CONF-9608154--1

ANL/DIS/CP--90928

RECEIVED
SEP 03 1996
OSTI

ADVANCED INFORMATION SCIENCE AND OBJECT-ORIENTED TECHNOLOGY FOR INFORMATION MANAGEMENT APPLICATIONS*

John R. Hummel and Craig E. Swietlik, Decision and Information Sciences Division, Argonne National Laboratory, Argonne, IL 60439-4832

1. INTRODUCTION

The role of the military has been undergoing rapid change since the fall of the Berlin Wall. The kinds of missions the US military has been asked to participate in have often fallen into the category of "Military Operations Other Than War" and those involving military responses have been more of a surgical nature directed against different kinds of threats, like rogue states or in response to terrorist actions.

As a result, the requirements on the military planner and analyst have also had to change dramatically. For example, preparing response options now requires rapid turnaround and a highly flexible simulation capability. This in turn requires that the planner or analyst have access to sophisticated information science and simulation technologies.

In this paper, we shall discuss how advanced information science and object-oriented technologies can be used in advanced information management applications. We shall also discuss how these technologies and tools can be applied to DoD applications by presenting examples with a system developed at Argonne, the Dynamic Information Architecture System (DIAS). DIAS has been developed to exploit advanced information science and simulation technologies to provide tools for future planners and analysts.

2. ADVANTAGES OF OBJECT-ORIENTED TECHNOLOGY AND DATABASES

Object-oriented modeling and simulation provides a powerful way to encapsulate parts of a problem in logical components, called objects, and to simulate them in a logical fashion. The objects that make up a problem can have a physical analog in the real world, such as a tank, or they can be of a conceptual nature, such as an "order."

The objects can be broken up into many parts to enable one to represent a "thing" to whatever level of detail required. For example, military aircraft can be described in terms of fixed or rotary wing aircraft. Then, the fixed wing aircraft could be further broken down into jet powered or propeller driven and so on. The degree of detail used to describe the objects can be based on whatever criteria are appropriate for a given problem. Figure 1 gives an example of an object taxonomy that is being used in the testing effort for the prototype DoD High Level Architecture being

*Work supported under a military interdepartmental purchase request from the U.S. Department of Defense, Joint Chiefs/J-8, through U.S. Department of Energy contract W-31-109-Eng-38.

developed by the Defense Modeling and Simulation Office. The object taxonomy shown is that being used by the Joint Training Federation Prototype that is being supported by the Joint Simulation System (JSIMS) Joint Program Office.

