Toward Inquiry-Based Education Through Interacting Software Agents

Daniel E. Atkins
William P. Birmingham
Edmund H. Durfee
Eric J. Glover
Tracy Mullen
Elke A. Rundensteiner
Elliot Soloway
José M. Vidal
Raven Wallace
Michael P. Wellman*
University of Michigan

In the University of Michigan Digital Library, interacting software agents cooperate and compete within a virtual information economy to provide library services to students, researchers, and educators.

Providing true access to the human record means offering relevant information without prohibitive search time or an overwhelming choice among sources. Conventional libraries provide such access through two mechanisms: information organization and librarian services. Librarians themselves often rely on services like information systems or bibliographic databases to do their jobs.

Digital libraries must likewise provide organizational schemes and a wide variety of services. Most observers focus on the vast amount of information digital libraries will offer, delivered in new and interesting ways. However, we believe it is the bounty of services that will ultimately demonstrate the potential of digital libraries.

The University of Michigan Digital Library (UMDL) project[1] is creating an infrastructure for rendering library services over a digital network. When fully developed, the UMDL will provide a wealth of information sources and library services. Of course, we cannot anticipate all the services that will eventually constitute a digital library. We therefore designed the UMDL to let third-party developers expand the library with new services and collections.

We are deploying the UMDL in three arenas: secondary-school science classrooms, the University of Michigan library, and space-science laboratories. Computer skills, information demands, and level of subject knowledge vary greatly among these user populations. Addressing the needs of high school students within a general-purpose digital library particularly stresses the flexibility of our underlying architecture. The UMDL must support services quite distinct from those that other digital libraries and the World Wide Web offer.

Many researchers and policy groups argue that students should engage in sustained inquiry to develop an in-depth understanding of science. Digital libraries provide an outstanding opportunity to vitalize science education in public schools through inquiry-based education. However, we must avoid the inflated expectations typical of technology in the schools. Technology is only one element of a complex educational environment. Students, teachers, and curriculum planners must work together for a digital classroom library to succeed.

We are addressing the UMDL's ambitious scale and heterogeneity requirements by designing an open, distributed environment for interacting software agents. Features such as automated team formation, information search-space structuring, and market-based resource allocation help coordinate agent activities that provide library services. We are deploying the UMDL in Ann Arbor high schools.

---

*Authors are listed alphabetically.

DISTRIBUTED AGENT ARCHITECTURE

Because digital-library technology is changing rapidly, user interfaces, search engines, and the structure of information sources must accommodate future innovations. Rather than adopt specific standards, we require the UMDL architecture to perform generic management operations, such as allocating resources and brokering connections. For instance, a language and protocol for communicating informational or processing capabilities and interests connects users and collections appropriately. However, determining how they interact to accomplish their task is beyond our architecture's scope.

Distributing tasks to numerous specialized, fine-grained modules promotes modularity, flexibility, and incrementality. It lets new services come and go without disturbing the overall system. We call these modules *agents*, emphasizing their local knowledge about specific tasks and their autonomy. Limiting the complexity of an individual agent simplifies control, promotes reusability, and provides a framework for tackling interoperability problems. Each agent performs a highly specialized library task and has a generic communication interface. This combination lets an agent apply specialized task competence to a wide variety of situations with other agents.

For example, an agent could generate synonyms for specified query terms and thereby produce variants likely to unearth relevant documents. Alternatively, an agent could use synonyms to assess how well some text matches an already formulated query. Encapsulating a general synonym service within a specialized thesaurus agent provides component functionality without committing to how it's employed systemwide.

Agent types

Figure 1 depicts the three classes of agents populating the UMDL: user interface agents, mediator agents, and collection interface agents. *User interface agents* (UIAs) manage the interface that connects human users to UMDL resources. Among other things, UIAs, perhaps with assistance from other agents,

- express user queries in a form that search agents can interpret,
- maintain user profiles based on specified, default, and inferred user characteristics,
- customize presentation of query results, and
- manage the user's resources available for fee-for-service activities.

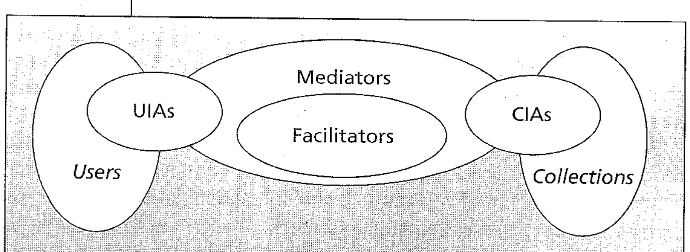

Figure 1. Three agent types populate the University of Michigan Digital Library, performing a variety of specialized tasks.

