In Search of Meta-knowledge

Antonio M. Lopez, Jr., Ph.D.
Mathematical Sciences
Loyola University Box 51
New Orleans, LA 70118
(504) 865-2657
e-mail: lopez@loynovm.bitnet

Abstract

Development of an Intelligent Information System (IIS) involves application of numerous artificial intelligence (AI) paradigms and advanced technologies. The National Aeronautics and Space Administration (NASA) is interested in an IIS that can automatically collect, classify, store and retrieve data, as well as develop, manipulate and restructure knowledge regarding the data and its application (Campbell et al., 1987, p.3). This interest stems in part from a NASA initiative in support of the interagency Global Change Research program. NASA's space data problems are so large and varied that scientific researchers will find it almost impossible to access the most suitable information from a software system if meta-information (metadata and meta-knowledge) is not embedded in that system. Even if more, faster, larger hardware is used, new innovative software systems will be required to organize, link, maintain, and properly archive the Earth Observing System (EOS) data that is to be stored and distributed by the EOS Data and Information System (EOSDIS) (Dozier, 1990). Although efforts are being made to specify the metadata that will be used in EOSDIS, meta-knowledge specification issues are not clear. With the expectation that EOSDIS might evolve into an IIS, this paper presents certain ideas on the concept of meta-knowledge and demonstrates how meta-knowledge might be represented in a pixel classification problem.

Introduction

There is no single view of what constitutes an IIS nor how to apply AI techniques to develop such a system (Goyal, 1989; Kerschberg, 1990). However, some researchers (Kaula & Ngwenyama, 1990) envision an IIS as evolving from a large number of independently developed systems that communicate and cooperate by passing messages (data, knowledge, and information). These independently developed systems will have evolved using various software paradigms including different AI paradigms such as object-oriented or logic-oriented ones. In addition, the use of neural networks or genetic algorithms to solve very domain specific problems will be supported by advanced technologies tailored for the independently developed system. Each of these independently developed systems will have their own assumptions, constraints, and goals. Yet, they will be "partners in a bigger scheme of things." In this development, there is no global schema. At times, one system will be called upon to pass portions of its knowledge to another system and, likewise, acquire knowledge from their communicating partners as the need arises.

Given the many and varied Earth science systems that have been independently developed by NASA to this point in time and the EOS project that will collect more data than ever collected before, EOSDIS seems ideally positioned to evolve into an IIS. EOSDIS will be responsible for the storage and distribution of large volumes of data that will support scientific research into the global change problem domain. The EOSDIS Information Management System (IMS) will provide the software tools to search, locate, select, and order data archived at Distributed Active

Archive Centers (DAACs). The IMS will manage a set of metadata that includes among other items, directory, catalog, and inventory level information, summary statistics, algorithm descriptions, mission information, and user profiles (McDonald & Blake, 1991). The current stage of development (Version 0) attempts to integrate and expand data management capabilities being used now by different Earth science disciplines. As the EOSDIS IMS evolves, challenges will exist in the specification of the scope of the system and in dealing with the many uncertainties found in the end-user community.

At NASA's Goddard Space Flight Center the Intelligent Data Management (IDM) project team is conducting research into the information and data management needs of Earth and space missions that will produce terabyte-sized spatial databases that cannot be effectively managed using present data management and mass storage technologies (Campbell & Cromp, 1990). This basic research may have an impact on the evolution of EOSDIS IMS. The IDM project team has proposed among many other techniques the use of semantic data modeling to organize object-oriented databases, thereby extending the mass storage model. To test this approach, they have developed an Intelligent Information Fusion System (IIFS) prototype. The IIFS employs several key AI concepts and methodologies such as object/frame representations, multiple inheritance, and rule-based decision making. Applications of AI throughout the IIFS attempt to remove from the end-user (novice to expert) the need to understand the various complexities and nuances of the system and of the particular problem domain. However, the IDM project team has recognized that future science research will require even more comprehensive pre-existing knowledge about the data granules, problem domains, and end-users of the system (Cromp et al., 1992). In short, more meta-information is needed.

