Larry Kerschberg
Professor and Director
Center for Information Systems Integration and Evolution
Department of Information and Software Engineering **(703) 993-1661**
http://cise.krl.gmu.edu/

George Mason University

kersch@gmu.edu
Fax: (703) 993-1638

To: Todd Carrico, DARPA ALP Program Manager
From: Larry Kerschberg, Project Manager for GMU Knowledge Rover Project
Subject: Final Report
Date: January 12, 1999

I. TECHNICAL STATUS

a. Demographic Information

Name: Larry Kerschberg, Hassan Gomaa, Sushil Jajodia and Ami Motro
Address: ISE Department, MSN 4A4
George Mason University, Fairfax, VA 22030-4444
Project Title: Knowledge Rovers: A Family of Intelligent Software Agents for Logistics for the Warrior.
Contract Number: N00600-96-D-3202.

Contract Agent: Ms. Down Van
Fleet and Industrial Supply Center Norfolk
Washington Detachment
Washington Navy Yard, Building 200, Fourth Floor
Washington, D.C. 20374-5014

Phone 202-433-2914
Fax 202-433-5414

Date of Report: January 12, 1999
Period Covered: 6/28/96 to 8/31/98 through three Delivery Orders:

Delivery Order 1:	6/28/96 - 6/30/97	$150,000
Delivery Order 2:	2/19/97 - 1/31/98	$250,000
Delivery Order 3:	2/27/98 - 8/31/98	$ 54,230
Total		$454,230

Please see note on last page regarding ***ongoing process*** to transfer funds from Delivery Order 3 to Delivery Order 2.

b. Accomplishments of the Project

The Knowledge Rover Project was divided into a number of tasks and we now present the accomplishments within each task.

Task 1: Knowledge Rover Architecture

PIs: Larry Kerschberg, Hassan Gomaa and Sushil Jajodia

The work in this task covered several research areas: 1) the specification and development of the Knowledge Rover Architecture, 2) research on distributed software architectures, and 3) research on semantic transaction processing.

1.1 Knowledge Rover Architecture

One major aspect of the project was the development of the Knowledge Rover Architecture which has the following characteristics:

a) **Concurrent & Distributed**. The paradigm used is that of multiple active agents who cooperate with each other in order to achieve the overall mission.
b) **Support Multiple Agents with different roles**. Agents have different roles in the architecture. The agents communicate with each other, as well as with user interface objects and server objects in a client/agent/server view of the world.
c) **Agent Configuration**. A **Knowledge Rover Architecture** with agents that are configured to specific mission objectives.
d) **Use object-oriented technology to promote evolution and reuse**. In particular, an Object Oriented Analysis and Modeling Method is being developed for active agent architectures. This uses the newly established Unified Modeling Language (UML) notation.
e) **Use Cases.** A major aspect of the object-oriented technology is that Use Case technology is being applied to develop scenarios of cooperating agents. The use cases are depicted on Object Collaboration and Sequence Diagrams.
f) **Domain Specific Agent Architectures** for knowledge rovers by developing a Logistics Agent Architecture. The goal is to be able to configure agents from the Logistics Agent Architecture to support specific logistics scenarios.
g) **Use Logistics Application Domain**. In particular a subset of the Rapid Supply (DPSC) application, as a proof-of-concept for the ideas.

Results Obtained in Knowledge Rover Architecture.

A demonstration prototype of a distributed software architecture for a Logistics Agent Architecture was developed, based on a subset of the procurement functionality required by DPSC. Heterogeneous Windows NT / Unix platforms were used, experimenting with Java, RMI (Remote Method Invocation), a CORBA object broker, with JDBC interfacing to relational databases on both Unix and Windows/NT platforms. The user interface is provided in the form of a WWW browser.

The demonstration prototype runs in a distributed heterogeneous environment using five nodes communicating via Java RMI. There are two NT client workstations, agents and web server run on a Solaris server, and two application servers on NT servers using ACCESS databases. Java applets are downloaded from the web server to execute on the GUI client nodes.

