The IIT Intranet Mediator: An Overview

D. Grossman, S. Beitzel, E. Jensen, and O. Frieder
Information Retrieval Laboratory
Illinois Institute of Technology
E-mail: grossman@ir.iit.edu

Introduction

A mediator sits between a user and a variety of data sources. It acts on behalf of the user to provide "one stop shopping" to an organization's data. An intranet, or digital library, is an excellent example of the need for a mediator because of the myriad of data types that exist in the digital library. A user query such as "*What are the three best restaurants in Chicago*?" should not search for the word *three*. A mediator will recognize that this query needs to access structured information and will pose the query as a conventional SQL query to a data warehouse of structured information. The mediator may submit a request to unstructured sources as well. The result will be both a suggested answer to the user's question as well as a set of related links to other potentially useful information. We note that our key goal is simply to answer natural language questions that cross both structured and unstructured data. Presently, work was done on the TREC Question Answering track that focuses on unstructured sources [Hara00]. Our work builds on this as we wish to include structured sources as well.

Integrating Structured Data and Text

Our group has worked in the area of integrating structured data and text since the early 1990's. We initially focused on the use of the relational model to provide seamless integration [Gros94, Gros97, Lund99]. More recently, we focused on the use of multi-dimensional databases to provide new functionality [McCa00]. This work uses recently developed on-line analytical processing tools (OLAP) to provide functionality that has eluded conventional text search engines. This has led us to the notion of an intranet mediator that provides seamless access to not only structured data and text, but also images, video and sound as well.

Physical vs. Logical Data Warehouse

Either a virtual or physical data warehouse may be used for data integration. A *Virtual Data Warehouse* (VDW) does not physically exist. All of the data are stored across the network and are housed in a variety of different databases. A query is sent to a mediator, which in turn accesses a single schema. The schema indicates how each datum can be obtained. The mediator to various data sources about books will send a query such as "Who published Gone With the Wind?". Schema reconciliation (e.g., handling a case where one source stores *publisherName* and another does not) is done at query time. Most existing research projects use mediators that are based solely on schema reconciliation at query time [Baru98, Chu93, Flor98, Shaw95]. The approach is appealing because no data are copied and the distributed sources work in a somewhat autonomous fashion. The reality is that schema integration must be done *at query time*.

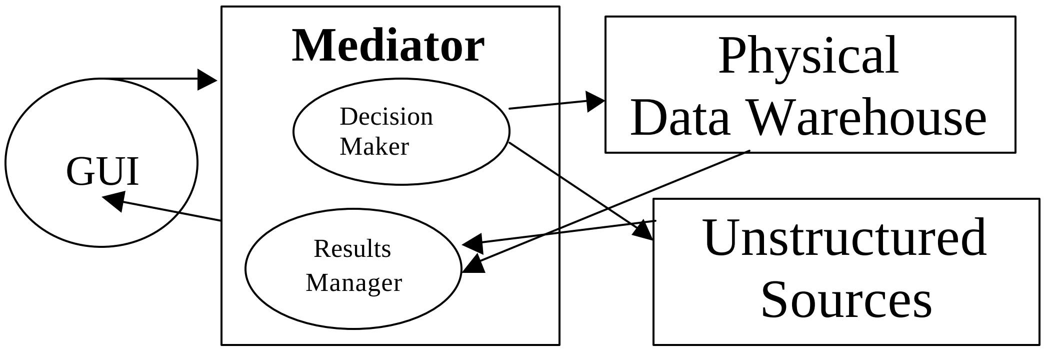
A *Physical Data Warehouse* (PDW) [Inmo93, Kimb96] is an actual copy of the structured data from a variety of sources. An ETL (extract, transform, and load) process is used to migrate data from the disparate source databases to the data warehouse well in advance of query time. Additionally, the warehouse can be optimized for decision support so various summary information can be stored in the data warehouse prior to a query. We note that this pre-summarization can only be done with structured data such as (*salary, sales, etc.)* and is not applicable to unstructured data. The metadata about the global schema is very similar for both the physical and logical data warehouse. *The only real difference is whether or not schema reconciliation is done at query time or prior to the query.*

The VDW is appealing for environments such as the internet where privacy issues surround each data source, as well as environments where various sources do not wish to house their data in another physical location. On the internet, access to the majority of data is somewhat restricted and this dramatically reduces the usefulness of a PDW for large volumes of data. However, on an intranet or digital library the data are by definition public, so no privacy issues exist; all of the data on an intranet are usually owned by a single organization. Hence, a PDW solution is possible. Given that the PDW only provides integrated access to structured data, a mediator is needed to ship requests for unstructured data to various data sources and for structured data access to the data warehouse.

