dimension as well as incorporating novel techniques for generating visualizations of high-dimensional data[29] and correlations between different objects over different time periods[30].

Calibrate: Data objects to be used by operator objects must be calibrated to a consistent spatial and temporal reference point and scale. Many calibration techniques in current use involve some degree of user interaction, although research is underway to automate the process in certain circumstances[31].

Check: Checks must be made of results from Predictors and Derivers for both consistency and sensitivity.

III. Example of Operational Characteristics

Some of the generic tasks one might want to perform in a STDBMS are:

1. **Find** all objects [in area A] which are similar to object X based on metric M.

2. **Estimate** the value (could be non-scalar) of feature F of object X at time T given its values at times $\{t_1, t_2, \ldots, t_n\}$ using predictor P (which may in turn invoke retrieval of other values, perform preliminary computations, involve process modeling, heuristics, etc.).

3. **Correlate** the changes in feature A of object X between times T_1 and T_2 with the changes in feature B of object Y between times T_3 and T_4.

4. **Display** objects $X_1, X_2, \ldots, X_n$ (results of queries or operations) using visualization operators $V_1, V_2, \ldots, V_m$ (some operators will assume certain types of objects, may employ combinations of objects, etc.).

5. **Compare** output from predictor P to real data for object X (remove most recent data in time-series, perform prediction, and compare results to actual data).

If we consider the level of interpretation of the data which is stored in the main spatio-temporal database (STODB) as shown in Figure 1, one can assume that hypercube data consists of *raw*, *calibrated*, *validated*, *inferred (or derived)*, and *interpreted* data constructs. Relationships between these high-level objects are represented following the object-oriented paradigm. To clarify the concepts introduced so far, let us consider an example related to the study of change based on vegetation index[27]. Considering sensor data accumulated twice every month, these data need to be "calibrated" before use. The calibration method is important. So, while the raw data are 3-dimensional (time, object, and attributes), the calibrated data are 4-dimensional as we need to attach the calibration method to it.

For example, considering the month of May for 1990 and 1991, we have the representation of Figure 3. The raw data are processed according to operator "Calibrator" using method `Ca1` and the resulting object (`O2`) is linked to `O1`. The same is performed for the other data accumulated. Now assume that we want to detect change and then analyze it. We may take the objects `O2` and `O3` which relate to calibrated vegetation indices for the same month of May for two consecutive years (1990, and 1991). We apply a "comparison" operator using method `Co2`, which is a straightforward difference operation. One may find that the data representing rainfall time series affects the selection of which object should be involved in the operation. Method `Co3` is specifically designed to do that. It analyzes the rainfall data and determines that a time warp is needed to compensate for rain lag or delay from year to year. It selects `O3` and `O4` for the difference operation. This results in the "hyper-object" `O6`.

Finally, we show an example of a display operator involving a region contour map object `O7` and the vegetation index `O2`. `O7` is generated using contour data stored in DLG format and converted using the Conversion operator method `Cv2`. The result object is `O8`.

IV. Impact and Conclusions

We believe this research will impact the fields of geographical information systems and global change research as well as the database community. Some of the important aspects of this project are:

1. It provides an integrated DBMS testbed for scientific database management. It is an effort to extend the database technology with an intrinsic class of operators, which is extensible and responds to the growing needs of scientific research.

