Network-Based Information Brokers

Richard Fikes
Robert Engelmore
Adam Farquhar
Wanda Pratt

Knowledge System Laboratory
Stanford University
{fikes, rse, adam_farquhar,pratt}@ksl.stanford.edu

Introduction

We describe a new project whose objective is to develop key technologies that will enable vendors and buyers to build and maintain network-based *information brokers* capable of retrieving information about services and products via the Internet from multiple vendor catalogs and data bases for both human and computer-based clients.

The ability to obtain relevant information in a timely and cost efficient manner is central to the effective performance of most tasks in our society. The widespread availability of computer-based information brokers will provide that ability to large communities by significantly facilitating effective access to the broad range of information that is rapidly becoming available on the Internet. The general availability of the technology to build and maintain information brokers will enable an industry to be established whose primary products are computer-based network-accessible brokering services.

We plan to demonstrate the benefits of computer-based information brokers in two examples, electronic commerce and medicine. In network-based electronic commerce, information brokers will enable buyers to cost effectively locate and obtain descriptions of desired products and services. The benefits to buyers will be comparable to the benefits provided by a knowledgeable purchasing agent. In the context of medicine, vast amounts of rapidly-changing information make it difficult for physicians to always make well-informed decisions. For example, when a physician is deciding what drug to prescribe for a patient, an information broker can help the physician determine possible side-effects, potential drug-drug interactions, which drugs are covered by the patient's insurance, how expensive the drugs are, etc.

Technical Barriers

Effective information brokering requires many capabilities, including

- Helping a human or computer client formulate a query in the broker's ontology about some class of products or services.
- Identifying information sources that are relevant to answering a query.
- Forming a plan to answer the query using a given set of relevant information sources.
- Executing the plan. This involves translating the query into the information source's ontology and syntax, obtaining responses to the query, and translating the responses into the broker's ontology and syntax.
- Presenting the responses to the client. This involves explaining to a client how the response relates to the query, defining the terms used in the response, and suggesting alternative queries that may provide additional relevant information.

While these tasks are characteristic of many information retrieval activities, the Internet environment imposes special requirements on the brokering problem: the need to deal with variety, change, and autonomy of both clients and information sources. The information seeking client

- May be a human or a software agent representing a human's interests.
- Does not know the vocabulary or access methods of all the information sources.
- May not know the vocabulary or the range of services of the information broker.

The information sources

- Are autonomous. A broker must provide value to the information providers so that they will be motivated to couple their sources to the broker.
- Have heterogeneous access methods (e.g., SQL databases, information agents, WAIS or HTTP document servers).
- Have heterogeneous domains of expertise, extents of domain coverage, and vocabularies used to model the domain.

- May be fully structured (e.g., database relations, sentences in a logic) or semi-structured (natural language documents on a document server, e-mail, indexed multimedia).
- May return information that is incomplete or irrelevant to a query because of an incomplete mapping from the broker query language to the contents of the heterogeneous information sources.
- Are subject to change over time in both vocabulary and available information.

Because widespread use of the Internet is a relatively new phenomena, these requirements imposed by the network environment are not satisfied by current information retrieval technology.

Our main technical claim is that domain-independent information gathering schemes are necessarily limited to using syntactic match techniques and are too weak for effective information brokering. Like human brokers of products, effective computer-based information brokers will take advantage of specialized domain knowledge, such as:

- The terminology used to describe products in that domain.
- Mappings between a product and its functionality (in order to support queries that ask for products providing a given functionality).
- Abstractions and assumptions that will enable the agent to retrieve information that is relevant to a query.
- Methods for appropriately combining and summarizing retrieved information.

Building such brokers as ad hoc, monolithic applications will not scale, and the resulting brokers will not be able to interoperate with the new protocols and services being developed for the Internet.

Technical Approach

Our approach in this project is to enable the construction and maintenance of domain-specific information brokers by developing

- Detailed specifications of a broker system architecture.
- A *broker shell* that implements the broker architecture and contains implementations of all the architecture's domain-independent brokering facilities.
- A domain and source modeling tool kit for developing and maintaining a broker's domain-specific expertise.
- Two example brokers: a broker in the Electronic Commerce testbed being built in the CommerceNet project, and a broker in the pharmaceutical domain for physicians to use when selecting drugs to prescribe for patients.

