An Intelligent Heterogeneous Autonomous Database Architecture for Semantic Heterogeneity Support*

Doyle Weishar and Larry Kerschberg
dweishar@gmuvax2.gmu.edu, kersch@gmuvax2.gmu.edu

Department of Information Systems and Systems Engineering
School of Information Technology and Engineering
George Mason University, Fairfax, VA 22030

Abstract

Semantic heterogeneity in heterogeneous autonomous databases poses problems in instance matches, units conversion (value interpretation), contextual and structural mismatches, etc. In this work we examine some of the research issues in semantic heterogeneity and propose a novel architecture for resolving such problems. The approach involves the use of Artificial Intelligence tools and techniques to construct "domain models," that is data and knowledge representations of the constituent databases and an overall domain model of the semantic interactions among the databases. These domain models are represented as Knowledge Sources (KSs) in a blackboard architecture. This architecture lends itself to an opportunistic approach to query processing and goal-directed problem solving. We introduce the notion of Data/Knowledge packets as a means of supporting semantic heterogeneity, and show how an active and intelligent global thesaurus can be used to reformulate queries based on knowledge associated with terms and their usage in local databases.

1. Introduction

In this paper we deal with the problem of semantic heterogeneity in accessing heterogeneous autonomous databases. This topic has been discussed in a special issue of the *Data Engineering Bulletin* (June 1990, Vol.13, No.2) and in other forums. By semantic heterogeneity we mean that the meanings of objects, attributes and relationships represented in the component databases are inconsistent with respect to one another, even though they refer to the same "real-world" objects. Examples include problems in instance matches, units conversion (value interpretation), contextual and structural mismatches, etc.

In this work we examine some of the research issues in semantic heterogeneity and propose a novel architecture for resolving such problems. The approach involves the use of Artificial Intelligence tools and techniques to construct "domain models," that is data and knowledge representations of the constituent databases and an overall domain model of the semantic interactions among the databases. These domain models are represented as Knowledge Sources

---

* Appeared in Proceedings of the *IEEE Workshop on Interoperability in Multidatabase Systems*, Kyoto, Japan, April, 1991

(KSs) in a blackboard architecture. This architecture lends itself to an opportunistic approach to query processing and goal-directed problem solving.

2. Research Issues in Semantic Heterogeneity

2.1. System Autonomy

We feel that autonomous heterogeneous databases will become increasingly important, as more and more applications require access to diverse databases for problem solving. Because each node of the network will be autonomous, we envision that the local schemata will *evolve* over time. Therefore, those architectures that rely on a global schema will also have to evolve as their local schemata change. In addition, external user schemata will be affected. Change and evolution are important because real-world enterprises are in constant flux, so that database evolution is a natural requirement.

Techniques to capture and represent change and evolution will become increasingly important for autonomous heterogeneous database systems. Evolution may involve changes to the database structural specification, its constraints, and also its operations.

We believe that one technique to support evolution is to construct, for each database, an object-oriented *domain model* that characterizes the data, meta-data, constraints, and knowledge associated with the object types and the object (database) instances they refer to. These domain models (KSs) are comparable to the local schemata referred to in the literature. In addition, we propose constructing a global domain model that provides an understanding of the relationships among the objects in the KSs. Our approach is to construct an active and intelligent thesaurus that provides the semantic relationships among the objects in the local KSs.

As the local databases and KSs evolve, we expect to be able to reason about the impact on the global KS so as to effect appropriate changes at the global level.

2.2. Export Schemata Revisited

Export schemata in their present form provide the data structures and relationships that can be accessed by other sites. We feel this definition is insufficient for our knowledge-based approach. What is needed is a representation that expresses the semantics of the data and knowledge of that portion of the local KS that will be accessible to other sites. We suggest encapsulating the semantics of an object with its structural specification; we can then export not only object structural semantics but also object operational semantics in the form of integrity constraints, methods, rules and task-specific knowledge. In this way we export information regarding the usage of objects within the local database. We introduce the notion of a "Data/Knowledge Packet." Our approach is outlined in Section 3.4.

2.3. Cooperative Problem Solving

In conventional architectures the export schemata do not fully define the semantics of objects. Because there is at most a unidirectional, one-to-many interaction among the constituents of a federated architecture, this approach is incapable of discovering the meaning of objects. To overcome this deficiency, the proposed architecture provides a natural foundation for cooperative problem-solving and knowledge discovery. In our approach the semantics of an object may be derived from the local export data/knowledge packets or in cooperation with other KSs.

