An Overview of the Gaea Project *

Nabil I. Hachem, Michael A. Gennert, Matthew O. Ward
Computer Science Department, Worcester Polytechnic Institute,
Worcester, MA. Email: {hachem,michaelg,matt}@cs.wpi.edu

1 Introduction

The long-term goal of the Gaea project is to develop an extensible, object-oriented data management and analysis system to be used by researchers in the field of global change [1].

The current goal is to develop a prototype, which can be used by geographers in a user-friendly manner, yet permits integration of heterogeneous and complex data types, and interactive development of sophisticated methods for data analysis, prediction, and display. Focus is on the object manipulation and analysis aspects of the system, and the management of "meta-data," that is, data about the data.

The general abilities of the system will include: 1) deriving information about spatio-temporal objects, 2) maintaining information on the evolution of data objects, 3) providing derivation semantics for data objects, 4) the management of user-specified experiments, 5) providing a visual environment for browsing, querying, and analysis, and 6) providing user-extensible data and operator types.

2 System Architecture of Gaea

Gaea's overall architecture is divided along three levels (Figure 1): 1) the Gaea Kernel, core of the prototype, provides the essential meta-data management capabilities; 2) the Visual Environment provides visual facilities for scientific experiment design, data definition and manipulation, including querying, and semantic browsing; and 3) the Postgres [3] 3rd Generation DBMS serves as the backend.

2.1 Managing Derived Data in the Gaea Kernel

The most important aspect of the Gaea Kernel is the meta-data manager. It provides a framework for capturing and managing scientific data derivation histories [2]. Meta-da¡ta are viewed by the system at three semantic levels (Figure 2): 1) the *high level semantics* view which will provide the user with means to design and develop logical views of experiments, 2) the *derivation semantics* view which provides for the management of (scientific) derivations of data, and 3) the *system level semantics* which are essentially the abstract data type (ADT) view of the system.

2.1.1 High Level Semantics, or the Experiment Level

This level records the information that is necessary for understanding a specific experiment. In global change research, it is difficult to agree on carefully designed experiments. The Gaea kernel supports experiments through the experiment manager module of the meta-data manager. This module is capable of manipulating conventional semantic modeling constructs. In addition, we introduce the notion of *concepts*. A **concept** is a representation of a spatio-temporal entity set, extended with an imprecise definition. Concepts are very common in scientific databases.

For example, PERSON is an entity set with a well understood definition. It may be considered as a concept with a well defined and agreed upon meaning. In GIS, a DESERTIC REGION is an entity set whose definition may differ from one user to another. An acceptable definition of a desert must include

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

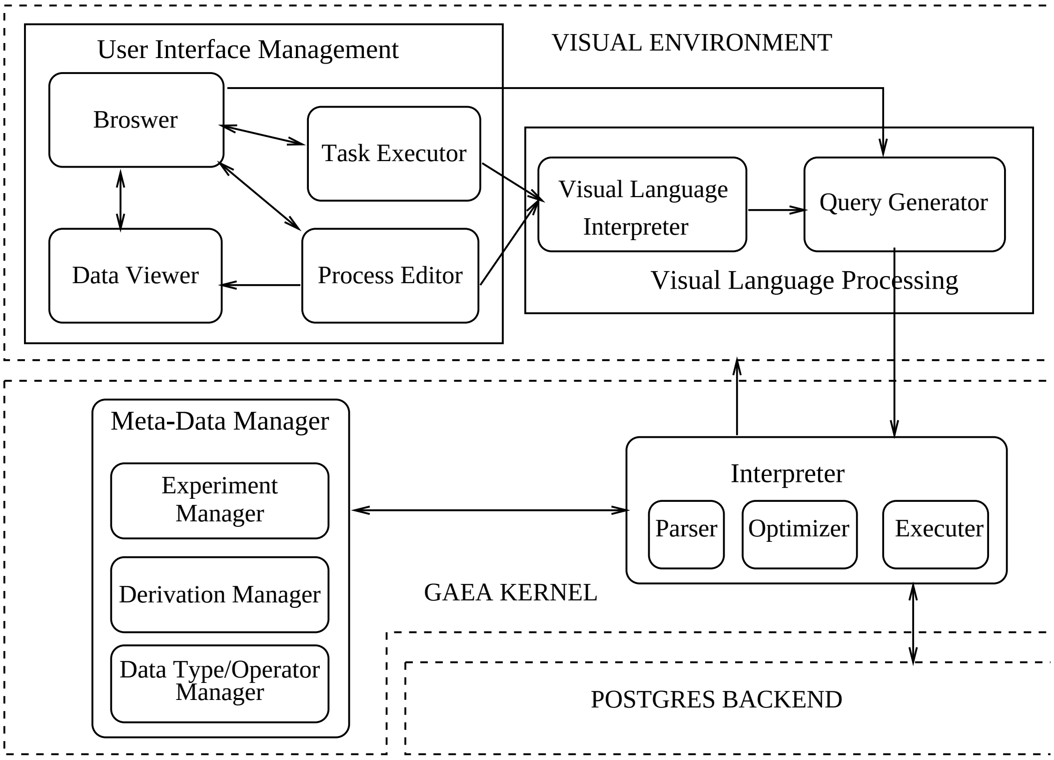

Figure 1: Architecture of the Gaea system prototype.

consideration of the amount of precipitation received, its distribution over a calendar year, the amount of evaporation, the mean temperature during the designated period, and the amount and utilization of the radiation received. Furthermore, every one of those factors may have different metrics.