Mediator agents, which come in many types, provide intermediate information services.[2] In the UMDL, mediators deal exclusively with other software agents, rather than end users or collections. They perform such functions as

- directing a query from a UIA to a collection,
- monitoring query progress,
- transmitting results,
- translating formats, and
- bookkeeping.

A subclass of mediators, called *facilitators*, exists expressly to team up other agents to accomplish a given task.

Collection interface agents (CIAs) manage the UMDL interface for collections, which are defined bodies of library content. Among other communication tasks, the CIA publishes the contents and capabilities of a collection in the registry (described below).

The agent architecture lets us develop specialized capabilities and add them to the UMDL as needed. For example, through new UIAs we can customize interfaces to user classes, rather than to collections or access mechanisms. These UIAs, in turn, can access any mediator services available in the system.

Agent teams

Complex UMDL tasks require the coordination of multiple specialized agents working together on behalf of users and collection providers. To form teams, agents must be able to describe their capabilities to each other in ways all can understand.

LEVELS OF AGENT COMMUNICATION. UMDL agents communicate at three distinct levels of abstraction. At the lowest level, agents employ network protocols such as TCP/IP to transmit messages among themselves. Task-specific protocols dictate how the agents interpret and process these messages. For example, agents could use SQL to convey a request to perform a data-retrieval task. UMDL generally doesn't restrict task-specific protocols: Whoever designs and introduces the agents can freely choose the language(s) those agents speak.

Of course, agents are more likely to be used frequently if they communicate in widely adopted languages. In particular, a desire for broad interoperability provides an incentive to support standards like Z39.50, which libraries often use. This increases the scope of collections accessible to an agent posing a given query. While standardization has significant benefits, and many UMDL agents do use Z39.50, it is not a requirement for joining UMDL.

A specialized agent's capabilities will remain untapped unless it makes its abilities and location known and participates in team formation. We thus defined special protocols for the team formation and negotiation tasks, which all UMDL agents share. These UMDL protocols represent the third level of abstraction in agent communication.

CONSPECTUS LANGUAGE. UMDL agents are defined by the information content they can deliver, the information services they can render, or both. To participate in UMDL protocols, agents need a language for describing these capabilities. Agents describe what they can contribute to an agent team and what their limitations are in the conspectus language (CL). Facilitators can also use CL to (perhaps partially) describe capabilities required for participation on a team. CL thus serves as a language for both disclosing and querying about abilities.

To ascertain a message's intent, UMDL protocols adopted a flexible notion of message types, patterned after KQML.[3] UMDL message types, the equivalent of KQML "performatives," correspond to high-level communication acts. For example, messages intended to inform are of type Tell, and the purpose of Ask messages is to elicit information. A message can contain CL expressions, with the message type conveying what the recipient should do with the supplied content. UMDL protocols define a small number of standard message types that all agents should be able to interpret and process.

REGISTRY AGENT. We designed the UMDL protocols so that agents advertise themselves and find each other on the basis of capabilities. Rather than have every agent maintain models of all others and periodically broadcast its descriptions to every other agent, we designated a registry agent. The registry is special in several respects. First, on inception, agents know how to access the registry, thus avoiding the bootstrapping problem. Second, all agents can communicate with the registry using the UMDL protocols, as further detailed below. Third, the registry provides its services for a static price (currently free) to avoid the need to negotiate. Negotiation with the registry could lead to deadlock, since the registry contains the information identifying which agents can facilitate negotiation.

The registry agent maintains a database of all agents in the UMDL system, including descriptions of their content and capabilities. It updates the database with descriptions expressed in CL. The registry agent collects descriptions that specify the following types of characteristics:

- identification (such as name, location, and type),
- content (broad topic, audience level, language, and so on),
- capability (search engine(s) supported, translation facilities, name authority services, and so forth),
- interface (for example, task-specific languages and resource requirements), and
- economic (pricing methods, standing offers, and negotiation protocols, for example).

One simple yet representative example of a CL description is that which characterizes an author index agent (Figure 2). The agent belongs to a class of UMDL agents that search across information sources without executing the search request in each. Its CL description specifies its type and describes its service in terms of what interactions it supports. The `<Capability>` field states that the agent accepts queries with a specific author $A as a bound input parameter. It then returns the associated CIAs ($U) for all collections in which the author appears.[4] It does not, however, accept requests of the reverse order—asking for authors associated with a particular collection.