Meta-information: Metadata and Meta-knowledge

Meta-information is the underpinning of any IIS. It is the information about the information stored within the system that allows the system to be perceived as intelligent. To take a page from Kidder's book (1981), "meta-information is the soul of an IIS." The IDM project team (Campbell & Cromp, 1990) describes meta-information as incorporating into an IMS knowledge about the structure (syntax) of and the relationships (semantics) between data components, and the hidden questions behind a user's query and the assumptions behind the system's response to that query (pragmatics). The pattern of evolution that this research into meta-information is taking is classical (Lenat & Guha, 1990). The metadata research and development addresses the factual knowledge or the zero-order correction. The research into object-oriented data management with related semantic data modeling holds promise for handling heuristic knowledge or first-order correction. The second-order correction is meta-knowledge. Meta-information is both metadata and meta-knowledge where the metadata is mostly syntax, the meta-knowledge is mostly pragmatics, and both share in the semantics between the data components and the current status of information in the system.

Generally speaking, the metadata for an IIS standardizes _what_ data describes the information resource, and it formalizes policies by specifying _what_ data must be maintained as the system is developed and used (March & Kim, 1989). Intelligent metadata management is a key ingredient in the performance of an IIS (Kaula & Ngwenyama, 1990). In addition, the performance of an IIS can be improved by supplying it with meta-knowledge. Meta-knowledge comes in many forms, but two general categories seem to encompass much of what is considered to be meta-knowledge. First, there is meta-knowledge that guides the user of the IIS to the "best" rules to apply, that is, strategies that will focus quickly on the relevant group of rules to be used on a particular problem (e.g., browsing and searching). This category contains knowledge regarding knowledge permanency, priorities of knowledge, and knowledge on how to resolve conflicting knowledge from different sources. For example, in this category, meta-knowledge

on an ozone data set would include knowledge about a derived data set obtained by a researcher (Is this an interim processing file with additional work forthcoming?), its level of reliability (Was the pixel classification work done for thoroughness or expedience?), its relative importance with respect to other derived data sets (How does this derived data set match the profile (level of expertise, desire for detail, etc.) of the researcher making the request?), and an evaluation of the performance of the cognitive processor (novice to expert) who developed it. Second, there is meta-knowledge that oversees the IIS. This category contains knowledge regarding the ability to explain system responses, to detect inconsistencies, and to restructure system knowledge. For example, Earth scientists will want to know why the system is responding the way that it is for a particular query. What is its justification? Is it the opinion of an established expert whose knowledge has been captured? Not only is this meta-knowledge, but the act of extracting the domain specific knowledge from the expert, coding it, and putting it into the system itself is also meta-knowledge (Cromp, 1990).

In the IIFS, the metadata for the object-oriented data management with related semantic data modeling has evolved into a knowledge-base with objects and relationships between objects being explicitly declared (Campbell et al., 1991). The meta-knowledge too has been recognized and dealt with explicitly as the "pre-existing" knowledge about the problem domain, the sensor device, and the interpretation of the sensor's measurements (Campbell et al., 1989). However, with the increased research that will naturally follow EOS, it is imperative that newly acquired knowledge (new meta-knowledge) be ingested and available to all in the scientific community (Short, Jr., 1991).

In the design and development of an IIS, the automation of meta-knowledge is essential. An IIS must recognize the limitations of its knowledge and gain new knowledge by interacting with the users that it is serving. To this end, meta-knowledge must be represented in a language that is high-level and robust yet has the appropriate primitives to integrate multi-paradigm software systems.

