A navigation assistant for data source selection and integration

Scott B. Huffman and David Steier
Price Waterhouse Technology Centre
68 Willow Road
Menlo Park, CA 94025
huffman@tc.pw.com

Abstract

We describe an intelligent assistant that is being designed to help information specialists select and combine data sources to produce new information services. Rather than fully planning out and answering queries autonomously, the assistant will provide interactive knowledge-based support for a specialist designing a query plan. The assistant uses heuristics and constraints to help its user navigate through a potentially large set of data sources to locate those needed to produce a desired information service.

Introduction

Players in the business world routinely generate requests for information that are difficult and expensive to fulfill, but are crucial for making business decisions. The answers to these requests must be composed by combining information contained in a potentially far-flung collection of different sources, which can include internal databases (owned by the business), commercial databases, and free public sources like the Web. To cope with the complexity of information retrieval amidst this diversity of sources, Price Waterhouse (PW), like other large firms, employs groups of information specialists (library scientists and the like) who are experts on retrieving information from a wide range of sources.

In this note, we describe an intelligent assistant that is being designed to help information specialists select and combine data sources to produce new information services. By information service, we simply mean a source or stream of information with particular content (i.e., addressing some class information requests), delivered in some specified way, and perhaps subject to quality and resource constraints [Wiederhold, 1992]. The assistant, called QPEA (for "Query Planning Environment Assistant") is a navigation aid in the sense that it helps the user navigate through a potentially large set of available data sources to locate those needed to produce the desired information service. QPEA is not a fully implemented system yet, although preliminary versions of some components and a mock-up of its user interface have been built. Despite our lack of a complete implementation, we felt that it would be worthwhile to present a description of QPEA because it represents an interesting and unexplored point in the space of agents that help users deal with multiple heterogeneous data sources. We hope to have an implementation of some of QPEA's basic functionality by the time of the symposium.

The key feature that distinguishes QPEA from other information agents is its role as an *assistant* to *information specialists*. A number of researchers are building systems that act as complete query processors for end users: the user enters a query in some global representation language, and the system plans, optimizes, and executes a query across multiple heterogeneous sources to answer the query. SIMS [Arens *et al.*, 1993] and the Information Manifold [Kirk *et al.*, 1995] are good examples of this type of system. QPEA's goal is different: rather than attempting to autonomously construct a full query plan and execute it, QPEA uses its knowledge to assist a specialist constructing such a plan. It can provides suggestions about sources, and help determine potential tradeoffs (e.g., using source A is cheaper, but B will give better results), but the specialist makes the final decisions.

We chose this route after discovering the complexity of tasks our information specialists perform. Typical searches can involve dozens of sources, and the sources are selected and combined using not only knowledge about their content, but knowledge about the domain, and knowledge about the task context such as the reasons for performing the search, its intended audience, etc. Thus, trying to "replace" our specialists with a fully automated system did not seem reasonable. Rather, we are focusing on providing them with "power tools" that will increase their productivity.

Because QPEA uses knowledge about data sources to help a user identify sources for a query, it is also related to "information broker" or "mediator" agents that contain repositories of source information [Fikes *et al.*, 1995; Wiederhold, 1992]. However, these broker/mediator agents are typically designed either to

interact with other information agents, or with an end user who has a query and wants an answer. QPEA instead interacts with an information specialist who wants to construct a query plan (and understands their task at the level of a query plan) to produce a new information service.

The next section briefly describes the major components envisioned for QPEA. The balance of the paper uses a real-world information service problem within PW as an example and walks through a potential scenario in which QPEA assists in producing the needed information service.

Overview of the QPEA assistant

QPEA's task is to assist in filling in missing parts of a query plan to produce a specified information service. An information service in this context can be thought of as simply a data repository/stream containing specified contents and with specified meta-information (e.g., how often it is updated, if the sources it is built from change over time). The user may specify more or less about the service and input databases. (A companion paper, [Steier *et al.*, 1995], provides a case study of the kinds of meta-information users might wish to specify, based on those observed in real-world business data integration problems).