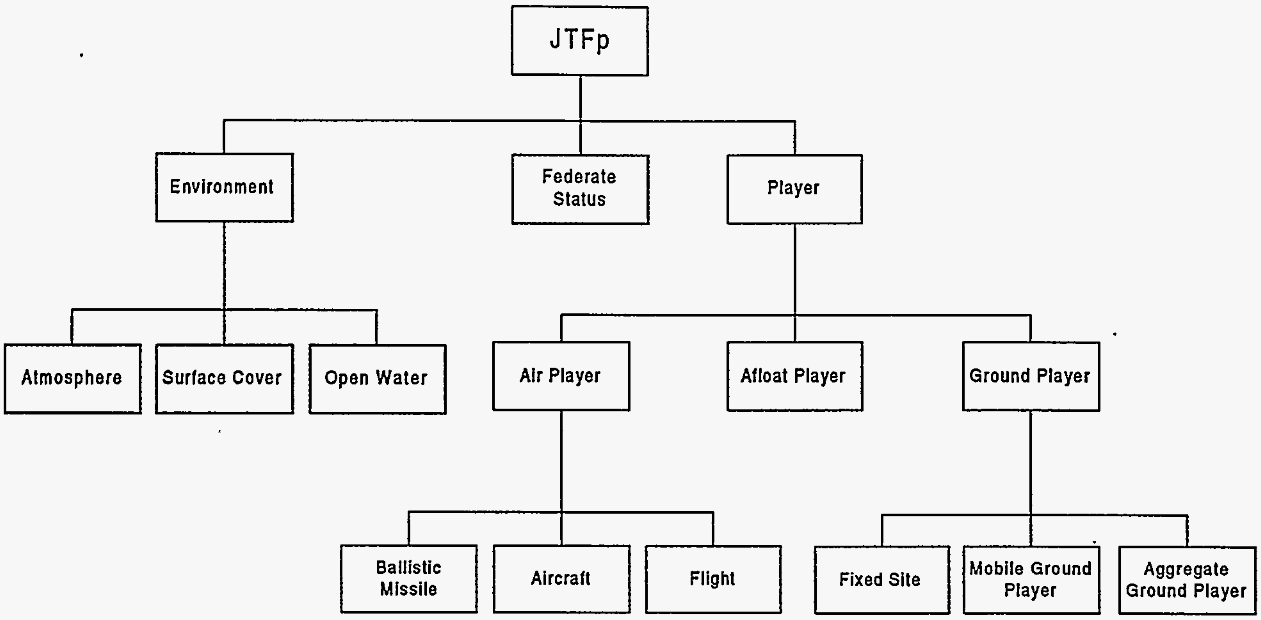

Figure 1. The Object Taxonomy Being Used By the JSIMS Joint Training Federation Prototype That is Testing the Prototype DoD High Level Architecture.

The subcomponents that make up an individual object are called attributes and can either be static or dynamic during a simulation. If they are dynamic, they could vary as a result of changes in the object itself or as a result of interactions with other objects in the simulation.

The data values that make up the attributes can be individual values or arrays of numerical values, textual information, multi-media products, or output from algorithms or models. The sources of these data can be external to the overall simulation or they can be provided by objects that interact with other objects.

All of the data related to an object-oriented simulation can be stored in an object-oriented database system as opposed to conventional and relational database systems. The traditional relational databases are well suited for tabular or "flat" data, but may not represent actual relationships between data elements and the objects which the data describe. Object-oriented databases, however, can express relationships between objects, including hierarchical dependencies where the data from one object may inherit properties from its class. For example, a "river" object may inherit an entire set of properties by inheritance relationships from its class, which may be a "body of water." The lower level objects are further specifications of the higher object classes. Object-oriented databases allow the data modeler to capture natural relationships between objects, without the artificial process of "flattening" the data to suit a relational database table.

The initial stages of development in the DIAS program began by storing data into relational tables. However, since DIAS was an object-oriented model, the data model was translated from a relational structure to an object model to accommodate the growing needs of the model. This restructuring process initially created significant performance problems and was not naturally compatible with the object model itself. As a result, the database was later translated into a more natural object-oriented database.

3. USING INFORMATION SCIENCE TECHNOLOGIES IN DOD SIMULATIONS AND APPLICATIONS

DoD simulations and applications require a heterogeneous collection of data. The different kinds of data types can range from conventional numerical forms of data to textual information to imagery or to unconventional forms such as sensory forms like touch, smell, sound, or video. The latter three forms are important in simulations and applications involving virtual reality systems.

In the R&D world in which one might have months or years to plan for a simulation, one would have the luxury of being able to collect the "right" forms of data for a given application. However, in the "real" world of the operational community, this luxury is lost and one must be able to collect the required data quickly and accurately. This requirement for a fast response means that one must be able to assemble data quickly from whatever forms of data or information that are available. This means that one must be able to exploit information science technologies to the fullest.

As an example of how one can use information science technologies to solve real world problems, consider the situation of an analyst who has been asked to prepare a response paper on the threat posed by a country or group that had been considered for sometime to be a benign threat. In order to rapidly prepare the response paper, the analyst might have to obtain data from non-numerical sources, such as textual sources. As an example, bibliographic entries might be searched for the names of military personnel who could be key personnel in a command and control structure. Or, databases of arms acquisition records could be searched in order to obtain data on arms shipments. In addition, these text sources themselves could become data sources themselves within the simulation. As an example, one might want to be able to simulate for training purposes a command and control system in which orders are generated and distributed among the different kinds components in the organization.