A Knowledge Representation Language for Meta-knowledge

Zarri (1990) proposed a "conceptual" knowledge representation language suited to the construction and use of intelligent information retrieval systems. This conceptual knowledge representation language exploits the organizational strength found in definitional hierarchies and the power realized in a theorem-prover with a unification algorithm. The components of the language are organized around a semantic predicate ("has", "produces", etc.) that identifies the basic type of situation to be described. The semantic predicates are frame-like in structure with "arguments" (objects) and "roles" (slots). The choice of semantic predicates is pragmatic and depends on the architecture of the system and on the problem domain, in particular the arguments and the roles of those arguments in an application. Roles can be categorized as descriptive (such as: SUBJECT, OBJECT, SOURCE, DESTINATION, etc.), binding (such as: COORDINATION, SPECIFICATION, ALTERNATIVE, ASSOCIATION, etc.), and causal (such as: CAUSE, MOTIVATION, CONFER, GOAL, etc.). As a conceptual unit, the semantic predicate can be further characterized by "determiners" (attributes), for example, location and temporality. Figure 1 is an example of how the conceptual knowledge representation language might be applied to remote sensing domain knowledge. It is a predicative conceptual unit (a predicative occurrence) having a semantic predicate "created_using", arguments such as "data_set_ssc150" and "CAMS", and determiners. The importance of this work is that it provides a conceptual base from which to study the inclusion of meta-knowledge into an IIS. Both the binding and the causal roles can be very useful toward this end; they can allow control strategies to be explicitly defined. An implementation of the proposed knowledge representation language would be a compromise between object-oriented and logic-oriented paradigms.

```
created_using    SUBJECT:          data _set_ssc150
                 OBJECT:           CAMS
                 SOURCE:           flight_7824
                 SPECIFICATION:    mission_request_8182
                 ALTERNATIVE:      data_set_ssc278
                 COORDINATION:     [5_percent_cloud_cover, 30000_feet_altitude]
                 MOTIVATION:       deforestation_study
                 [location: Mexico_Guatemala_border]
                 [date: 24_july_1990]
```

Figure 1. An example of a predicative occurrence.

The knowledge representation language described above allows the system designer to declare data, metadata, and meta-knowledge without knowing the details of its implementation. A preprocessor could then be used to produce an effective and efficient implementation of the design. Such a preprocessor could be either a meta-interpreter (Sterling & Beer, 1989) or a translator (Console & Rossi, 1989). The latter approach is being taken for several reasons. First, the knowledge representation language lends itself to this method. Second, Zaniolo (1984) demonstrated that object-oriented programming can be embedded in a logic programming language (PROLOG). Furthermore, today, the integration of object-oriented and logic-oriented paradigms is a robust and productive area of research (McCabe, 1992). Finally, since logic programming has already been used in metadata specification to make designs of semantic networks and frames into executable code that can be queried (Lopez and Saacks, 1992), it seems only natural to extend its use via a translator to implement the knowledge representation language.

FROG (Frames in PROLOG) is a logic programming language that combines frames, production rules, and PROLOG (Console & Rossi, 1989). In FROG each frame can contain either slots or production rules (with various kinds of inference strategies). Descriptive meta-knowledge on the relationships that exist between frames can be embedded in various kinds of links supported by FROG. Trigger links stipulate conditions under which a frame will be activated. Specialization links structure the hierarchy of frames. Associational links connect highly correlated frames. Alternative links suggest other possible hypotheses of solution to be considered when a frame cannot be instantiated. In addition to these links, FROG frames have knowledge components that can either be local production systems or prototypical descriptions. Control knowledge is vested in a "superframe," which is the top most frame in the frame hierarchy as stipulated by the specialization links.

Many of the concepts and ideas expressed in Zarri's conceptual knowledge representation language seem to have been implemented in FROG. In particular, the binding links of SPECIFICATION, ASSOCIATION, and ALTERNATIVE seem to match directly with the FROG links of specialization, associational, and alternative. The superframe allows the explicit specification of the control strategy and a separation from the knowledge components of the frames. The knowledge components allow the system designers to embed even more meta-knowledge in the form of prototypical descriptions or production systems. The knowledge-base itself is an object-oriented structure.