The registry agent communicates using UMDL protocols, translating incoming requests into queries on the registry database. Since this service's availability and fault tolerance are critical, we employed a persistent implementation of the registry database. An SQL server provides the basic properties of consistency, concurrency, and recovery, and supports high throughput of concurrent agent requests. Our second-generation registry agent, under development, uses a more powerful distributed, open architecture. We are implementing the distributed registry using commercial database technology. Replication servers support a powerful distributed search paradigm that, while robust and scalable, is transparent to the rest of the UMDL.

The preliminary version of the distributed agent architecture contains about a hundred CIAs and spawns a UIA for each active user. In addition to the registry, we have implemented several other mediator agent types. We describe three of these—the query planner, the market facilitator, and the remora—later on.

SEARCH TYPES

In any UMDL context, the core task is to find the right combination of information and services to satisfy the participants' objectives. This could mean answering a user's question, finding customers for a publisher's content, or applying a sequence of format-translation services. In these cases, the fundamental activity is searching for useful content or services using minimal effort, time, and money.

Within UMDL, searching takes several forms. Once a user's UIA contacts a collection's CIA, the search concerns documents from the collection that satisfy the user's specifications. This level of search is a *collection search*. Before collection search takes place, however, the UIA must identify appropriate collections on the basis of how agents describe themselves in conspectus language. This is a *conspectus search*. Finding mediators with particular capabilities is another form of conspectus search. UMDL agents interleave these various types of search to accomplish more complex tasks.

Collection search

The UMDL architecture supports arbitrary types of collections and search engines by encapsulating them using CIAs. Thus we can accommodate even those collections that require custom browsers, such as the Blue-Skies weather service.[5] We extended the class of collections accessible through more standard retrieval protocols by developing

```
< CL description {
 <Agent_ID AID_777>
 <Agent_type Author_index>
 <Capability
   <Author *$A> <CIA $U*> >
 <Task_Language SQL>
 <Content
   <Broad_Topics 'SCIENCES'>
   <Last_updated 12.31.1995>
   <Frequency_of_update end_of_year> >
 <Pricing fixed (1-bibliobuck-per-search) >
 <Content_Language {English,German,Latin}> }
```

Figure 2. Conspectus language description of an author index agent.

Z39.50 interfaces for Mirlyn, FTL, and WAIS. (Mirlyn provides access to the University of Michigan library catalog and several abstracting and indexing databases, while FTL is a UMDL-specific search engine.) We are also investigating structuring techniques that search across complex objects such as SGML (Standard Generalized Markup Language) documents.[6]

There are two modes for interacting with collections: searching and browsing. In the first, the UIA knows which collection to access, perhaps because of a prior conspectus search. In this case, the user connects directly to that collection's CIA and uses native retrieval facilities. Alternately, the UIA could conduct a search across collections. An information fusion agent then organizes the results, combining or ranking the retrieved information for presentation to the user.

Conspectus search

Conspectus search seeks to connect content providers and consumers on the basis of agents' needs and capabilities as described in conspectus language. Typical tasks include locating appropriate collections, identifying a particular work's authors, and determining the cheapest way to access certain information. This generally involves several intermediate tasks, including other conspectus searches. For example, while looking for appropriate collections, a UIA might conduct a conspectus search for a thesaurus agent.

UMDL agents formulate conspectus search tasks in terms of content or services sought and search processes by which to find them. A particular conspectus search task's description includes

- conspectus language specifications for the content or capabilities sought,
- deal parameters (such as acceptable cost ranges and delivery constraints),
- search-effort parameters (allowable search time, number of sources, and so forth), and
- search modification guidelines (for example, preferences toward using particular agents and trade-offs among the other parameters).

A conspectus search returns a set of agent deals. Each deal represents an agent's offer to provide the desired services or content, and the terms of the offer. The initiating agent can accept deals on the basis of criteria such as price and reputation. It then works with the chosen agent(s) in a task-specific language. If no deals are acceptable, the initiating agent can reinitiate conspectus search to find alternative deals.

Conspectus searches can be as simple as retrieving relevant entries from the registry as a direct result of the user's request. Other searches require the combined abilities of a team of agents to reformulate the request and balance thoroughness against cost. A query-planning mediator coordinates this kind of search.