The Knowledge Rover Architecture is described in two book chapters, one technical report and four presentations:

Published Papers

[1] L. Kerschberg, "Knowledge Rovers: Cooperative Intelligent Agent Support for Enterprise Information Architectures," In *Cooperative Information Agents*, vol. 1202, *Lecture Notes in Artificial Intelligence*, P. Kandzia and M. Klusch, Eds. Berlin: Springer-Verlag, 1997, pp. 79-100.

[2] L. Kerschberg, "The Role of Intelligent Agents in Advanced Information Systems," In *Advanced in Databases*, vol. 1271, *Lecture Notes in Computer Science*, C. Small, P. Douglas, R. Johnson, P. King, and N. Martin, Eds. London: Springer-Verlag, 1997, pp. 1-22.

Technical Report

H. Gomaa , "Knowledge Rover Architecture", GMU Technical Report, September 1997, Revised August 1998.

Presentations

1. H. Gomaa and L. Kerschberg, "Knowledge Rover Architecture", September 1997.
2. H. Gomaa, "Agent Based Electronic Commerce System", August 1998.
3. L. Kerschberg, Keynote Address, First International Workshop on Cooperative Information Agents, Kiel, Germany, February 1997.
4. L. Kerschberg, Keynote Address, BNCOD, British National Conference on Databases, Birkbeck College, University of London, July 1997.

1.2 Distributed and Reusable Software Architectures

In this research, a reuse-oriented perspective is taken to designing and implementing configurable distributed applications. An application domain is defined as a family of systems that have some features in common and others that differentiate them. During domain engineering, reusable specifications, architectures and component types are developed, which capture the similarities and variations of the family of systems that compose the application domain. Target systems are generated by tailoring the reusable specification and architecture given the requirements of the target system, and configuring a target system based on the tailored architecture. This research has investigated methods and tools for the development and evolution of reusable software architectures.

This research is described in five papers:

Published Papers

1. H. Gomaa and G. Farrukh, "An Approach for Generating Executable Distributed Applications from Reusable Software Architectures", Proceedings IEEE International Conference on the Engineering of Complex Computer Systems, Montreal, October 1996.
2. H. Gomaa and G. Farrukh, "A Software Engineering Environment for Configuring Distributed Applications from Reusable Software Architectures", Proceedings IEEE International Workshop on Software Technology and Practice, London, June 1997.

3. H. Gomaa, “Software Design Methods for Distributed Applications”, Virtual Roundtable on Software Engineering for Parallel and Distributed Systems, IEEE Concurrency, July-September, 1997.
4. H. Gomaa and G. Farrukh, “Automated Configuration of Distributed Applications from Reusable Software Architectures”, Proceedings IEEE International Conference on Automated Software Engineering, Lake Tahoe, November 1997.
5. H. Gomaa and G. Farrukh, “Composition of Software Architectures from Reusable Architecture Patterns, Proceedings IEEE International Workshop on Software Architectures, Orlando, Florida, November 1998.

Doctoral Dissertation

Ghulam Farrukh, “A Method and Software Engineering Environment for Configuring Applications from Reusable Specifications and Architectures”, Defended Spring 1998.

1.3 Semantics-Based Transaction Processing: Satisfying Conflicting Objectives

The traditional correctness criteria of serializability forces database designers into tradeoffs among design objectives. For example, in multidatabases, the designer balances the objectives of local design and execution autonomy, decentralized management of global transactions, maintenance of global integrity constraints, and execution history correctness.

The last objective is typically assessed with respect to some variant of conflict serializability. Switching to a semantics-based perspective of correctness can greatly reduce the conflict between the remaining objectives.

In the case of multidatabases, the conflict can be entirely avoided for certain applications. We have successfully applied the semantics-based perspective in three distinct application areas: multidatabases, secure multilevel databases, and long duration transactions. Additionally, the method holds promise for such areas as database recovery and survivability. The cost of the semantics-based approach is additional off-line transaction analysis early in the lifecycle of a system; however, this up-front cost is amortized over the numerous transaction invocations during the system’s lifetime.