To perform this task, QPEA must include the following components:

1. **Domain ontology.** Because QPEA must reason about the content of heterogeneous sources, it must have a central domain ontology/vocabulary that can be used to express meta-information about content (both what different fields contain, and how it is represented – its units, etc.). This central vocabulary is also be used by the user to specify content. Such an ontology is a standard element of agents that interact with multiple heterogeneous sources. For instance, SIMS [Arens *et al.*, 1993] uses LOOM to represent a global ontology; the Information Manifold uses CLASSIC [Kirk *et al.*, 1995]; and the Stanford Information Brokers project uses KIF/Ontolingua and context logic [Dappert *et al.*, 1995].

2. **Source meta-information.** QPEA must have knowledge about available data sources: both what information they contain (cast in terms of the global vocabulary); and meta properties such as access cost, accuracy, and timeliness of data. Users may specify any of these meta properties as constraints on they information service they are trying to produce as well. For instance, they can specify that the service can only cost a maximum per week to produce, or that its accuracy must be above a certain threshold.

3. **Heuristic relational query operators.** QPEA aids in constructing query plans that contain *heuristic* relational query operators in addition to regular relational operators. For instance, in addition to regular equi-joins, QPEA can plan for a *heuristic join* [Huffman and Steier, 1995] across sources that refer to the same ontological category (e.g. `company`, `person`, etc.) in different ways. This is required because the underlying sources are typically heterogeneous not only at the schema level (e.g., source A's field `com-name` and source B's field `Company` both contain company names), but at the level of individual data instances (e.g., tuples use both `ABC Inc` and `Any Big Company Incorporated` to refer to the same company).

4. **Heuristic and constraint-based source selector/query planner.** Given a partially specified query plan, QPEA will use standard planning techniques to fill in details of the plan. These details include:

   - Possible input data sources.
   - Relational query operators used to combine input sources to produce the desired output information service.
   - Meta-properties of the output information service (e.g., its predicted accuracy, total cost, timeliness).

   The planning techniques we implement will be somewhat similar to those used by Sage in SIMS [Arens *et al.*, 1993], but using a heuristic set of query operators, and allowing the initial plan to be partially specified ahead of time. Also, we will not initially focus on query plan optimizations, such as those described in [Knoblock and Levy, 1995].

5. **Graphical user interface.** QPEA will interact with information specialists using a graphical representation of the query/service being produced. This is illustrated in the next section.

A QPEA Scenario: Selecting management changes

As a simple example of using QPEA, consider the following (real) problem. We have recently developed a data source of high-level corporate management changes, that contains tuples reporting a date, company name, person's name, and position they have obtained (e.g. "CEO"). This source is constructed automatically using an information extraction system [Huffman, 1995] that scans newswire articles. Thus, it is not 100% accurate or complete. In addition, since names of companies and people are drawn directly from newswire text, they do not follow any uniform format.

Our end-users wish to view or access the entries in the management changes database according to various properties of the company involved; e.g., its industry classification, its region of the country, its size, its current auditor. A variety of data sources contain these company properties. The problem, then, is to produce information services that index management changes

according to useful company properties, by combining the management changes database with whatever other sources are needed.

Figure 1 shows what a QPEA user might enter into a graphical interface to specify an information service that delivers management changes within a particular industry grouping. The user creates Figure 1 by dragging and dropping the graphical symbols from a palette, linking them together, and adding attributes through a cascade of menus that shows available choices.

Starting with Figure 1(a), the cylinder represents a data source; the shield, a data "sink" to be produced. The boxes represent a set of cascaded menus, where the backmost box pops up first, the user makes the choices with arrows highlighting them, and then the next menu box forward pops up, etc. In this case, the user specifies that the sink is to be delivered as a set of weekly email messages to a particular mailing list (e.g. PW personnel concerned with an industry). It could equally well be delivered as a new database, or an addition to an existing database. Other meta-information like resource limitations on producing the sink, or accuracy requirements, could also be specified here. The user could also have started by specifying source attributes.

In (b), the user specifies the content of the sink, in terms of the domain ontology. Content could be specified either by concept features (like `Person.Name`) or by events (like `management-change`), as shown by the two different menus in the center of the figure. Let's assume in this example that the user specifies content by selecting the highlighted concept features.