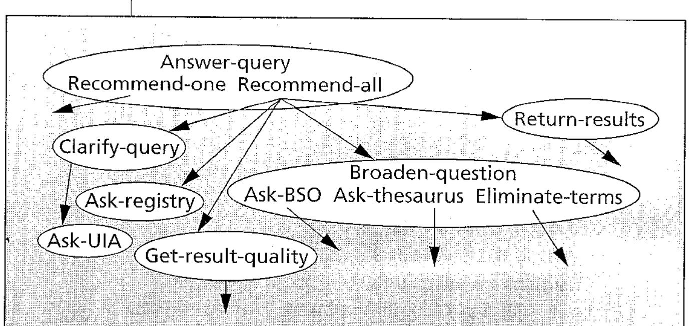

Figure 3. The query-planner procedure can be elaborated to build a team of agents for accomplishing search tasks.

Query-planning mediators

Agents capable of accomplishing conspectus search tasks are classified as task planners. As noted above, a task planner might require additional information or services from other agents to accomplish its task. Query-planning mediators, a subclass of task-planning agents, specifically tackle conspectus search tasks that seek collections to satisfy a query. Our initial query planner uses the UM version of the Procedural Reasoning System (UM-PRS), which provides facilities for flexible procedure specification and execution.[7] Our UM-PRS task planners communicate using UMDL protocols. They are goal-driven, persistent, independent, and proactive.

Query-planning mediators embody specialized knowledge about how to seek out information sources in response to a user's query. Based on interviews with librarians, these procedures specify the control flow among various resources within the UMDL. Depending on user characteristics, library load, and desired completeness and timeliness of the search, the query planner invokes different procedures. These procedures in turn can post subtasks that could be accomplished in a variety of ways, depending again on context. Thus, query-planning mediators provide a flexible mechanism for performing conspectus search.

Figure 3 illustrates the kinds of activities the query planner might invoke. The nodes contain the name of the task and in some cases the names of some procedures for achieving it. The arrows represent subtask relationships. The actual procedure the query planner executes depends on context, in ways specified by our consulting librarians. The task requires capabilities that are distributed among various agents within the UMDL. Thus, by elaborating the procedures, the query planner dynamically builds a team of agents that together accomplish the task. See the later section "Example queries" for a brief description of this procedure.

MARKET-BASED RESOURCE ALLOCATION

The digital library creates a potentially unbounded demand for computational resources. For example, any preprocessing of collection data—indexing, metadata gathering, or caching—might improve system response to subsequent user requests. With only finite resources, however, we cannot take advantage of all such opportunities. Neither can we try every method for accomplishing a given task. Rather, we must choose among available methods on the basis of resource requirements and prospects for success.

Information service economy

We model alternative information services as economic activities that compete to provide the highest service level for minimal computational resources. The goal of UMDL as a whole is to allocate resources efficiently to optimize user services.

To organize processing activities within an economic framework, we treat agent interactions as supplier-producer relationships. Each agent creates value-added information products from the input products others provide.[8] Agents connect dynamically as opportunities arise for mutually beneficial exchanges. The collections provide "raw materials" in this process, whereas end users are the ultimate consumers of the "finished goods." The mediators ("middlemen") improve the value of information along the way using knowledge, processing, storage, or other computational resources.

Market facilitators

Market facilitators, or auctions, operate by collecting offers and determining agreements among agents. One simple kind of auction collects bids and settles them by some market-clearing process. Others perform a more complicated matching and search process. In our basic UMDL market protocol, one auction agent represents each good. A good could be delivery of digital objects, translation services, or other agent product. Each auction agent accepts offer messages from agents interested in buying or selling that good. Offers include a demand schedule that specifies the amount (quantity or quality) of information good the agent will transact at various prices. The auction finds a price that balances supply and demand, reports the price to the agents, and executes the transaction.

Describing goods and services

To design a market in library services, we must determine the goods and services and how to represent them in the system.[9] However, in large-scale dynamic markets, the set of goods and their important distinctions change over time. A structured, expressive good description language (part of our conspectus language) defines goods as variations and combinations of primitive concepts. From these descriptions, agents can automatically determine how to perform the necessary transformations.

For example, if the language contains the concepts NPR and Broadcast, we can construct the concept NPR Broadcast. Since one operation that agents can perform on Broadcasts is to make Transcripts, we have a meaningful notion of NPR Transcript. Parameterization provides extra degrees of freedom; for example, descriptions can qualify NPR Transcripts by date and topic.