An Application

The classification of pixels in a data set obtained from aerial or satellite images is a difficult and time-consuming process. Rules used to classify regions in remotely sensed images are not universal truths. Human experts have developed heuristic knowledge that allows them to focus on those classification features that help refine the initial analysis of the image that might have been done by unsupervised training algorithms. In using unsupervised training, the data analyst specifies some parameters that the algorithm can use to determine statistical patterns that are inherent in the data. This is useful only if the classes that are produced can be manually interpreted. The interpretation depends on the expertise of the data analyst because the classes do not necessarily correspond directly to meaningful classifications such as water, crops, manmade objects, etc. The process involves the ingredients of data, metadata, and meta-knowledge, and can be used as a testbed for research ideas involved in the development of IIS.

During FY92, a small study was done at Stennis Space Center on a knowledge-based pixel classification approach using PROLOG as the vehicle to investigate the relationship between low and high resolution feature identification in Calibrated Airborne Multispectral Scanner (CAMS) data sets (Lopez et al., 1992). The goal was to be able to use knowledge in various forms to construct a system with the potential of changing the means by which it characterizes a given class of pixels (structuring and restructuring knowledge). Knowledge-based methods when used with statistical classifiers tend to improve the accuracy of the overall classification of pixels in an image (Short, Jr., 1991). Rules for image classification for this study were developed on the basis of the expert data analyst's knowledge of the numerical values produced by the statistical (maximum likelihood classification) unsupervised training. Knowledge obtained from this study is used below to demonstrate some of the constructs of FROG.

If a pixel class is to be identified as water, it will usually exhibit a low near-infrared reflectance. An expert data analyst's own interpretation of this previous statement might be that the mean in the red channel of the class is less than 40 and the near-infrared mean is less than 30. If this is realized by a pixel class it will "trigger" further investigation into whether or not the pixel class is indeed water. There are also both necessary and sufficient conditions for a pixel class to be water but if the pixel class meets the necessary conditions, then it does not have to meet the sufficient conditions to be interpreted as water. Furthermore, there can always be supplemental knowledge that can support the interpretation. This is particularly important when uncertainty factors are added to the "knowledge components". The constructs of FROG are used below to explicitly embed the meta-knowledge that has been discussed. Uncertainty factors have not been incorporated into this example.

```
frame_control(water_class,activation) :-
    knowledge_component(water_class,trigger),
    (     (knowledge_component(water_class,necessary);
          knowledge_component(water_class,sufficient)   ) +
    knowledge_component(water_class,supplementary),
    frame_control(water_class,specialization).

knowledge_component(water_class,trigger) :-
    slot(water_class).

slot(water_class) :-
    conditions(water_class).
```

```
conditions(water_class) :-
    implies(water_class,red_channel_mean_less_than_40),
    implies(water_class,near_infrared_channel_mean_less_than_30).

knowledge_component(water_class,necessary) :-
    slot(implies(water_class,red_to_green_ratio_is_0.4),
    slot(implies(water_class,red_to_near_infrared_is_0.1).

knowledge_component(water_class,sufficient) :-
    slot(implies(water_class,minimum_spectral_distance_from_water_classes));
    slot(implies(water_class,above_diagonal_and_left_in_green_near_infrared_plot)).

knowledge_component(water_class,supplementary) :-
    slot(contextual_information).

frame_control(water_class,specialization) :-
    frame_control(clear_water_class,activation);
    frame_control(muddy_water_class,activation).
```

The activation of the water_class frame succeeds if the trigger knowledge component can be instantiated and either the necessary or the sufficient knowledge component instantiated. As in standard PROLOG coding, the comma is used for the connective "and," and the semicolon is used for the connective "or." The plus symbol in FROG is an additive evidence combination operator and, if certainty factors were being used, would increase the certainty that the pixel class was water if the supplemental knowledge component was instantiated. This operator allows two knowledge components with knowledge from different sources leading to the same conclusion to be combined. Finally, a subframe is invoked for specialization.