In (c), the user turns to specifying a source. QPEA aids in this process. It knows what concept features the sink must include, so it searches its knowledge of sources to find ones that include those features. In this example there are a few such sources. However, only one of the sources produces timely data on an ongoing basis. QPEA uses the heuristic that an ongoing email stream most likely delivers timely data, to put the source `management-change-db` at the top of the list of choices.

Figure 2 continues the scenario. In (a), the user inserts a "select" filter (the triangle) between the source and sink. The filter can use any features of the concepts in the input source(s). Here, the input source contains concepts `Person` and `Company`. The user chooses `Company.Industry` to filter on (an attribute classifying a company into one of a static set of industry types), and selects *Retail* as the value.

Now QPEA goes into action. The input source does not contain `Industry`, but rather refers to the concept `Company` only using a `Name` string extracted from a newswire article. To fill in the query plan, QPEA must determine how to map from `Company.Name` to `Company.Industry`.

Figure 2(b) shows the result. QPEA finds that to map from `Company.Name` to `Company.Industry`, it must go through two sources. Going from top to bottom in Figure 2, the first source, `Company-SIC-map`, contains tuples that relate company names to four-digit industry codes called SIC codes. This source must be joined with the tuples in `management-changes-db`. Unfortunately, the text string specifying a given company's name in the `Company-SIC-map` source may not exactly match its name string in the `management-changes-db` source. (In fact, because it is built from newswire articles, the `management-changes-db` source is not even consistent internally about always using the same name string for a given company). Thus, QPEA selects a *heuristic join* (H-Join in the figure) to join the sources. Heuristic joins have a number of parameters not present in normal equi-joins, as described in Huffman and Steier [1995]; these include a match threshold and an indicator of whether a tuple in source A can match one or multiple tuples in source B. QPEA uses default settings for these parameters as shown; the user can adjust them if needed.

Given companies' SIC codes, the source `SIC-Industry-map` (at the bottom left of Figure 2 will provide their `Industry` values. Since SIC codes are an exact four-digit value, QPEA uses a normal equi-join to join the output of the previous heuristic join with `SIC-Industry-map`. This gives `Industry` values for each company appearing in the `management-changes-db`. These values can then be used to perform the filter for *Retail* companies.

Discussion

This simple scenario illustrates how even a basic navigation and query planning assistant can be of great help to information specialists in producing new information services. Here, the agent assisted the information specialist in locating the three input sources needed to produce the desired output, based on an incomplete conceptual specification. The specialist did not have to search through specs of sources on different platforms and mediums, determine what sources were combinable and how, etc.

The planning techniques required to support this scenario are straightforward – basic means-ends analysis and some simple heuristics are enough (more complex scenarios will require more complicated planning). Building a useful assistant is less dependent on complex reasoning, and more on a strong knowledge base: a repository of meta-information about available data sources, and a set of heuristics for how to select and combine sources in sensible, resource-effective ways. Because the meta-information about sources can be updated over time by multiple information specialists, the assistant reduces the chances of a specialist overlooking a new data source that they are personally unfamiliar with.