Intellectual property usage licenses

In an information and information services market, the essence of goods is information content, not realization in some physical medium. This suggests that an exchange in information goods should distinguish between the intellectual property and its physical manifestations. Having a copy of an intellectual work does not imply the authority to do anything with the information that work represents. We refer to such authority generically as intellectual property usage licenses. Licenses are the primary type of information good exchanged in the system.

SUPPORTING INQUIRY-BASED EDUCATION

Merely wiring a classroom to the Internet—or even to a digital library—will not make students learn through inquiry.[10] Existing Internet-based tools do not effectively support access to digital resources or address the special constraints of a secondary-school classroom for sustained inquiry. For example, 50-minute class periods are very confining for students and teachers trying to engage in inquiry. Our strategy is to understand the real challenges in the classroom and design UMDL services that explicitly address these needs.

Teacher challenges

Developing good curriculum materials is a time-consuming task under any circumstances. The search for motivating, engaging, content-filled on-line materials is particularly so. Moreover, our experiences with on-line curriculum delivery suggest that a teacher should seed the Web pages with a few jump-start collections. Students need to find something quickly and have some immediate success to maintain their motivation and engagement.

At least two types of UMDL agent services can assist teachers in developing and managing curriculum materials. First, we are developing a customized version of the query-planning agent called QuickScan. Its specialized knowledge of pedagogical relevance helps a teacher quickly search and retrieve material useful to high school science classes. The QuickScan agent focuses on collections that are age-appropriate and have a range of nontextual media types (video, images, audio). Students, too, will be able to use QuickScan to find relevant information in a timely manner.

Second, remora agents (see sidebar on next page) provide a time-saving way for teachers to monitor the development of on-line materials. The Web contains many potentially relevant sites. However, a large percentage of them are still not sufficiently developed to permit effective classroom use. Also, while many Web sites provide information about current events, like volcanic eruptions, checking sites manually is tedious and time-consuming. Remora agents help teachers monitor the evolution of these sites and incorporate the materials into an on-line curriculum.

Student challenges

Teachers are often reluctant to have their students "waste precious classroom time" searching for materials. They would rather just show the students sites that provide answers. However, the inquiry-based approach, by definition, requires students to engage in on-line search. Finding and evaluating sites for relevance is an intrinsic component of inquiry. The tension is real: Current search technology, particularly keywords, is time-consuming, frequently unproductive, and fosters a random approach to searching.

Our strategy is to provide UMDL interfaces and agents that support students' learning through the search process. For instance, the UMDL search interface will provide tools like spell-checking and content-specific thesauri to help sharpen query formulation. We are also developing a UIA with an interface designed to scaffold query reformulation. This will help students who find "re-searching" and following a coherent line of exploration difficult.

A second real problem in the classroom is the lack of collaboration among students. Substantive classroom conversation is a key component of learning.[11] Professionals continually engage in discourse to invent, explicate, and

The remora agent

The remora is one example of the value-added services the UMDL accommodates. A mediator agent, the remora offers event-driven notification services for a variety of library resources. Users specify events of interest and receive notifications when such events, like new items appearing in a collection, occur.

We got the name "remora" from a kind of fish that attaches itself to sharks and other large oceanic creatures. In the UMDL, remoras attach themselves to CIAs for the purpose of detecting events. On behalf of other UMDL agents, the remora accepts scripts that specify events of interest and the actions they trigger. For example, one script might ask for e-mail notification whenever a collection adds a new Hubble Space Telescope image. Another script might define filters to extract articles matching current curricular items from a Web page, and the script might include processing instructions to add the articles to a particular portfolio document in a specified way. Figure A depicts the interaction of the remora with other UMDL agents.

The remora participates in the UMDL information economy through several markets. Remoras compete with each other, and perhaps with other subscription agents, to supply the service of running scripts. They must also bid to receive events—that is, attach to CIAs—and to acquire the necessary computational resources.

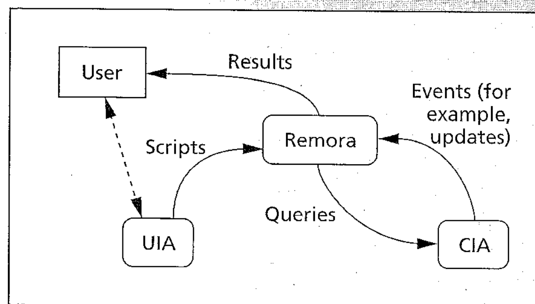

Figure A. The remora agent provides event-driven notification services by querying collections according to user scripts.