In order to be able to extract the required information, one must be able to utilize advanced information management technologies such as full text retrieval, key word searches, etc. Although models and simulations traditionally make extensive use of relational databases to store data, much of the real world data is textual data. Military orders, for example, are highly structured text, which can be parsed to populate databases which can then be used in modeling.

The recent emergence of full-text information retrieval engines has given rise to text databases, which can now be effective means of handling large amounts of text data. Military orders, reports, guidelines, policy manuals and the like can be housed in large data repositories and searched using the increasingly sophisticated commercial search engines using natural language and Boolean queries.

Advanced text processing and query tool suites can query against these text databases, possibly geographically separated, to integrate data from multiple text sources. Advancing World Wide

Web technologies, including Java and Web-crawlers are now being used to index large data collections from multiple sources across the Web. The applications of this technology for open source data collections, which contain much of the intelligence data used by analysts, can leverage this full-text technology base for the intelligence community.

The Decision and Information Sciences Division of Argonne National Laboratory has been developing advanced information science and object-oriented simulation technologies for use in addressing many of the difficult problems facing the DoD community. One such example is the Dynamic Information Architecture System (DIAS) that is a fully object-oriented system that can be used by planners and analysts.

4. SUMMARY OF THE DYNAMIC INFORMATION ARCHITECTURE SYSTEM

DIAS is a **general** software framework intended to facilitate holistic management of information processes, including the construction of simulations from component modules according to a user supplied context. The domain of DIAS is flexible, determined by the objects available within DIAS and by the collection of models and other data processing applications which have been gathered by users to address specific information processing concerns. This means that DIAS is easily reconfigurable to different applications, subject domains, and user communities (*e.g.* CINCS, OSD, the Joint Staff, etc.)

The Dynamic Environmental Effects Model (DEEM) is the first well developed application using DIAS. In addition to addressing the simulation needs of the military community, DEEM/DIAS is also being used in the civilian community. For example, DEEM/DIAS will be used by the South Florida Water Management District as their framework for an advanced hydrological modeling system.

4.1 Key Architectural Features of DIAS

Figure 2 shows a high level schematic representation of the DIAS architecture. An external Graphical User Interface (GUI) is used to input the commands that comprise the "context", or the goals and constraints, of the simulation as well as display the results from the simulation. A "Frame Toolkit" contains an extensive library of domain objects and data ingestion utilities that one can use to populate the Frame, or area of interest. In its simplest form, the Frame can represent the geographical region under study.

The Context Manager is used to interpret the requests and instructions supplied by the user (*e.g.* mission requirements, course of action, analysis requirements, etc.) to be utilized during the course of an application development. During simulation set-up, the Context Manager uses an inference engine to match up the goals and constraints supplied by the user to the data and modeling resources linked to DIAS.

The users instructions can be expressed in formats and styles appropriate to a given user community (CINC, analyst, ...). The Context can be expressed in the form of simple instructions, such as "calculate the time required to move a unit of troops from point A to point B" or can be more complicated like "calculate the time required to move a unit of troops from point A to point insuring that the troops avoid detection by opposing forces."