Because it is targeted as an assistant for special-

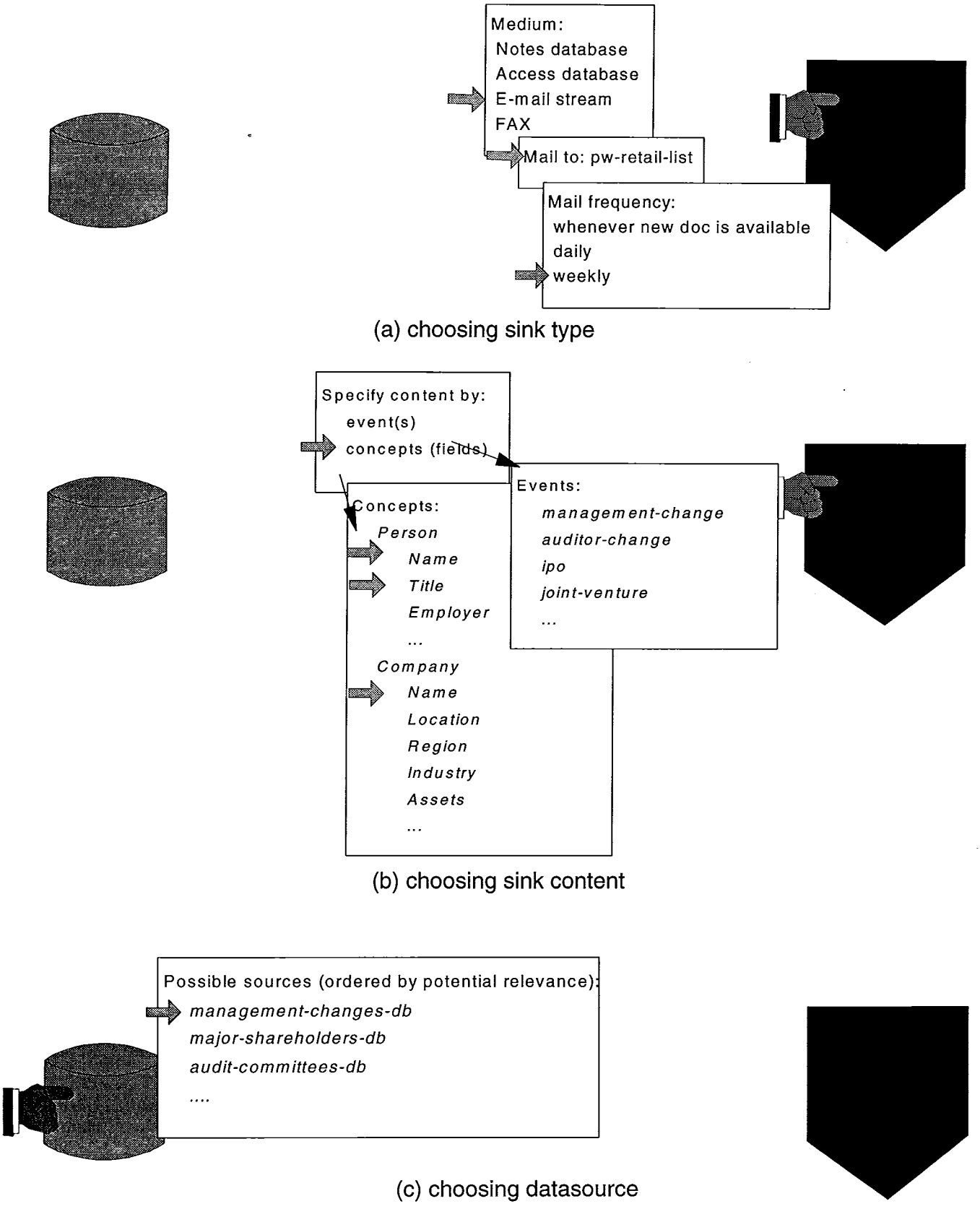