We are building on technology developed in ARPA's Knowledge Sharing Effort and on experience with real-world electronic commerce being acquired in the CommerceNet project.

An Information Broker Architecture

We are developing a detailed system architecture for network-based, domain-specific information brokers. The architecture will include the following modules:

Domain Model -- The broker's logical theory of its domain of expertise. The theory specifies the broker's vocabulary of objects, relations, functions, and product classes that it uses to model the domain. The theory will include models of typical tasks that clients are performing when they ask queries in the domain, mappings between products and functions, heuristics for identifying relevant information sources, etc.

Source Models -- Structured descriptions of the *competence* of each information source that the broker uses. Each such description includes a logical theory of the portion of the broker's domain of expertise about which the source provides information. The theory describes the source's vocabulary of objects, relations, and functions in which it accepts queries and provides information. A source description also includes specifications of which relations in the source's logical theory the source can provide instances of, whether the source has complete information about some relation, the cost of accessing the source, which subsets of arguments are sufficient to uniquely determine the remaining arguments, etc.

Formulator -- Assists a client in the formulation of queries in the vocabulary of the broker's domain model. The formulator will include the following modules:

Product Description Browser -- A service for informing clients about the broker's query vocabulary and domain of expertise by providing a browsable product description taxonomy consisting of hierarchies of object-oriented product descriptions from the broker's domain model;

Query By Reformulation Assistant -- A service which helps a client iteratively refine a query by providing example responses to portions or all of the query.

Alternatives Advisor -- A proposer of alternative queries which may provide additional useful information or better satisfy a client's goals.

Planner -- Determines a plan for answering a query that specifies a sequence of subqueries to find instances of a given relation or members of a given class, constraints against which retrieved instances are to be filtered, and answer composition operations to be performed on retrieved descriptions.

Executor-- Answers the query by performing the query plan. Each subquery is answered by the following submodules:

Source Identifier -- Identifies information sources that are relevant to a subquery by determining which source model contains relations or classes whose instances can be used to answer the subquery.

Query Translator -- Translates a subquery from the broker's query language to queries in the syntax and vocabularies of the identified relevant information sources.

Information Retriever -- Obtains product descriptions by sending translated queries to information sources.

Information Translator -- Translates product descriptions obtained from information sources into the broker's vocabulary and syntax.

Presenter -- Presents to a client in an appropriate format the broker's response to a query. The response may include information that is known to answer the query, describes likely answers to the query, is relevant to answering the query, elaborates the answers to the query, or explains the answers to the query. The presenter will make use of an:

Explanation Generator -- Composes explanations of the rationale for the relevance of information presented in response to queries and of the meanings of the terms occurring in the presented information.

Critical technologies in our approach

Domain and Source Modeling using Context Logic

We are developing brokers that maintain declarative, logic-based, object-oriented models of their domain of expertise and of the domains of expertise of each of their information sources. The broker's domain model will be a logical theory that specifies the vocabulary of object, relation, function, and product class names that the broker's clients can use to formulate queries. Each source description includes a logical theory of the broker's domain of expertise. The theory describes the vocabulary in which the source accepts queries and provides information.

Our modeling language is based on a **predicate logic with contexts** [1,8,11] in which first order logic is augmented with capabilities for representing multiple domain models and axioms for translating information from one model to another. The context logic will be used to state axioms that translate predicates in the information source to predicates in the broker's query language. The broker will answer queries by using these axioms to identify instances of database relations that satisfy the query. The broker will also use these axioms to translate answers retrieved from an information source into the broker's vocabulary.

The broker will use standard deduction techniques to identify which information sources can provide bindings for variables in a user query (e.g., bindings for x in "find all P(x) such that ..."). In addition, the context logic's representational machinery will address the difficult problem of dealing with sources whose descriptive vocabulary only approximates the vocabulary used in a subquery. Translating a subquery into the vocabulary of relevant sources may involve approximating, abstracting, or eliminating portions of the subquery. Only some of the descriptions retrieved by the sources may satisfy the original subquery. For example, the client may be interested in a 'blue' sweater. One information source may not have any color information. Another may use the terms 'teal' and 'navy'. Using context logic, we can declaratively specify approximation and abstraction relationships among vocabulary terms in multiple domain models and then use a standard first-order predicate logic reasoner to infer translations of queries and of query results. In addition, it allows a reasoner to simultaneously represent multiple alternative translations so that they can be compared to determine which is preferable.