At this high level of abstraction, we model deserts with a specialization hierarchy (Figure 2). This hierarchy does not capture the relationships between other concepts involved in the definitions of deserts. While general relationships can be provided using well-proven semantic modeling technology, new semantics for data derivation are necessary.

2.1.2 Derivation Semantics Level

The leaves of the concept hierarchy in the high level semantic layer map to a set of non-primitive classes in the derivation semantics layer. For example, in Figure 2, "hot trade-wind desert" maps to the set of (non-primitive) classes {C2, C3, C4, C5}. "Vegetation change" can be characterized using a derivation process (P7) based on principle component analysis (PCA); the outcome of which is class C7. Alternatively P8, based on standardized PCA, can be used and results into class C8. Class C5 could be the result of an interpolation on C2 using P5.

The derivation semantics layer records the derivation relationship among classes of data. Such relationships can also be used for the generation of new classes of data. Typically, when data are not stored in the database, we may generate the needed data with the help of such derivation relationships. The basic constructs used are: 1) A **Process** which captures the description of a scientific procedure used for the generation of new concepts from other concepts. 2) A **Task** which is the instantiation of a process with input data objects. Every task will generate a set of objects for the output class.

Formally, a process defines a mapping between a set of input object classes and an output object class. Essentially, the outcome of a process is a unique class which is a member of a concept. Thus, object classes which do not represent base data are solely defined by their derivation processes. In this way a process captures the semantics of data derivations.