Publications

1. Elisa Bertino, Sushil Jajodia, Luigi V. Mancini, Indrajit Ray: Advanced Transaction Processing in Multilevel Secure File Stores. *Transactions on Knowledge and Data Engineering 10(1)*: 120-135 (1998)

2. Paul Ammann, Sushil Jajodia, Indrakshi Ray: Semantic-Based Decomposition of Transactions. *Advanced Transaction Models and Architectures* , Kluwer Academic Publishers, 1997

3. Sushil Jajodia, Indrakshi Ray, Paul Ammann: Implementing Semantic-Based Decomposition of Transactions. *CAiSE 1997*: 75-88.

4. Paul Ammann, Sushil Jajodia, and Indrakshi Ray, “Applying Formal Methods to Semantic -Based Decomposition of Transactions, *ACM Transactions on Database Systems*, Vol 22, No. 2, June 1997, pp. 215-254.

5. S. Jajodia and L. Kerschberg, “Advanced Transaction Models and Architectures,” . Norwall, MA: Kluwer Academic Publishers, 1997.

Task 2: Specific Knowledge Rover Agents

PI: Larry Kerschberg

2.1 Optimal Constraint Decompositions and the Active View Agent

We studied the problem of deriving the *optimal decomposition* of global integrity constraints in a distributed database into a set of local constraints that will serve as a conservative approximation, i.e., if, during an update at site i, the local constraint C_i is satisfied, no global constraint checking would be necessary, thus saving communication costs.

This is important to Logistics application where the systems are highly distributed, there may be hundreds of depots and staging areas, and it is of utmost importance that local sites be able to update their inventories independently of other sites, while knowing that they are not violating global constraints. If such constraints are violated, however, we present techniques to allocate *constraint-based resources* among neighboring sites, to *rebalance* the constraints so as to comply with the global constraints.

First, a generic optimization framework for all types of constraints is introduced for four possible scenarios, classified by: 1) whether the decomposition is made at design time versus update time, when the database instance is known, and 2) whether all versus a subset of distributed sites are involved, assuming for the latter case that the local constraints for the remaining sites are fixed.

Second, constraints which are conjunctions of linear inequalities or equations over reals are considered. It is shown that, under certain local uniformity assumptions, decompositions that maximize the probability of not violating local constraints can be effectively computed. To do that, a machinery of parametric (i.e., in terms of constraint coefficients) characterization of safe decompositions is developed, and it is shown how to analytically express the probability functions in terms of parameters, so that the optimal decomposition problem can be formulated as a standard one of mathematical programming.

Third, an algorithmic framework to solve the resulting optimization problems is developed, based on the combination of local and global search techniques, and an optimization engine for a specific type of decompositions has been implemented. Experiments run for this case suggest that the approach is feasible and scalable.

With regard to the Active View Agent (AVA), we have used the above results to develop a protocol which satisfies all properties (local and global consistency, update soundness and completeness, and re-decomposition optimality). We have also solved the “view-selection problem” because the design and operation of an active view is based on in two orthogonal problems: view-selection and constraint decomposition.

The AVA allows the user to: 1) define a materialized view of data on distributed heterogeneous databases, specifying the conditions under which that view should be updated, for example, when inventory amounts fall below a certain level, or when a possible congestion situation develops at a port, and 2) formulate a set of rules to be executed should those conditions by triggered. We use the theory developed to decompose the view into queries, alerters, and polling agents against

the underlying data sources. Optimization techniques are used to determine the optimal polling times based on quality-of-service criteria.

The AVA currently runs on Windows95 and NT, and we have ported the AVA proof-of-concept prototype to the WWW by using Java. The ODBC connection has been completed, and an ACCESS database can now be accessed using the web.

Publications

Doctoral Dissertation

Samuel Varas, "On Optimal Constraint Decomposition, Monitoring and Management in Distributed Environments," George Mason University, Fairfax, VA, August 1998.

Journal Paper

Len Seligman and Larry Kerschberg, "A Mediator for Approximate Consistency: Supporting "Good Enough" Materialized Views," *Journal of Intelligent Information Systems*, vol. 8, pp. 203 - 225, 1997.

Technical Report

Alex Brodsky, Larry Kerschberg, and Samuel Varas, "On Optimal Constraint Management in Distributed Databases," Technical Report, ISE Department, George Mason University, Fairfax, VA, 1998. (Submitted for publication).