The knowledge sources can share the knowledge that is embedded in their "Data/Knowledge Packets" with other KSs on one or many blackboards; controlling KSs can use their strategic knowledge to integrate and focus the attention of the overall system. Thus, we have a sense of cooperation among all of the conceptual units.

2.4. Multiple Viewpoints

In an autonomous heterogeneous environment, each local site will characterize a viewpoint of its real-world objects. Each viewpoint will have its own functional orientation, that is, the uses of the data at that site, as well as a structural orientation, that is, how the data is organized for efficient query and transaction processing. One goal of our heterogeneous database architecture is to provide a coherent view of the entire database.

For example, for a federation of databases supporting a weapons deployment expert system there might a characteristics database, an object-oriented component database, and a logistics support database which characterizes the additional materiel needed to sustain a system in the field. Each supports a unique viewpoint of the system and it is the cooperation among the viewpoints that is essential for intelligent query processing.

An intelligent query processing capability is essential in resolving semantic ambiguities. These ambiguities arise because users pose queries in terms of high-level concepts that may not be understood by the local databases. However, by the cooperative problem-solving mechanism proposed above, the system can decompose user defined concepts into those understood by the existing KSs. If the system cannot resolve the semantic ambiguity, it will consult the user to obtain clarification and refinements of the query.

3. The Intelligent Heterogeneous Autonomous Database Architecture (InHead)

The InHead approach is to extend the state of the art in heterogeneous DBMS interface technology by integrating Artificial Intelligence (AI) problem-solving techniques with advanced semantic data modeling techniques. The approach draws upon the flexible and opportunistic problem-solving capability of blackboard architectures, and the expressive power of the Knowledge/Data Model (KDM) which allows both knowledge and data to be represented in a unified data knowledge representation.

In contrast to the sequential and hierarchical nature of standard interfaces to heterogeneous databases [e.g., Mermaid, Multibase and MRDSM], the InHead system incorporates an object-oriented Knowledge/Data Model, the KDM, and knowledge sources (KSs) possessing global and local domain expertise.

These KSs work together to provide users with *simultaneous, multiple* viewpoints of the system at varying levels of abstraction. One KS can be looking at a user's problem from a global perspective, while another can be viewing it in terms of a local database. In this way users can more fully determine system-wide data relationships.

Furthermore, in decomposing user queries, the opportunistic nature of the blackboard provides users with responses that are not only more complete (by KSs being activated on partial and incremental information), but reflect a deeper understanding of the user's desires. Also, it is envisioned that one KS will be a "User Model" and agent that contains a characterization of the user's intentions or goals during the query sessions.

A novel feature of InHead is a *global thesaurus* KS. The thesaurus addresses semantic heterogeneity issues in which data items may be similarly named, are related, are a subclass of, and/or are a superclass of data items located elsewhere within the system's databases. In this regard, the thesaurus plays the traditional role of data dictionary. What separates this approach from others is that the thesaurus takes on a new role: it ***actively*** works with users to reformulate queries based on knowledge associated with terms and their usage in the local databases. Figure 1 illustrates the initial architecture.

Conceptually, the system works as follows. Upon accepting a user query, the system controller consults the global thesaurus. If the solution to the query is obvious, the controller acts upon the

query by sending it to the appropriate database(s). If not, the controller posts the query on the blackboard. That enlists the aid of the KSs, which are domain experts over their respective DBMSs. The KSs cooperatively try to find a solution to the query. If no solution can be found, a request to the user for clarification or further information is generated.

Thus the InHead thesaurus is used to perform semantic query processing of the user's original request before submitting the reformulated query to the local databases for processing. The controller then conducts the necessary query translation and optimization, sends the query (or subqueries) to the appropriate database(s), integrates the query results (if required), and provides the answer to the user.