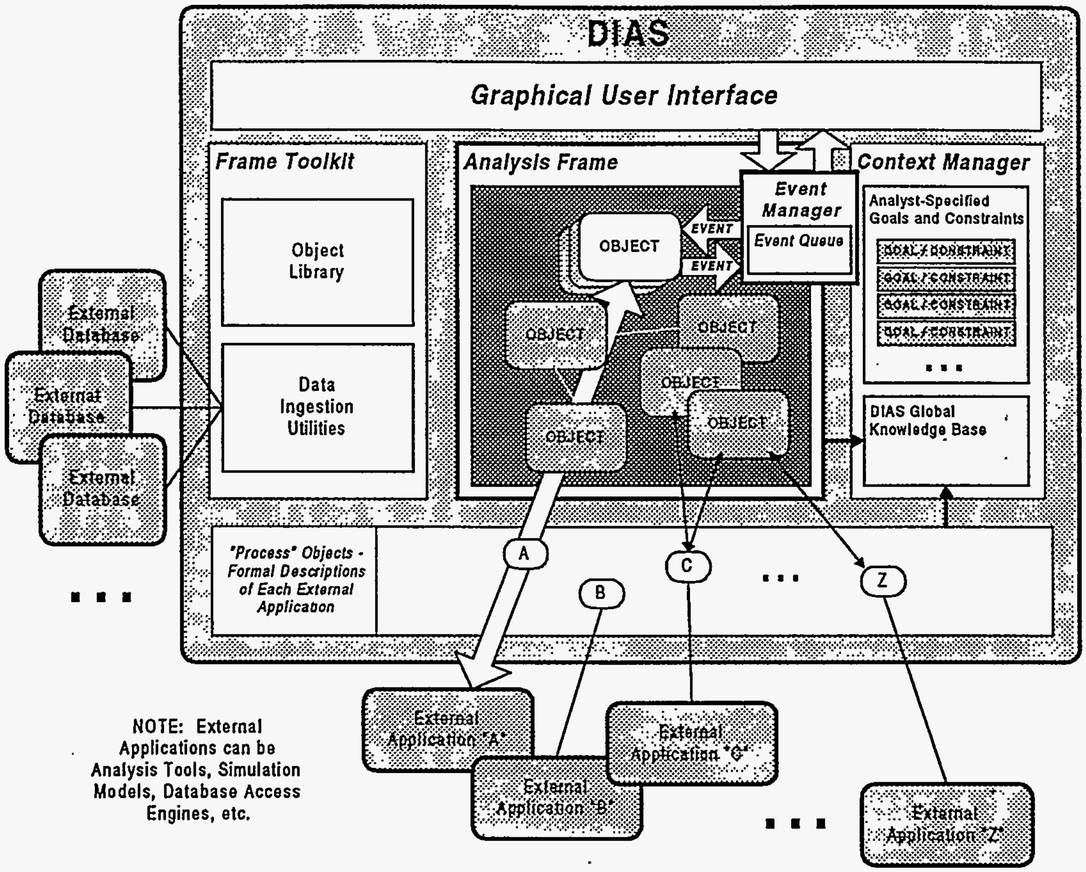

Figure 2. High Level Representation of the Dynamic Information Architecture System.

Once the goals and constraints of the simulation have been expressed, the Context Manager supervises the creation of transient or persistent "Projection" Objects that are used for interprocess data transfer. In addition, once the simulation has begun, the Context Manager also supervises the application process via a global Event Manager which provides the mechanism by which events are posted, activated, received, and activated upon by DIAS objects. Also during the execution of a simulation or analysis, the Context Manager controls the interactions between objects and models or applications using a rules-based expert system that employs rules and knowledge from the DIAS Global Knowledge Base.

DIAS is a fully object-based software system based on the object paradigm which supports distributed, dynamic representation of interlinked processes at variable scales of resolution and aggregation. This means that a DIAS simulation is performed at the level of resolution appropriate to the problem at hand.

A DIAS application is built from a collection of diverse "Entity" objects which may have real-world counterparts (*e.g.* Platform, Unit, Atmosphere, Railyard, etc.) or which may be other modules such as information processing applications. The state of an entity at any instant is defined by the values of state variables held as private data by the DIAS Entity objects. Various aspects of the behavior of the entity are addressed by models or other modules, either external or internal, which are accessed by methods (functions, subroutine) of these objects. These methods are invoked in response to Events which are selectively broadcast by the Event Manager.

DIAS's GUI system includes the "GeoViewer" module, an object-oriented Geographical Information System, that is designed to provide the ability to display results in a spatially oriented manner at user-selectable levels of resolution. The GeoViewer also provides the ability to manipulate, query, and analyze the displayed data. DIAS can also provide the ability to visualize scenes in a photorealistic manner.

DIAS has been developed to provide easy integration of new as well as legacy-type models or objects and application tools, such as the wealth of DoD-funded physics models. In order to integrate a model or application, a "registration" process is employed. Once a model or application has been "registered", it can interact, as appropriate, with any DIAS simulation.