2.2 InfoFed Multidatabase System

InfoFed is a prototype system that integrates data from diverse information sources. This system is unique in that it handles what we term *dynamic metadata*, that is *data about the data* being retrieved. This dynamic metadata is important in assessing the quality of the retrieved information as well as the reliability of the source. Examples of the dynamic metadata include the name of the data source, the units of measure, e.g., British Pounds, Euro, Japanese Yen, any transformations or processing of the data, the names of persons (curators) who oversee and manage the data, etc. In this way we can trace the *heritage* of the data.

We have implemented some "tailor-made" *Internet Rovers* with InfoFed, to seek out information contained not only in relational databases, but also at Web Sites with proprietary interfaces. In particular, built the GMU Shipguide Rover for General Research Corporation International (GRCI). The GMU Shipguide Rover accesses the Electronic Shipping Guide Web Page which has a point-and-click interface showing the disposition of shipping vessels and their location in ports or on the water. The GMU Shipguide Rover retrieves this up-to-the-minute information on ships into several tables for processing by InfoFED. Once the data is cached within InfoFED, the query processor can answer queries that would be very difficult to answer by using the Electronic Shipping Guide Web Page.

Publications

Wiput Phijaisanit, "Dynamic Meta-data Support for Information Integration and Sharing Across Heterogeneous Databases," Doctoral Dissertation, George Mason University, 1997.

2.3 Agents for Electronic Commerce

As a collaboration with AMS, we developed a technical report describing GMU's view of the role of intelligent agents in supporting the AMS's Electronic Marketspace concept. We surveyed the area of e-commerce and e-business and developed an agent-based architecture. This report was used within our Knowledge Rover Architecture task to build our prototype.

Publications

Larry Kerschberg and Sonali Banerjee, "Electronic Business: An Agent-based Approach," Technical Report, 1997.

Task 3: Information Integration and Information Quality

PI: Ami Motro

Our research under this contract focused on two related areas:

1. **Information Integration**: The integration of information sources under conditions of both semantic heterogeneity and data inconsistency.
2. **Information Quality**: The estimation of the quality of information sources and the answers that they provide in response to arbitrary queries.

In the first area we have achieved the following results:

1. **Conceptual Work**. We formulated a new model for database integration, called multiplex. Some of the prominent features of Multiplex are:
   - *Heterogeneity*: The integration model of Multiplex (user's side) is relational; the only limit on the type of information providers is that they should communicate their results in tabular form (providers may be relational, object-oriented, WAIS, etc.)
   - *Quick, ad-hoc reconfiguration*: It is possible to add and remove information sources quickly by editing a simple "mapping file".
   - *No need to completely understand the contributing information systems*: It is possible to map only the information needed from each source.
   - *Dynamic adjustment of queries*: Multiplex does not reject queries when some of the information they target is unavailable (e.g., when some information providers are "down"). Instead, it adjusts the queries automatically (i.e., answers them "as best as possible").
   - *Inconsistency resolution*: when information sources "overlap", providing inconsistent information, Multiplex integrates their answers. Inconsistency resolution is based on a powerful set of strategies provided by the global database designer.

2. **Implementation and Experimentation**.
   - In early 1997 we completed a pilot version of Multiplex. This version had only limited inconsistency resolution capabilities. It was tested in several applications including project Maria at GRCI.
   - In late 1998 we completed the implementation of a full version of Multiplex (which we refer to as Multiplex Ver. 2.0). This version involved total reprogramming, including new query decomposition algorithms, a full solution to inconsistency resolution, and a cooperative user interface for assisting users in the construction of global queries.

3. **Publication**. The work is described in three documents:

- A. Motro. *A Formal Framework for Integrating Inconsistent Answers from Multiple Information Sources*. Technical Report ISSE-TR-93-106. Department of Information and Software Systems Engineering, George Mason University. Fairfax, Virginia. October 1993**.**
- A. Motro. Multiplex: *A Formal Model for Multidatabases and Its Implementation*. Technical Report ISSE-TR-95-103. Department of Information and Software Systems Engineering, George Mason University. Fairfax, Virginia. March 1995 (revised March 1998). Submitted for publication.
- P. Anokhin and A. Motro. *Multiplex: A Data Integration System that Manages both Intensional and Extensional Inconsistencies*. In preparation.