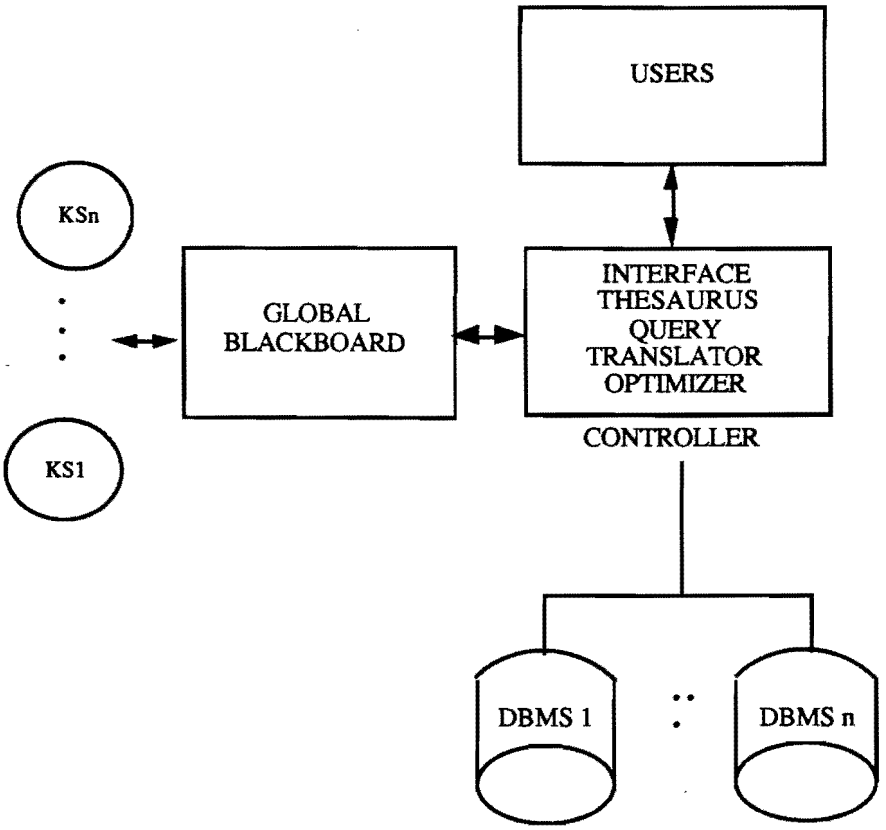

FIGURE 1: The Intelligent Heterogeneous Database (InHead) Architecture

3.1. Role of the Blackboard

The blackboard framework is regarded by many AI researchers as the most general and flexible knowledge system architecture. It offers expert system programming techniques that are difficult to achieve in other frameworks. Among them are:

- Dynamic control. At each step in the formation of the solution, a decision can be made as to how best to make inferences related to that step.

- Focus of attention. There is no rigidity with respect to what part of the emerging solution should be attended to next; for example, whether attention should go to an element at a low level of abstraction or at a high conceptual level.

- Flexibility of programming the control. Knowledge about how control should be applied in various domains can be codified in control rules or in complex control regimes.

• Modularity. Because the architecture is inherently modular (knowledge sources, blackboard levels, control structures, etc.) the design, testing and maintenance of the system simplified.

The blackboard framework is particularly well-suited to the class of problems possessing one or more of the following characteristics :

- The need to represent many specialized and distinct kinds of knowledge.
- The need to integrate disparate information.
- A natural domain hierarchy (or hierarchies).
- Having continuous data input (e.g., signal tracking).
- Having sparse knowledge/data.

The problem of supporting semantic heterogeneity in a system of heterogeneous autonomous databases exhibits many of these characteristics. There are many specialized sources of knowledge to be represented. For example, there is the need to represent knowledge of query decomposition techniques, individual database contents, individual database query formulation methods, individual database contents from a system-wide viewpoint (i.e., an active knowledge thesaurus of the databases' terms and of their usage), user intentions and preferences, and past performance of databases.

Inherent in the heterogeneous nature of the system considered is the need to integrate disparate information. The blackboard provides an organizational framework to hold, combine and interchange the fundamentally different kinds of information required to improve query specifications. This blackboard feature could be applied in other capacities such as conducting system-wide query optimization.

Furthermore, it is likely such a system would possess sparse knowledge/data. Thus, there could be cases in which uncertain knowledge or limited available data would make absolute determination of a solution impossible (i.e., queries in which entity and attribute ambiguities could not be fully resolved). In these cases the incremental problem-solving nature of blackboard systems still allows progress to be made.

3.2. Opportunistic Query Processing

The general strategy for processing queries in a heterogeneous environment is for a global controller to decompose a global query (made by a local database) into appropriate local subqueries, to supervise the execution of the subqueries, and to integrate the subquery results into an answer for the requesting database site.

An alternative strategy, appearing in the federated and interoperable approaches, is to have a single local query cause the generation of multiple queries to system databases without the aid of a global controller. These queries are based upon the local database's view of the external schemata of each of the other databases. Integration of results is left to the requesting database. In both cases these query processing procedures tend to be rigorously defined and sequential.

Query processing in the InHead system differs, however, in that there is not necessarily a predetermined sequence of operations for the query's solution. The blackboard control structure allows query processing to be performed in a manner that can change as system goals, subgoals and hypotheses change, that is, opportunistically. This is important in heterogeneous database systems where there are redundant and overlapping data. Sequential processing techniques may not provide the most computationally efficient solution. Because queries in InHead can be processed incrementally and opportunistically, processing can be halted when the control structure determines that the query has been satisfied.