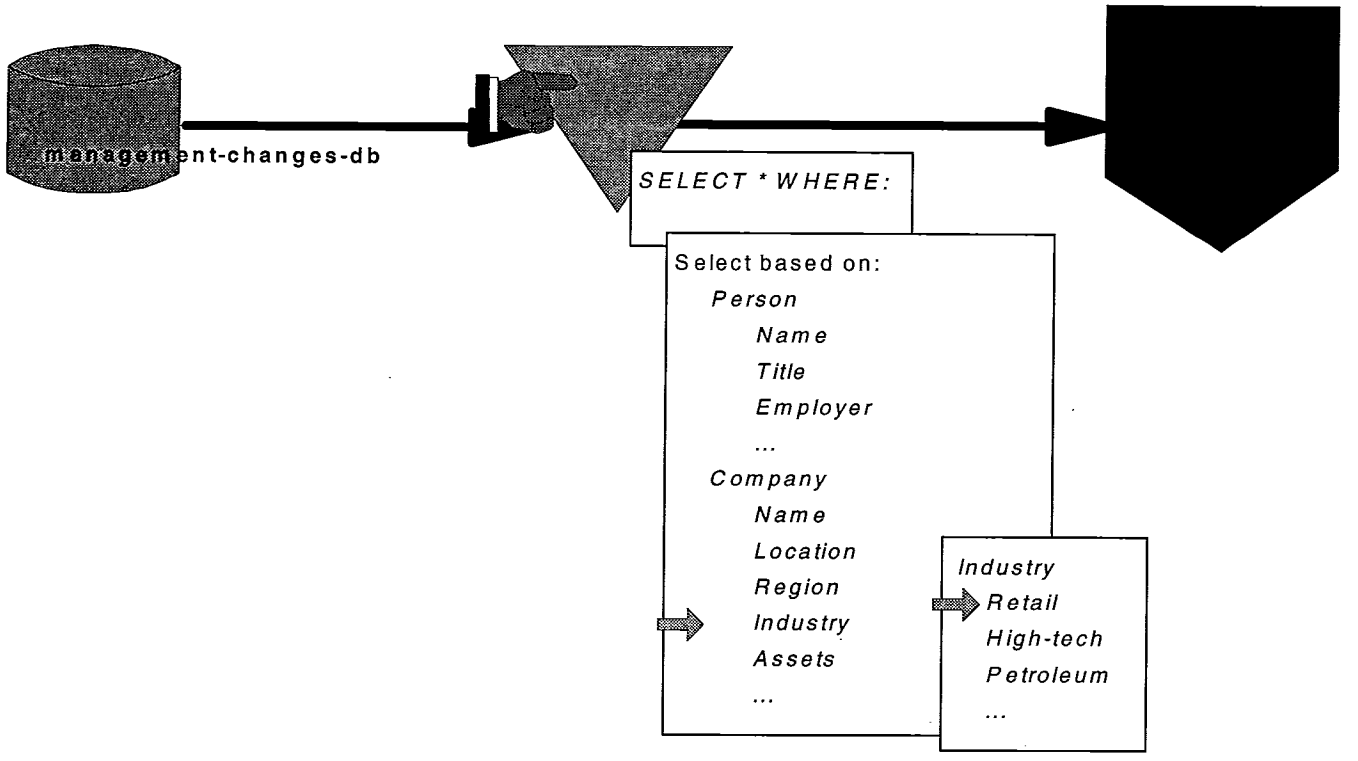
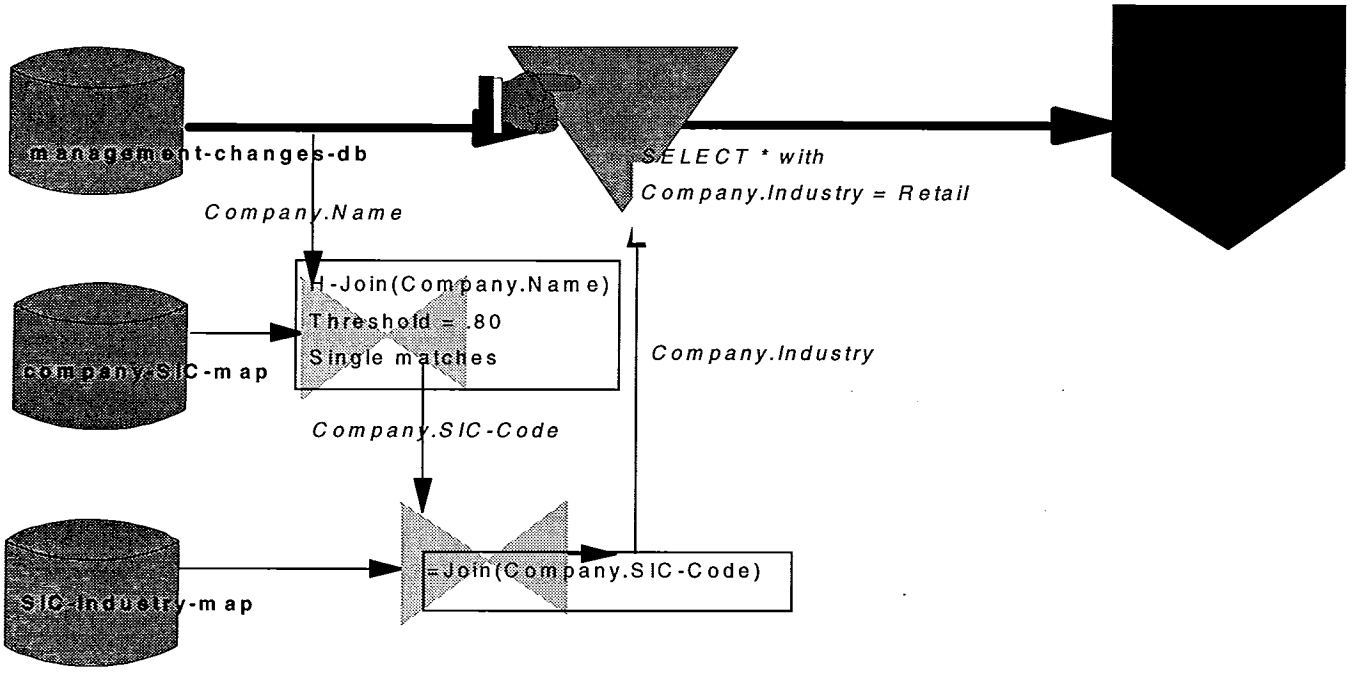