In the second area, ***information quality***, we have achieved the following results.

1. **Conceptual Work**. We formulated a new approach for evaluating the quality of information sources and for deriving the quality of arbitrary subsets of sources (i.e., information targeted by queries). In particular:

    - *Quality measures*: We proposed a standard for specifying the quality of databases, which is based on the dual concepts of data soundness and data completeness.
    - *Quality inference*: We extended the relational model of data by associating a quality specification with each relation instance, and by extending its algebra to calculate the quality specifications of derived relation instances. This provides a method for calculating the quality of answers to arbitrary queries from the overall quality specification of the database.
    - *Initial quality estimation*: We developed practical methods for estimating the initial quality specifications of given database.

2. **Implementation and Experimentation.** We implemented a pilot system, that implements the techniques described above, and we conducted laboratory experiments (using synthesized data) to demonstrate the validity of our methods.
3. **Publications**. the results of this work have been described in a doctoral dissertation and three papers:

- Igor Rakov. *Data Quality and Its Use for Reconciling Inconsistencies in Multidatabase Environments*, Ph.D. Dissertation, George Mason University, May 1998.
- A. Motro and I. Rakov. Not All Answers Are Equally Good: Estimating the Quality of Database Answers. In *Flexible Query-Answering Systems* (T. Andreasen, H. Christiansen, and H.L. Larsen, Editors). Kluwer Academic Publishers, 1997, pp. 1-21.
- A. Motro and I. Rakov. Estimating the Quality of Data in Relational Databases. *In Proceedings of the 1996 Conference on Information Quality*, Cambridge, Massachusetts, October 1996, pp. 94-106.
- A. Motro and I. Rakov. Estimating the Quality of Databases. In *Proceedings of FQAS 98: Third International Conference on Flexible Query Answering Systems* (T. Andreasen, H. Christiansen, and H.L. Larsen, Editors), Roskilde, Denmark, May 1998. Lecture Notes in Artificial Intelligence No. 1495, Springer-Verlag, Berlin, Germany, pp. 298-307.

Task 4: Technology Transition

The GMU team worked with AMS, Global InfoTek, and GRCI to transition Knowledge Rover technology into their environments.

AMS and GMU: We supported AMS on the Rapid Supply project for DPSC, and produced a technical report on the role of agents for their concept of the Electronic Marketspace. We used that application domain as the example for the knowledge rover architecture prototype described in Task 1 above.

Global InfoTek and GMU: GMU and GITI have signed an MOU for future cooperation, and GITI is interested in incorporating Multiplex and Active Views into their Knowledge Broker. We collaborated with GITI in the Beacon Proposal, submitted to the Command Post of the Future (CPoF) Program; the proposal was deemed selectable but has not been funded at this time.

GRCI and GMU worked together to embed GMU's Multiplex and InfoFed prototypes into their MARIA rover architecture.

Task 5: Knowledge Rover Laboratory and Other Direct Costs.

The Knowledge Rover Laboratory for Logistics and Commerce is being used extensively to prototype our software, in both the Windows NT environment and Solaris, on our DELL 4300 Server and our Sun Ultra 1 Server, respectively.

The Knowledge Rover Architecture prototype was developed using these facilities, as was Multiplex and InfoFed.

The Home Pages for the Center and the Knowledge Rover Research Group are respectively: http://cise.krl.gmu.edu/ and http://cise.krl.gmu.edu/KRG/

e. Staff Hour Expenditure Report.

Contract Ceiling:
Hours funded:
Hours Expended This Reporting Period:
Hours Expended Since Inception:

Note: We have been reporting our time based on the attached spreadsheet.

II. FINANCIAL

See Attached Excel Workbook

Please Note! Delivery Order 3 has $17,280 in unspent funds while Delivery Order 2 is negative by $17, 448. We have been working with our contract officer to increase Delivery Order 2 by $17,280 and decreasing Delivery Order 3 by the same amount. This process is still underway and has not been completed.