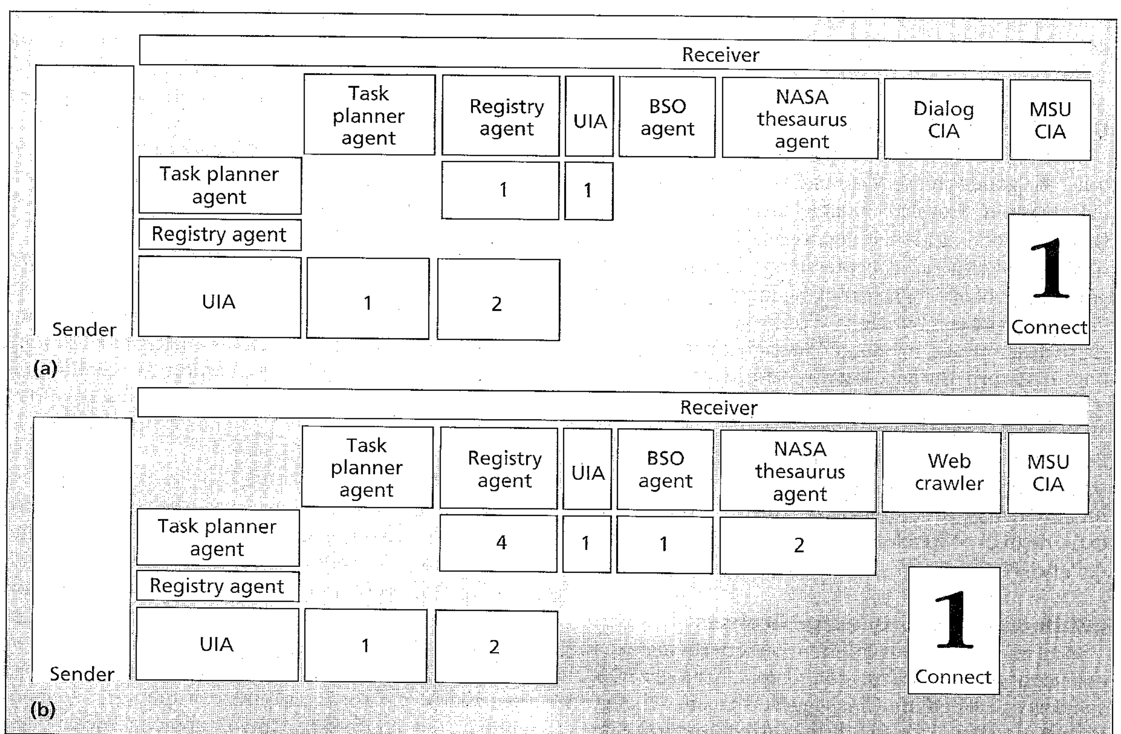

Figure 4. The remora agent monitors the number of messages passed between agents during two simple tasks. (a) The query planner returns a single CIA ("MSU") that can respond to the query. (b) The query planner consults the Broad System of Ordering (BSO) and thesaurus agents before passing the query to a Web crawler.

refine their ideas; students need dialogue for the same reasons. We are developing interface, registry, and search agents that let students share the fruits of their on-line searches. This encourages classroom interaction by providing artifacts for students to discuss. For example, a group of students could register in the UMDL their collection of on-line materials regarding a specific topic. The search agents will direct other groups of students in the class to that collection first.

Fast, simple registry of student-generated work is also allowing students to publish their findings more easily in the UMDL. For example, a class of 11th-graders recently completed a six-week unit on water contaminants. Each pair of students wrote a report on a different water contaminant, then published it on the World Wide Web. These students filled a gap. Until their efforts, no site on the Web had a comparable in-depth treatment of various water contaminants. Feeling that their ideas are respected—even desired—greatly motivates students. This typically translates into more engagement and more effective learning.

UMDL STATUS

The first version of the UMDL is currently operational at the university and is being deployed at Ann Arbor high schools. The earth and atmospheric sciences collections include material from the popular press, academic journals, encyclopedias, the World Wide Web, and local curriculum. The system is highly extensible, and we are continually expanding and enhancing content and services.