Conclusion and Future Research Direction

In the past, NASA has just provided data to researchers and done little to capture into its archives the knowledge derived from the researcher's use of the data. If the acquired knowledge is to be unified and made available to the entire scientific community, then any future IIS will have to rely more heavily on meta-information. In particular, meta-knowledge will have to be recognized and explicitly coded into such systems. To support this effort, more research needs to be done on the application of Zarri's conceptual knowledge representation language to space systems such as EOSDIS. Hand in hand with this effort is the research that is needed in implementation languages. A logic programming language such as FROG holds great promise. However, it is safe to say that the search for meta-knowledge in IIS is just beginning.

References

Campbell, W., & Cromp, R. (1990). Evolution of an Intelligent Information Fusion System. *Photogrammetric Engineering and Remote Sensing*, 56(6), 867-870.

Campbell, W., Hill, S. & Cromp, R. (1989). Automatic labeling and characterization of objects using artificial neural networks. *Telematics and Informatics*, 6(3/4), 259-271.

Campbell, W. Roelofs, L. & Short, Jr., N. (1987). The development of a prototype intelligent user interface system for NASA's scientific database systems. *NASA Technical Memorandum 87821*, Greenbelt, MD.

Campbell, W., Short, Jr. N., Roelofs, L. & Dorfman, E. (1991). Using semantic data modeling techniques to organize an object-oriented database for extending the mass storage model. *Proceedings of the 42nd Congress of the International Astronautical Federation*, Montreal, Canada.

Console, L. and Rossi, G. (1989). Using PROLOG for building FROG, a hybrid knowledge representation system. *New Generation Computing*, 6(4), 361-388.

Cromp, R. (1990). Knowledge acquisition for spatial reasoning about satellite imagery. *Proceedings of the 4th International Symposium on Data Handing*, Zurich, Switzerland.

Cromp, R., Campbell, W., & Short, Jr., N. (1992). An Intelligent Information Fusion System for handling the archiving and querying of terabyte-sized spatial databases. To appear in the *American Institute of Physics Conference Proceedings*.

Dozier, J. (1990). Looking ahead to EOS: The Earth Observing System. *Computers in Physics*, 4(3), 248-259.

Goyal, P. (1989). Intelligent information systems: The concept of an intelligent document. *Information Systems*, 14(4), 351-358.

Kaula, R. & Ngwenyama, O. (1990). An approach to open intelligent information systems. *Information Systems*, 15(4), 489-496.

Kerschberg, L. (1990). Expert database systems: Knowledge/data management environments for intelligent information systems. *Information Systems*, 15(1), 151-160.

Kidder, T. (1981). *The soul of a new machine*. Boston, MA: Little Brown.

Lenat, D. & Guha, R. (1990). *Building large knowledge-based systems: Representation and inference in the CYC project*. Reading, MA: Addison-Wesley.

Lopez, A., Junkin, B. & L. McGregor. (1992). An investigation into the use of artificial intelligence in the classification of remotely sensed data. Under review.

Lopez, A. & Saacks, M. (1992). Logic programming and metadata specification. *Proceedings of the Goddard Conference on Space Applications of Artificial Intelligence*, Greenbelt, MD, 245-253.

March, S., & Kim, Y. (1989). Information resource management: A metadata perspective. *Journal of Management Information Systems*, 5, 3, 5-17.

McCabe, F. (1992). *Logic and Objects*. New York, NY: Prentice Hall.

McDonald, K., & Blake, D. (1991). Information management challenges of the EOS Data and Information System. *Proceedings Technical Papers ACSM-ASPRS Annual Conference*, Baltimore, MD, 258-267.

Short, Jr. N. (1991). A real-time expert system and neural network for the classification of remotely sensed data. *Proceedings Technical Papers ACSM-ASPRS Annual Conference*, Baltimore, MD, 406-418.

Sterling, L., & Beer, R. (1989). Metainterpreters for expert system construction. *Journal of Logic Programming*, 6(1/2), 163-178.

Zaniolo, C. (1984). Objected-oriented programming in PROLOG. *Proceedings of the IEEE International Symposium on Logic Programming*, Atlantic City, NJ, 265-270.

Zarri, G. (1990). A knowledge representation language for large knowledge bases and "intelligent" information retrieval systems. *Information Processing and Management*, 26(3), 349-370.