We are developing a tool kit for broker developers based on the tools in the Ontolingua system [5,6]. Ontolingua, which is being developed by KSL as part of the ARPA Knowledge Sharing Initiative [2,13], is an integrated tool system for developing domain-specific ontologies in the Knowledge Interchange Format (KIF) [4,12]and for translating the resulting ontologies into application-oriented representation languages. Ontolingua:

- Augments KIF with a frame language ontology that provides convenient representation primitives for specifying class-subclass taxonomies in an object-oriented style;
- Identifies many classes of errors in ontology specification such as underconstrained variables, undefined concepts, and missing theory inclusion relationships;
- Produces hypertext documentation for browsing; and
- Provides high fidelity translation into multiple representation languages, including LOOM [10]and C-based CLIPS.

The hypertext documentation produced by Ontolingua is of particular importance to information brokering because it enables the definitions of product description terminology to be easily accessed by both broker developers and clients. Ontolingua generates hypertext webs of ontologies in the format of the World Wide Web (WWW) that read like reference manuals instead of source code. Putting ontologies in WWW format also makes it easy to integrate formal KIF theories with semiformal text used in documentation. For example, the introductory documents explaining the purpose and use of an ontology can have hypertext pointers directly from the use of a word in text to its formal definition in KIF. Similarly, terms used in the formal definitions can point directly to their definitions.

Domain-Specific Query Formulation Assistance

An important service provided by an information broker to a client is assistance with the formulation of queries. For example, clients may not know or understand the idiosyncratic vocabularies used by vendors to describe the features of their products and may not know how to relate their functional objectives for the product to those product feature descriptions. The broker must use its knowledge to appropriately constrain the query and elicit information that will be sufficient to answer it given the information sources known to the broker. Furthermore, the broker must ensure that an answer to the query will provide sufficient information to satisfy the client's goals.

The broker architecture we are developing will include the following query formulation facilities:

- A product description browsing service for informing clients about the broker's query vocabulary and domain of expertise;
- A query vocabulary for retrieving products by functional objectives;
- A "query by reformulation" service which helps a client iteratively refine a query by providing example responses to portions or all of the query.
- A proposer of alternative queries which may provide additional useful information or better satisfy a client's goals.

The basic facility in our broker architecture for assisting clients with query formulation is a browsable product description taxonomy consisting of hierarchies of object-oriented product descriptions from the broker's domain model. The class descriptions in the taxonomies will indicate to a client both the types of products accessible by the broker and the vocabulary that can be used in queries to the broker to describe those products. The product taxonomies will be accessible as both a fully cross-referenced HTML document for browsing by human clients and as a formally defined knowledge base of structured descriptions for computer-based clients.

A broker can assist a client by providing a sample of possible answers to a query or portions of a query in situations where finding all answers to the complete query would require significant time or produce a large number of answers. The user then has the option of refining the query based on the feedback or authorizing the broker to continue retrieving answers to the query as stated. If the client wishes to refine the query, the descriptions in the example answers that are to be excluded by the refinement can be examined to determine specifications that need to be changed or added to the query. In addition, the descriptions in the example answers that are representative of desired answers may contain features that can be added to the query to further refine it. Such *query by reformulation* techniques [9,14] are particularly useful for assisting clients who are unfamiliar with the descriptive vocabulary available for use in queries or with the range of information that is available for responding to queries. We anticipate those to be common problems in the information broker context and that query by reformulation capabilities will be an significant augmentation to a broker's product description service.

An important form of domain-specific expertise for an information broker is the ability to suggest alternative queries to a client that may provide a more desirable solution to the client's goals. (E.g., consider departing from a different airport when all flights are booked or extending a trip over a Saturday night to reduce the cost of the airline ticket.) In order to suggest useful alternatives, the broker needs to be able to assume or determine the client's goals and to know for a given query-goal pair what alternative queries might be useful. We plan to provide this capability by developing techniques for extending the broker's domain model to include descriptions of typical goals that a client would have when asking queries about a product class,. For each goal, we will provide rules for transforming the query to produce candidate alternative queries. We expect typical query transformations to involve abstracting or removing some condition in the query.