Example queries

We can illustrate a subset of the UMDL's current capabilities by summarizing its behavior for two example queries. The agents in this example include a query planner, a thesaurus agent, a BSO agent, and a remora agent. The Broad System of Ordering, or BSO, agent uses a hierarchy of terms to broaden or narrow a topical search. The remora agent has the task of persistently monitoring and summarizing message traffic in the UMDL.

For a simple task, the query planner gets a query that matches entries in the registry, requiring little interaction among the various services. The communication matrix generated by the remora, Figure 4a, shows this low level of interaction. In a more difficult query, however, the query planner must invoke the BSO and thesaurus agents. They then reformulate the query in terms of topics about which some collections have professed capability (Figure 4b). These simple examples suggest the dynamic, flexible interactions that we rely on to fulfill our ambitious vision for the UMDL.

High school deployment

We're initially deploying the UMDL in four high schools and two middle schools in Ann Arbor, with other locations planned. Besides installing the UMDL infrastructure, we have developed a substantial body of associated curricular material that includes tutorials on searching for online information, and specific topics in high school earth and space science.

By May 1996, we expect that over one thousand students will have used UMDL services. Working in a handful of classrooms is an important start. However, our aim is not merely to create a successful, innovative pilot project. We want to understand the fundamental issues involved in implementing digital libraries in schools and making them relevant to today's classrooms.

AS THE PREVIOUS SECTION SUGGESTS, many challenges remain in making technologies such as the UMDL meaningful in inquiry-based education. We are only in the initial stages of deploying the UMDL in high school and middle school classrooms. However, we already find that the UMDL agent architecture provides welcome flexibility for creating technology-based strategies to meet the challenges.

Building the UMDL raises many difficult problems of scale, decentralization, interoperability, and resource allocation. Our approach has been to define very general mechanisms and then test them with specific instances of software agents and protocols that use these mechanisms to provide library services.

Although our work on the UMDL is preliminary, the first year and a half made some things clear: First, the scale and diversity of the project will test our technical ideas—distributed agents, interoperability, mediation, and economical resource allocation. Second, the UMDL project will test our theories about the role and impact of educational technology. ■

Acknowledgments

Other project members contributing to the work described herein include Ken Alexander, Gene Alloway, Karen Drabenstott, Randall Frank, Olivia Frost, George Furnas, Daniel Kiskis, Wendy Lougee, Jeffrey MacKie-Mason, Greg Peters, John Price-Wilkin, and Amy Warner. This work was supported by the NSF/ARPA/NASA Digital Library initiative. Further information is available at http:www.si.umich.edu/UMDL/.

References

1. W.P. Birmingham et al., "The University of Michigan Digital Library: This Is Not Your Father's Library," *Proc. Digital Libraries 94*, Hypermedia Research Laboratory, Texas A&M University, College Station, Tex., pp. 53-60.
2. G. Wiederhold, "Mediators in the Architecture of Future Information Systems," *Computer*, Mar. 1992, pp. 38-49.
3. T. Finin et al., "KQML as an Agent Communication Language," *Proc. Third Int'l Conf. Information and Knowledge Management*, ACM Press, New York, 1994.
4. A. Rajaraman, Y. Sayiv, and J.D. Ullman, "Answering Queries Using Templates with Binding Patterns," *Proc. ACM Symp. Principles of Database Systems*, ACM Press, New York, 1995, pp. 105-112.
5. P.J. Samson, K. Hay, and J. Ferguson, "Blue-Skies: Curriculum Development for K-12 Education," *Proc. Conf. Interactive Information and Processing Systems*, American Meteorological Soc., Boston, 1994.
6. A. Nica and E.A. Rundensteiner, "Uniform Structured Document Handling Using a Constraint-Based Object Approach," in *Advances in Digital Libraries*, N.R. Adam, B.K. Bhargava, M. Halem, and Y. Yesha, eds., Springer-Verlag, New York, 1995, pp. 41-60.

7. J. Lee et al., "UM-PRS: An Implementation of the Procedural Reasoning System for Multirobot Applications," *Proc. AIAA/NASA Conf. Intelligent Robotics in Field, Factory, Service, and Space*, NASA Center for Aerospace Information, Linthicum Heights, Md., 1994, pp. 842-849.
8. M.P. Wellman, "A Market-Oriented Programming Environment and Its Application to Distributed Multicommodity Flow Problems," *J. Artificial Intelligence Research*, Vol. 1, No. 1, Aug. 1993, pp. 1-23.
9. T. Mullen and M.P. Wellman, "A Simple Computational Market for Network Information Services," *Proc. First Int'l Conf. Multiagent Systems*, Amer. Assn. Artificial Intelligence Press, Menlo Park, Calif., 1995, pp. 283-289.
10. E. Soloway, "Beware, Techies Bearing Gifts," *Comm. ACM*, Vol. 38, No. 1, Jan. 1995, pp. 17-24.
11. A.L. Brown and J.C. Campione, "Psychological Theory and the Design of Innovative Learning Environments: On Procedures, Principles, and Systems," in *Contributions of Instructional Innovation to Understanding Learning*, L. Schauble and R. Glaser, eds., Erlbaum, Hillsdale, N.J., 1996 (in press).