3.3. The Active and Intelligent Global Thesaurus

Conventional thesaurus functions include meta-data management, data descriptions, descriptions of the relationships between the terms used to describe meta-data, as well as term definitions and descriptions. Thesauri become *active* when they are extended to provide functions such as: (1) validating and performing consistency checks on input to the thesaurus itself; (2) indexing and converting data values; (3) automatically translating queries using different variants of names; and (4) actively participating in on-line help (i.e., offer suggestions). By incorporating knowledge (in the form of heuristics and constraints attached to objects) into the thesaurus, we have not only made the thesaurus more active, but intelligent! Our thesaurus can act both as a repository of knowledge of data-item terms and of their usage, and as an active participant in formulating improved query specifications (i.e., by providing global data-item definitions and locations).

In essence, our active and intelligent global thesaurus provides the strategic problem-solving knowledge required to control semantic heterogeneity. The thesaurus is the ideal KS to resolve issues of semantic heterogeneity. Local database terms, concepts, relationships, constraints and operations that have semantic variants can be "encapsulated" into an abstract object type with sufficient domain knowledge to be able to translate and interpret the appropriate meaning for an object. The active thesaurus objects can invoke appropriate methods to present the proper views to the various local databases and to translate the global concept to the corresponding terms for the local KS.

The thesaurus could be used to incorporate newly discovered knowledge that might result from an examination and integration of KSs associated with each local site and with the overall problem-solving interactions among the sites in responding to semantically ambiguous queries. This knowledge could then be used as strategic case-based knowledge in future problem-solving exercises. The configuration management issues associated with knowledge evolution are topics of current research in InHead.

3.4. Object Encapsulation — Data/Knowledge Packets

Data/knowledge packets are proposed as a means of encapsulating object structure, relationships, operations, constraints, and rules into a meaningful unit, or packet. Data/knowledge packets are suitable for specifying abstract object types that at the global level provide a unified viewpoint, both structurally and operationally, of semantically heterogeneous objects.

In a distributed version of the InHead architecture one can envision each local database system communicating with its own expert system supporting the intelligent InHead features such as the blackboard and the thesaurus. This loosely-coupled expert database system could receive data/knowledge packets from the global KS and then incorporate this additional knowledge into its collection of KS. This might involve a transformation of the KDM specifications into the local "dialects" of the expert system shells.

Thus the local databases would remain the same, but the data/knowledge packets together with the local domain model KS could be used to perform semantic reconciliation at the local level first, before routing the query to the global knowledge source.

The data/knowledge packet is an important concept in dealing with the interoperability of not only data but also knowledge among heterogeneous, autonomous database and knowledgebase systems. Such architectures will be more common in future intelligent information systems.

3.5. An Example — The Artillery Movement Problem

For this example, suppose that we have an expert system whose task is to provision 10 M110 Howitzer Weapon Systems for departure to the Middle East in 5 days. This expert system is

written to interact with three primary databases: 1) a characteristics database, which describes the physical characteristics of the component parts of weapons systems; 2) a weapons systems database which describes the components of weapons systems; and 3) a logistics database which describes the logistics support required to sustain weapons systems in combat. Two secondary, but related databases are: a personnel database for crew requisitioning, and a ships database for obtaining space on seagoing vessels.

The expert system, which plays the role of the user in this example, has a task-oriented functional view of the problem as follows. Potentially semantic ambiguous terms are denoted in bold-face.

Overall Goal: Provision 10 M110 Howitzer Weapon Systems for departure to the Middle East in 5 days

Subgoals

1.0 Determine Availability of 10 M110 Howitzer Weapon Systems.

- 1.1 Determine the locations of such items, subject to constraint of being within 500 miles of Norfolk, Virginia.
- 1.2 Send requests for items to locations to hold for shipment.

2.0 Determine Availability of Logistics Support Units

- 2.1 Specialize camouflage to **desert** conditions.
- 2.2 Specialize radar to **desert** night vision.
- 2.3 Specialize rations to **high water content** rations.
- 2.4 Specialize clothing to lightweight, chemically resistant.

3.0 Determine Availability of Sealift Capability along the Eastern Seaboard.

- 3.1 Calculate total **weight** and **volume** for each system.
- 3.2 Provision **crews** for each system.
- 3.3 Assign **crews** and weapons to ships.
  - 3.3.1 Notify crews.
  - 3.3.2 Send shipment requisitions to sites holding weapons systems.

We now discuss several of the possible cases in which semantic heterogeneity is manifested in this system.