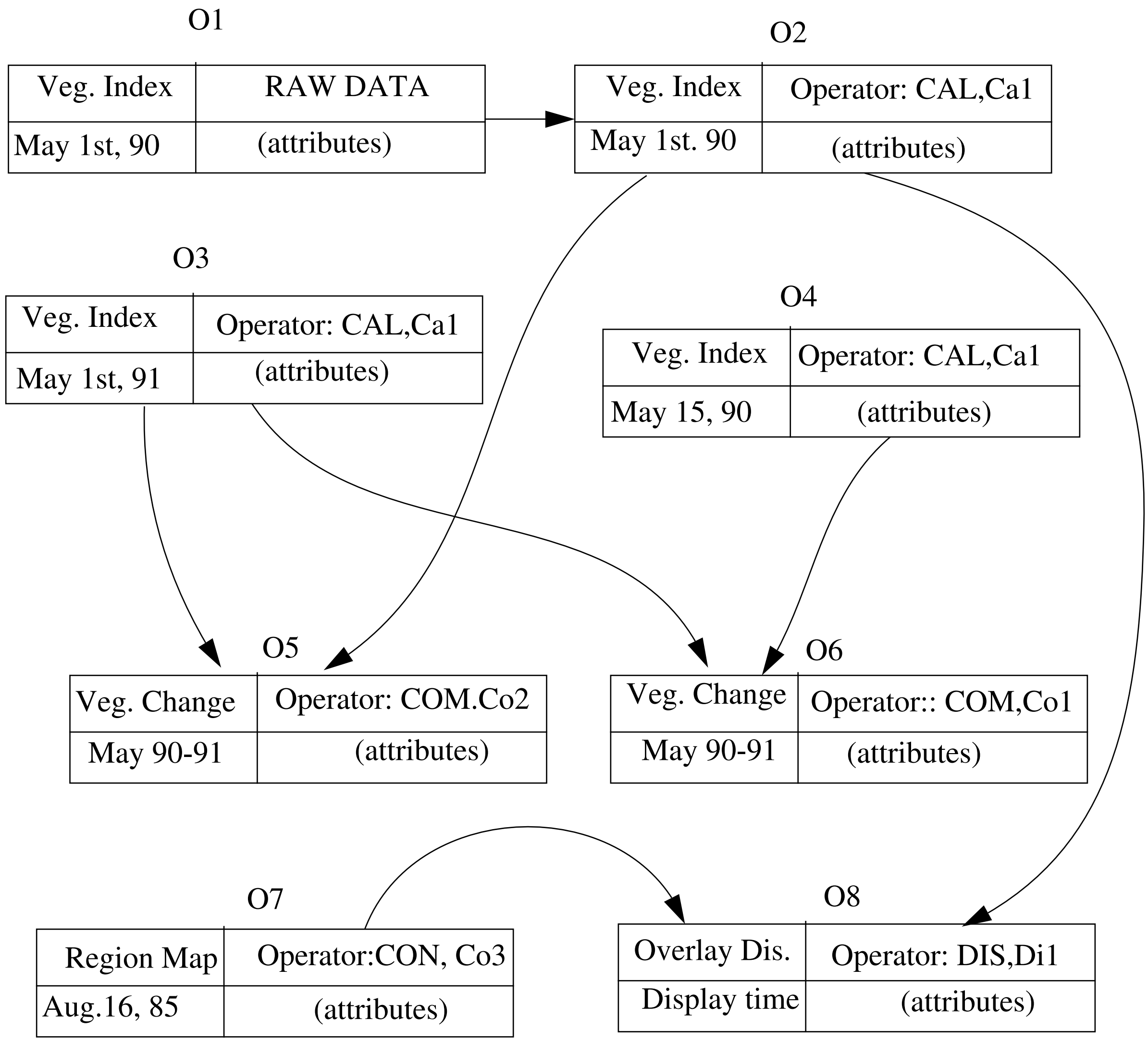

Figure 3. Hyper-object representation

2. It integrates the spatial and temporal domains of scientific databases together with the interactive, dynamic, and evolving aspects of scientific investigations involving very large amounts of data.

3. It provides a testbed for the development and testing of new algorithms for the analysis and search of very large scientific databases. It will provide a framework for the evaluation of the results, along with feedback to the scientific user for further refinements and modifications.

4. It addresses requirements for the manipulation of very large scientific databases at different levels of data interpretation. The object-oriented paradigm is coupled with the temporal dimension and applied at the level of the data and scientific anal-

ysis techniques to provide a coherent and inter-related view of very large scientific databases.