Figure 1: Interaction with QPEA.



(a) specifying a selection



(b) QPEA selects needed sources and fills out query plan

Figure 2: Interaction with QPEA, continued.

ists, QPEA represents a different point in the space of information agents than most others have been exploring. It brings together the ideas of a mediator (with knowledge about sources), a query planner (with knowledge about selecting sources to combine), and a set of heuristic query operators (with knowledge about combining heterogeneous data) into an interactive assistant.

As mentioned earlier, QPEA is not fully implemented. The main piece that has been completed is a heuristic join system that performs joins over sources containing heterogeneous data of specific types (e.g. company names). This tool is in daily use, updating sources like the management changes database to include related corporate information. However, the tool requires that its user select the sources and fields to be joined. Our current work involves extending the tool to aid its user in source and field selection, using source meta-information and simple planning heuristics.

References

[Arens *et al.*, 1993] Y. Arens, C. Y. Chee, C. Hsu, and C. A. Knoblock. Retrieving and integrating data from multiple information sources. *Int'l. Journal on Intelligent and Cooperative Information Systems*, 2(2):127–58, 1993.

[Dappert *et al.*, 1995] A. Farquhar A. Dappert, R. Fikes, and W. Pratt. Integrating information sources using context logic. In C. Knoblock, A. Levy, S. Chen, and G. Wiederhold, editors, *Working notes of the 1995 AAAI Spring Symposium on Information Gathering in Heterogeneous, Distributed Environments.* 1995.

[Fikes *et al.*, 1995] R. Fikes, R. Engelmore, A. Farquhar, and W. Pratt. Network-based information brokers. In C. Knoblock, A. Levy, S. Chen, and G. Wiederhold, editors, *Working notes of the 1995 AAAI Spring Symposium on Information Gathering in Heterogeneous, Distributed Environments.* 1995.

[Huffman and Steier, 1995] S. B. Huffman and D. Steier. Heuristic joins to integrate structured heterogeneous data. In C. Knoblock, A. Levy, S. Chen, and G. Wiederhold, editors, *Working notes of the 1995 AAAI Spring Symposium on Information Gathering in Heterogeneous, Distributed Environments.* 1995.

[Huffman, 1995] S. B. Huffman. Learning information extraction patterns from examples. In S. Wermter, G. Scheler, and E. Riloff, editors, *Proceedings of the 1995 IJCAI Workshop on New Approaches to Learning for Natural Language Processing.* 1995.

[Kirk *et al.*, 1995] T. Kirk, A. Y. Levy, Y. Sagiv, and D. Srivastava. The information manifold. In C. Knoblock, A. Levy, S. Chen, and G. Wiederhold, editors, *Working notes of the 1995 AAAI Spring Symposium on Information Gathering in Heterogeneous, Distributed Environments.* 1995.

[Knoblock and Levy, 1995] C. A. Knoblock and A. Y. Levy. Exploiting run-time information for efficient processing of queries. In C. Knoblock, A. Levy, S. Chen, and G. Wiederhold, editors, *Working notes of the 1995 AAAI Spring Symposium on Information Gathering in Heterogeneous, Distributed Environments.* 1995.

[Steier *et al.*, 1995] D. Steier, S. B. Huffman, and W. C. Hamscher. Meta-information for knowledge navigation and retrieval: What's in there. In *Working notes of the 1995 AAAI Fall Symposium on AI Applications in Knowledge Navigation and Retrieval.* 1995.

[Wiederhold, 1992] G. Wiederhold. Mediators in the architecture of future information systems. *IEEE Computer*, 25:38–49, 1992.