The first ambiguity occurs in the meaning of the word **miles**. The expert system may be assuming nautical miles while the logistics database might be assuming statute miles. When the logistics database sends its answers to the expert system, the data includes measurement units in its data/knowledge packets. The thesaurus is consulted for any unit translations. If ambiguity persists, the user can be consulted to provide appropriate definitions.

By focusing on Subgoal 1.0, we now look at the cooperative problem-solving and active database aspects of InHead. Suppose that in addition to the above databases, the system had an Army installation database, with attributes such as name, type, and location, and a database of Army units that described a unit's location, its weapons systems,its readiness status, and its deployment status. After the expert system retrieves all of the M110 locations, it begins to determine if these installations satisfy the 500 mile constraint. These locations have been returned by installation name and therefore, must be converted to grid coordinates to compute their distance in statute miles from Norfolk, VA. Thus the expert system places a knowledge packet on the blackboard in the

form of <OPERATION,OBJECT,RTN_VALUE>, for example, <SELECT,WPN_SYS.LOCN="FT BRAGG, NC",GRID>).

Because the blackboard allows KSs to see and understand system goals and subgoals, they can actively contribute to the process. In this case the installation database KS, understanding locations in longitude and latitude, helps by placing <nil,INST.LOCN="FT BRAGG, NC",39°N35°W> on the blackboard. The thesaurus KS knowing that there are several instances of LOCN invokes a method to translate long/lat to grid coordinates and places <nil,LOCN="FT BRAGG, NC",12344321> on the blackboard, which is then used by the expert system to compute the distance from Ft, Bragg, NC to Norfolk, VA.

Another example of a KS actively contributing to satisfy Subgoal 1.0 is found in the KS for the unit database. Noticing on the blackboard that M110s from Ft. Pickett, VA have been targeted for Middle East deployment, the KS checks that availability of M110s on Ft. Pickett. By querying its database the KS can determine the deployment status and readiness category of M110s on Ft. Pickett. If an M110 is already deployed or unfit for combat, the KS could place that information on the blackboard.

KS heuristics can also make their respective databases appear active. For example, reacting to an instruction to transfer 2 M110s from Ft. Pickett to the Middle East, the KS could alert the expert system that the action will cause the readiness category to fall below a certain threshold, resulting in a condition that violates a stated local database constraint.

The expert system might specify the task, "Provide logistic support for ten M110 howitzer systems with desert camouflage." But the logistics support database has an attribute for camouflage in terms of color combinations, rather than the term **desert**. The global thesaurus *knows* that desert colors are grey and brown, so that the semantic heterogeneity is handled easily. Also, if that information were not in the thesaurus, the expert system would engage a dialog with the user to define the term for the thesaurus.

Another instance of ambiguity surrounds the use of the term **crews**. Crews can be either operational crews or maintenance crews. In provisioning crews for each system, the system must know if one or both is needed. That type of information can be placed in the thesaurus. A default rule could be that operational crews take precedence over maintenance crews with the assumption that one maintenance crew can maintain several systems.

4. Conclusions

We feel that the InHead architecture provides some unique insights and features to overcome the problems associated with semantic heterogeneity in multiple heterogeneous autonomous databases.

The construction of a domain model allows the system's constituent databases to evolve over time, yet maintain system-wide semantic consistency. The use of the "Data/Knowledge Packet" allows the encapsulation of both structural and operational semantics in data/knowledge packets. And by using the blackboard paradigm, semantic ambiguity is reduced through incremental cooperative problem-solving techniques that support multiple viewpoints of database objects simultaneously.

5. References

W. Litwin, "An Overview of the Multidatabase System MRDSM," Proceedings of the ACM National Conference, Denver, Colorado, October 1985.

K. Kaufman, R. Michalski, and L. Kerschberg, "Mining for Knowledge in Data: Goals and General Description of the INLEN System," IJCAI-89 Workshop on Knowledge Discovery in Databases, Detroit MI, August 1989.

W.D. Potter and L. Kerschberg, "A Unified Approach to Modeling Knowledge and Data," IFIP WG 2.6 Working Conference on Knowledge and Data, September 1986.

J.M. Smith et al., "MULTIBASE -- Integrating Heterogeneous Distributed Database Systems," Proceedings of the 1981 National Computer Conference, Reston, VA: AFIPS Press, pp. 487-499.

M. Templeton et al., "Mermaid -- A Front-End to Distributed Heterogeneous Databases," Proceedings of the IEEE, Vol. 75, No. 5, May 1987.

D. Weishar and L. Kerschberg, "An Intelligent Interface for Query specification to Heterogeneous Database," NSF Workshop on Heterogeneous Database Systems, Northwestern University, Evanston IL, December 1989.