References

1. J.C. French, A.K. Jones, and J.L. Pfaltz, "Summary of the Final Report of the NSF Workshop on Scientific Database Management," SIGMOD Rec. **19**-4, pp. 32–40 (1990).
2. Committee on Earth and Environmental Sciences, "Our Changing Planet: The FY 1991 Research Plan of the U.S. Global Change Research Program" (1990).
3. Committee on Global Change of the Commission on Geosciences, Environment, and Resources, National Research Council, "Research Strategies for the U.S. Global Change Research Program," Natl. Acad. Press (1990).
4. L.T. Steyaert, "Investigating the use of geographic information systems technology in the computer workstation environment for global change research," Proc. 1989 ASPRS/ACSM, pp. 46–54, (1989).
5. M.E. Jakubauskas, "Utilizing a geographic information system for vegetation change detection," GIS/LIS, pp. 56–64 (1989).
6. S.M. Jampoler and B.N. Haack, "Use of GIS to identify land use change: Kathmandu, Nepal," GIS/LIS, pp. 75–84 (1989).
7. R. Leemans, "Possible changes in natural vegetation patterns due to global warming," Global Change Database Project-Pilot Project for Africa, Workbook, J.R. Eastman, J. Kineman, R. Dodson, M. Livingston, N. Azimi, ICSU-IGBP and Panel on World Data Centers (1990).
8. A. Silberschatz, M. Stonebraker, and J.D. Ullman, "Database Systems: Achievements and Opportunities," SIGMOD Rec. **19**-4, pp. 6–22 (1990).
9. B.C. Ooi, R. Sacks-Davis, and K.J. McDonell "Extending a DBMS for Geographical Applications," Int. Conf. Data Eng. pp. 590–597 (1989).
10. J.A. Orenstein and F.A. Manola, "PROBE Spatial Data Modeling and Query Processing in an Image Database Application," IEEE Trans. Software Eng. **14**-5, pp. 611–629 (1988).
11. L. Mohan and R.L. Kashyap, "An Object-Oriented Knowledge Representation for Spatial Information," IEEE Trans. Software Eng. **14**-5, pp. 675–680 (1988).
12. ARC News 12, 1, Environmental Systems Research Inst. (1990).
13. T. Joseph and A.F. Cardenas, "PICQUERY: A High Level Query Language for Pictorial Database Management," IEEE Trans. Software Eng. **14**-5, pp. 630–638 (1988).
14. N. Roussopoulos, C. Faloutsos, and T. Sellis, "An Efficient Pictorial Database System for PSQL," IEEE Trans. Software Eng. **14**-5, pp. 639–650 (1988).
15. B.C. Ooi, "Efficient Query Processing in Geographic Information Systems," Lecture-Notes in Computer Science **471**, Springer-Verlag (1990).
16. R. Berman and M. Stonebraker, "GEO-QUEL: A system for the manipulation of geographic data," Comp. Graphics **11**-2, pp. 186–191 (1977).
17. O. Guenther and A. Buchmann, "Research Issues in Spatial Databases," SIGMOD Rec. **19**-4, pp. 61–68 (1990).
18. M. Engenhofer, "Spatial Query Languages," Ph.D. Thesis, U. Maine, Orono (1989).
19. H. Samet, "The Design and Analysis of Spatial Data Structures," Addison-Wesley (1990).

20. H. Samet, “Applications of Spatial Data Structures, Computer Graphics, Image Processing, and GIS,” Addison-Wesley (1990).

21. G. Langram, “A review of temporal database research and its use in GIS applications,” Int. J. Geo. Inf. Sys. **3**-3, pp. 215–232 (1989).

22. R. Snodgrass, “Temporal Databases: Status and Research Directions,” SIGMOD Rec. **19**-4, pp. 83–89 (1990).

23. M.D. Soo, “Bibliography on Temporal Databases,” SIGMOD Rec. **20**-1 (1991).

24. K. Qiu, N.I. Hachem, M.O. Ward, and M.A. Gennert, “Providing Temporal Support in Spatio-Temporal Data Base Management Systems,” WPI CS-TR-92-3, (1992).

25. G. Ariav, “A Temporally Oriented Data Model,” ACM Trans. Database Systems **11**-4, pp. 499–527 (1986).

26. M. Stonebraker, L.A. Rowe, and M. Hirohima, “The Implementation of POSTGRES,” IEEE Trans. Knowledge and Data Eng. **2**-1, pp. 125–142 (1990).

27. J.R. Eastman and J. McKendry, “Change and Time Series Analysis in GIS,” UNITAR (1991).

28. M. Monmonier, “Strategies for the visualization of geographic time-series data,” Cartographia **27**-1, pp. 30–45 (1990).

29. J. LeBlanc, M.O. Ward, and N. Wittels, “Exploring N-Dimensional Databases,” Proc. IEEE Conf. Visualization, pp. 230–237 (1990).

30. M.O. Ward and D.S. Adams, “Nucleotide Sequence Analysis Using Correlation Images,” Proc. IEEE Conf. Visualization Biomed. Comp., pp. 49–56 (1990).

31. D.G. Corr, A.M. Tailor, A. Cross, D.C. Hogg, D.H. Lawrence, D.C. Mason, and M. Petrou, “Progress in Automatic Analysis of Multi-Temporal Remotely Sensed Data,” Int. J. Remote Sensing **10**-7, pp. 1175–1195 (1989).