Daniel E. Atkins *is dean and professor at the School of Information and professor of electrical engineering and computer science at the University of Michigan. He is the director of the NSF-ARPA-NASA UM Digital Library (UMDL) Project, the NSF Upper Atmospheric Research Collaboratory (UARC), and a Kellogg Foundation grant to restructure graduate education for information systems professionals. His research focuses on the design and evaluation of network-based knowledge work environments. He received a PhD in computer science at the University of Illinois in 1970.*

William P. Birmingham *is an associate professor in the Electrical Engineering and Computer Science Department at the University of Michigan, with a joint appointment in the School of Information. His research interests include large, distributed information systems in areas such as distributed optimization and design, concurrent engineering, and digital libraries. He received a PhD from Carnegie Mellon University in 1988 for his dissertation on developing and maintaining large knowledge bases for design applications. Birmingham was named an NSF Presidential Young Investigator and is a member of Sigma Xi, AAAI, ACM, and IEEE.*

Edmund H. Durfee *is an associate professor in the Department of Electrical Engineering and Computer Science at the University of Michigan, where he conducts research in multiagent systems, real-time intelligent control, and cooperative problem-solving for applications ranging from interacting unmanned vehicles to supporting human collaboration. He received a PhD in computer science from the University of Massachusetts in 1987 and was named an NSF Presidential Young Investigator in 1991.*

Eric J. Glover *is a graduate student in the Department of Electrical Engineering and Computer Science at the University of Michigan, pursuing degrees in VLSI and computer science. He received a magna cum laude BSE in electrical engineering in 1990 from the University of Michigan.*

Tracy Mullen *is a PhD student in the Department of Electrical Engineering and Computer Science at the University of Michigan. Her research interests include the design of distributed information service environments based on computational market technology. She previously worked at Lockheed Software Technology Center in Palo Alto, California, and received a BS and an MS from Rutgers University.*

Elke A. Rundensteiner *is an assistant professor in the Department of Electrical Engineering and Computer Science at the University of Michigan. Her current research interests include object-oriented database technology for nontraditional applications, view and schema evolution tools, database support for digital libraries, and multimedia information systems. She received a PhD in computer science from the University of California, Irvine. Rundensteiner has received a Fulbright Scholarship, an IBM Scholarship, an NSF National Young Investigator Award, and an Intel Young Investigator Engineering Award from the Engineering Foundation.*

Elliot Soloway *is a professor in the Department of Electrical Engineering and Computer Science and in the School of Education at the University of Michigan. His current research interests lie in exploring the roles that computational media can play in self-expression, communication, and learning and teaching. Soloway is editor of* Interactive Learning Environments, *a journal devoted to exploring next-generation computational and communications technologies for learning and teaching. He received a PhD from the University of Massachusetts, Amherst, in 1978.*

José M. Vidal *is a PhD student in the Department of Electrical Engineering and Computer Science at the University of Michigan. His research interests are in agent modeling, software agents for multiagent systems, and distributed AI. He received an SB from the Massachusetts Institute of Technology and an MS from Rensselaer Polytechnic Institute, both in computer science.*

Raven Wallace *is a PhD student in educational technology at the University of Michigan. Since receiving MS degrees in mathematics and civil engineering, she has taught at the college, secondary, and elementary school levels. Her current research addresses cognitive implications of digital libraries in secondary schools.*

Michael P. Wellman *is an assistant professor in the Department of Electrical Engineering and Computer Science at the University of Michigan. His current research focuses on computational market mechanisms for distributed decision making. He received a PhD in computer science from the Massachusetts Institute of Technology in 1988 for work in qualitative probabilistic reasoning and decision-theoretic planning. He received an NSF National Young Investigator Award in 1994.*

For more information about this article, contact Wellman at the Department of EECS, University of Michigan, Ann Arbor, MI 48109, wellman@umich.edu.