Results Presentation and Explanation

Effective information brokering requires presenting information obtained in response to a query in an easily understandable format and assisting the client in understanding that information. The task is nontrivial because the results may not provide precisely the information needed or intended. The query reformulation process will typically be incomplete and approximate so that the response may include information that is known to answer the query, describes likely answers to the query, is relevant to answering the query, or elaborates the answers to the query. Also, when the amount of information gathered is large, summarization will be required for human readers.

In order to enable brokers to assist their clients in understanding retrieved information, we will include in our broker architecture an explanation generation facility that can be used to provide clients with explanations of the rationale for the relevance of information presented in response to queries and of the meanings of the terms occurring in the presented information. For example, if the query is to find airline flights from London to Washington D.C. on a given date, and the retrieved information describes flights from Heathrow to Dulles, an explanation could be added to Heathrow (e.g., as a hypermedia link) saying that it is a London airport and to Dulles saying that it is a Washington D.C. airport.

We are developing tools that enable a broker to compose explanations from term definitions and from the inferences it makes, and to annotate the information it provides to a client with those explanations, either as hypertext links for human clients or as relational links for computer-based clients. The tools are based on the explanation technology we have developed in our ARPA-sponsored How Things Work Project [3] for providing interactive documentation of engineering designs. That technology dynamically generates device descriptions and causal explanations of simulated device behavior from symbolic device models, mathematical simulation models, and simulator output [7]. The explanations are produced as HTML documents that are generated dynamically by a network server when a user requests an explanation.

Acknowledgement

This work was supported by a grant from ARPA and NASA Ames Research Center (NAG 2-581).

Bibliography

[1] S. Buvac & I. Mason. Propositional logic of context. 1993.

[2] R. Fikes, M. Cutkosky, T. Gruber, & J. van Baalen. Knowledge Sharing Technology Project Overview. Stanford University, Knowledge Systems Laboratory, Technical Report KSL 91-71, November 1991.

[3] R. Fikes, T. Gruber, Y. Iwasaki, A. Levy, & P. Nayak. How Things Work Project Overview. Stanford University, Knowledge Systems Laboratory, Technical Report KSL 91-70, November 1991.

[4] M. R. Genesereth & R. E. Fikes. Knowledge Interchange Format, Version 3.0 Reference Manual. Computer Science Department, Stanford University, Technical Report Logic-92-1, March 1992.

[5] T. R. Gruber. Ontolingua: A mechanism to support portable ontologies. Stanford University, Knowledge Systems Laboratory, Technical Report KSL 91-66, March 1992.

[6] T. R. Gruber. Toward principles for the design of ontologies used for knowledge sharing. In Nicola Guarino, Ed., *International Workshop on Formal Ontology,* Padova, Italy, 1993.

[7] T. R. Gruber & P. O. Gautier. Machine-generated explanations of engineering models: A compositional modeling approach. *Proceedings of the 13th International Joint Conference on Artificial Intelligence,* Chambéry, France, 1993.

[8] R. V. Guha. *Contexts: A formalization and some applications.* doctoral dissertation, Stanford University, 1991.

[9] R. N. John Yen Michael DeBellis. Specification By Reformulation: A Paradigm for Building Integrated User Support Environments. *AAAI-88,* pages 814-818. 1988.

[10] R. MacGregor. The Evolving Technology of Classification-based Knowledge Representation Systems. In John Sowa, Ed., *Principles of Semantic Networks: Explorations in the Representation of Knowledge*, pages 385-400. Morgan Kaufmann, San Mateo, CA, 1991.

[11] J. McCarthy. Notes on formalizing context. *Proceedings of the thirteenth international joint conference on artificial intelligence,* 1993.

[12] R. Neches, R. E. Fikes, T. Finin, T. R. Gruber, R. Patil, T. Senator, & W. R. Swartout. Enabling technology for knowledge sharing. *AI Magazine*, **12**(3):16-36, 1991.

[13] R. S. Patil, R. E. Fikes, P. F. Patel-Schneider, D. McKay, T. Finin, T. R. Gruber, & R. Neches. The DARPA Knowledge Sharing Effort: Progress report. *Principles of Knowledge Representation and Reasoning: Proceedings of the Third International Conference,* Cambridge, MA, Morgan Kaufmann, 1992.

[14] F. N. Tou, M. D. Williams, R. E. Fikes, D. A. Henderson, & T. W. Malone. RABBIT: An Intelligent Database Assistant. *Proceedings of the National Conference on Artificial Intelligence,* Pittsburgh, PA, pages 314-318. 1982.