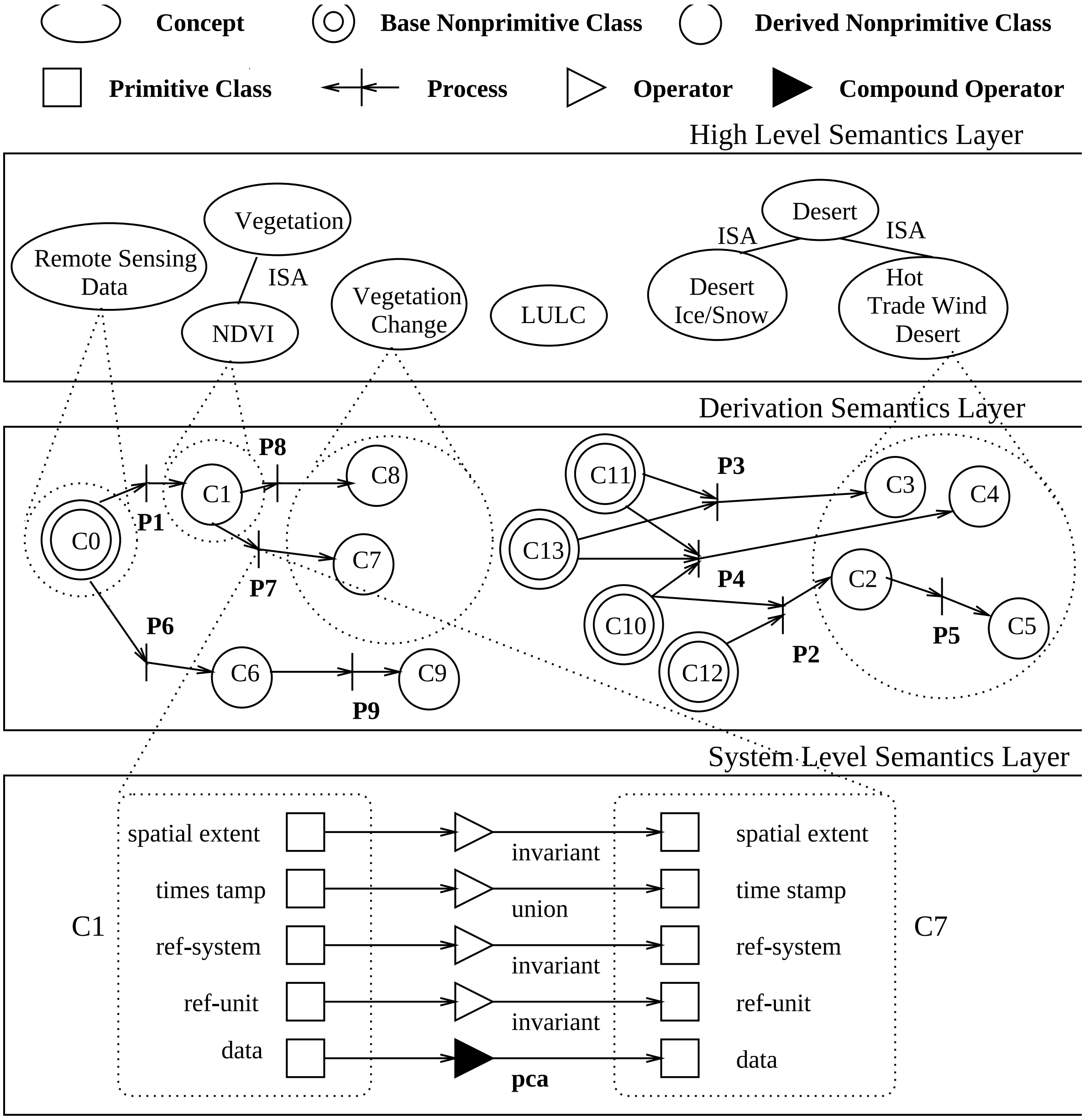

Figure 2: The three semantic layers in Gaea.

2.1.3 System Level or Low Level Semantics

The system level semantics of Gaea is responsible for the management of abstract data types (ADTs). Following the object-oriented paradigm, ADTs in Gaea are primitive classes encapsulated with the methods or functions applicable to them.

The mapping between the derivation semantics layer and the system layer consists of the mapping of a process as a transformation of a set of input classes to an output class using operators that are applied to primitive classes. For example, process P7 transforms class C1 into class C7. The mapping between input and output attributes is shown in the lower portion of Figure 2. It is observed that `pca` is a compound operator. It is composed of a network of intercommunicating operators, whose structure is discussed in [2]. This network can be considered as a data flow network of functional operators that are applied on primitive classes, such as spatial coordinates, temporal attributes, and raster images.

2.2 Gaea Visual Environment

The Gaea Visual Environment provides the user interface to Gaea [4](Figure 1). The interface generates commands in the visual langauge, which are converted to queries for interpretation by the Gaea Kernel. The user interface functionality can be decomposed into four interrelated activities.

The **Browser** permits the perusal of the contents of the database and its meta-data. Users specify actions in the browser by interacting with visual representations such as images, graphs, lists and icons to indicate constraints on searches. Thus, to determine for which years there exists rainfall data for Australia, the user would draw a box around the section of the world map which contains Australia, browse all concepts available for Australia, choose rainfall from the result list, and specify the temporal resolution to be yearly. The particular years for which data exists would then be shown on a time line.

The **Data Viewer** is an interactive data visualization toolkit. Data of various types, such as tabular data, image data, and vector data, can be visualized using a set of visualization operators. Results of a visualization, such as a map of vector data, can be used for specifying spatial context in the Browser.

The **Process Editor** permits the user to interactively specify a process via a visual language, whereby data and operators are linked together in a network. Users may choose to edit an existing process, or create a new one from scratch and associate it with a new or existing concept. The system performs consistency checking on the connections between nodes, and the Browser may be invoked to search the database for compatible operator or data classes. Processes can be tested in the Process Editor by specifying input data objects. Results can be visualized by the Data Viewer.

The **Task Executor** allows the user to associate specific data objects of specific concepts to a process or a series of processes and execute the result. The Browser may also be used to locate and specify the desired objects. A fully specified task may be saved as an experiment. Meta-data, extracted from the nodes and supplemented by the user, may be stored with it.

3 Summary

In the first phase of this project, we have synthesized the needs of the global change research community in terms of the management of data, operators, and experiments. This led to a preliminary design for the Gaea system architecture, including the spatio-temporal data and operator models, data language, and the visual environment. By the end of the summer of 1993 we will have completed the implementation of a prototype system with a significant percentage of the desired functionality, including the visual browser, simplified object editors, translator/interpreter, data, operator, and derivation managers, and the interface to the Postgres backend. The prototype will be tested with a small set of geographical data types and some simple analysis and display operators used in global change studies.

References

[1] N.I. Hachem, M.A. Gennert and M.O. Ward, "A DBMS Architecture for Global Change Research," Proceedings of ISY Conference on Earth and Space Science, Pasadena, CA, February 1992.

[2] N.I. Hachem, K. Qiu, M.A. Gennert, and M.O. Ward, "Managing Derived Data in the Gaea Scientific DBMS," WPI-CS-TR 92-08, December 1992.

[3] M. Stonebraker, L.A. Rowe, and M. Hirohima, "The Implementation of Postgres," IEEE Trans. Knowledge and Data Eng. **2**-1, pp. 125–142, 1990.

[4] Y. Zhang, M.O. Ward, N.I. Hachem, and M.A. Gennert, "A Visual Programming Environment for Supporting Scientific Data Analysis," WPI-CS-TR 93-01, March 1993.