4.2 DIAS Object Class Characteristics

A number of object classes are used within the DIAS architecture. Table 1 lists the primary DIAS object classes. Some of the object classes, like the Entity objects, have a connection to the "real" world and can be created and/or modified by the User. Other DIAS objects, like the Event and Projection objects, are employed to provide specific functionality during a simulation and operate in a fashion transparent to the User. Finally, other objects like the Context and Aspect objects have a connection to the "real world" but are more a conceptual nature rather than a physical one.

Table 1. A Summary of the DIAS Object Classes.

Object Type	**Description**
Frame	The DIAS object class representing the area of interest in a simulation.
Entity	An object with a real world analog (*e.g.*, such as a vehicle, road, atmospheric layer, etc.) with many possible types of dynamic behavior.
Aspect	An object class that is the notional expression of a single kind of an Entity's behavior, such as vehicle movement over terrain.
Context	The object class that is a collection of metadata governing the protocol by which DIAS models or applications operate on state variables.
Event	An object which cues an Entity object to perform a stated Aspect.
Model	An object encapsulating the means of assessing a particular model or application and a formal description of its function via a network of Process objects.
Process	An object class that represents a specific model or algorithm which addresses a particular Entity Aspect.
Projection	An object class that is a transformation of DIAS data for a specific use. A projection object can either be transient or persistent.
Model Controller	A component of a Model object that constitutes a shell around an external model to allow it to converse with DIAS Entity objects.

4.3 Utilizing Information Management Technologies to Provide Data for a DIAS Application

DIAS has been designed to accommodate a wide variety of data types and formats. For example, DIAS can accept terrain elevation and surface features in DMA, USGS, and TIGER formats; interpreted remote sensing imagery; CAD-type data of 3-D objects; weather data in World Meteorological Organization formats, and data from most Commercial-Off-the-Shelf (COTS) Database Management Systems.

Much of the data, however, that can be used in a DIAS simulation can be found in non-standard forms. For example, the rules of engagement in a military operation form one of the kinds of information that are required to form the "context" of a simulation. These rules are generally expressed as textual information rather than numerical data. This kind of information could be capture by a full text retrieval system for use with a DIAS simulation.

In another example, a DIAS simulation may involve a simulation in an area of interest not covered by conventional terrain data products. Instead, one might only have imagery to work with to provide the necessary terrain data. In a situation like this, one would want to have the ability to apply artificial intelligence processes to extract information from the imagery. Argonne has developed just such a system, the Hyperspectral Image Data Exploration Workbench, for extracting surface information from satellite imagery.

5. SUMMARY

The military of the future will face numerous challenges in terms of providing responses and options in response to fast changing events. This will mean that the military community may have to be able to act rapidly and, possibly, in a data poor environment. This will mean that analysts and planners will have to be able to exploit any and all forms of data and information that is available. Argonne National Laboratory has been developing advanced information science technologies and simulation tools to be able to provide these tools kinds of to the DoD community. With these tools, the military planner and analyst will be able to provide answers and options to conditions in a rapidly changing world.

About the Authors

Dr. John R. Hummel is a Program Coordinator/Manager in the Advanced Computer Applications Group of the Decision and Information Sciences Division at Argonne. Dr. Hummel has over 20 years experience in the areas of the atmospheric sciences and advanced computer simulations. Address: Decision and Information Sciences Division, Argonne National Laboratory, 9700 S. Cass Avenue/DIS-900, Argonne, IL 60439-4832; voice 630-252-7189; fax 630-252-5128; e-mail hummel@athens.dis.anl.gov.

Dr. Craig E. Swietlik is the Group Leader of the Advanced Computer Applications Group of the Decision and Information Sciences Division at Argonne. Dr. Swietlik has over 20 years experience doing work in information science, modeling and simulation, and numerical mathematics. Address: Decision and Information Sciences Division, Argonne National Laboratory, 9700 S. Cass Avenue/DIS-900, Argonne, IL 60439-4832; voice 630-252-8912; fax 630-252-5128; e-mail swietlik@dis.anl.gov.