Architecture

Our mediator is currently in use locally at the Illinois Institute of Technology. Figure 1 illustrates the architecture of our mediator. Two components are included: a physical data warehouse and a decision-making engine. The engine is subdivided into four main components: the user interface, the query processor, the decision maker, and the query modules. The user interface presents to the user a query entry field along with a structured database selection option. The structured selection option simply allows the user to override the decision maker's automatic source selections and specifically instruct the mediator as to what sources should be used. The query processor performs a light parse of the query, which is split into a concept tree and passed to the decision maker. The decision maker's responsibility is to decide which of the available sources are most appropriate to the query and send information in the query to them. Here, the global schema is accessed and a few (at present somewhat simple) handcrafted rules are followed to make a decision as to which sources should be queried. If the data warehouse is included as a source, the corresponding query module generates a SQL request. Queries are then submitted to each source and results are returned to a results manager. Duplicates are then removed and final results are passed on to the user interface. The result accumulation system of the engine allows for asynchronous results gathering. The user interface does not wait for all results to be returned before some are displayed; it displays them as they arrive.

Figure 1



When a query arrives from the user interface, it is translated into a simple concept tree which identifies the key concepts in the query. No phrase processing or stop word removal is done here. Such processing is done locally because each source can better determine the value of each token to their particular database. Each individual source system performs these tasks. After query processing is complete, the query is passed on to the decision-making components of the mediator. These components use information available from our store of metadata to determine which of our available sources are appropriate.

First, we identify key phrases from the initial query. The metadata manager is consulted to determine the presence of these phrases as either entities or attributes in the structured data. Hand-crafted synonyms also exist in the metadata manager to assist with this process. If structured data are found, a SQL query is built from the initial natural language query. Otherwise, a metasearch of unstructured sources is used. Hence, our mediator is no worse than a typical metasearch engine [Bart94, Drei97, Glov99, Glov00]. *A typical metasearch engine simply submits the query to a group of unstructured sources - we do this and consult structured sources as well.*

Following the selection of sources for query processing, the query is sent to the corresponding query modules and results are returned to the mediator component. The query modules are implemented as independent threads that are inherently capable of obtaining and returning results concurrently. The asynchronous nature of results gathering is handled by the acceptor object, which serializes the reception of results from all active query modules. These results are then sent to the results manager for unification.

The mediator uses unstructured query modules to query unstructured sources. Currently, we have implemented modules that interface with popular search engines such as Excite and Infoseek. These modules simultaneously query their respective sources and propagate results back to the acceptor.

The basic mediator algorithm can be described as:

Step 1: Accept query from user.
Step 2: Check metadata for potential structured sources for the query.
Step 3: If structured sources exist, build SQL queries to access the intranet data warehouse.
Step 4: Submit natural language query to all unstructured sources.
Step 5: Collect results, remove duplicates, and display to the user.

Notice the algorithm lacks a schema reconciliation step. This has been done long before the query was input. The result is a very simple mediator at run time. The only complex step here is Step 2 where we determine if any structured sources exist for the query. At present, we send the query to all unstructured sources. Ultimately, the decision maker could be improved to selectively submit the query to specific unstructured sources.

Reconciling Results

The independent operation of the modules entails that results obtained from the disparate unstructured sources might not arrive in a predictable order or in a reasonable time frame. Hence, we conducted a detailed analysis of various methods for managing the asynchronous arrival of results. This facilitates a better-unified ranking in the results manager and ultimately more timely access to results set in the user interface.

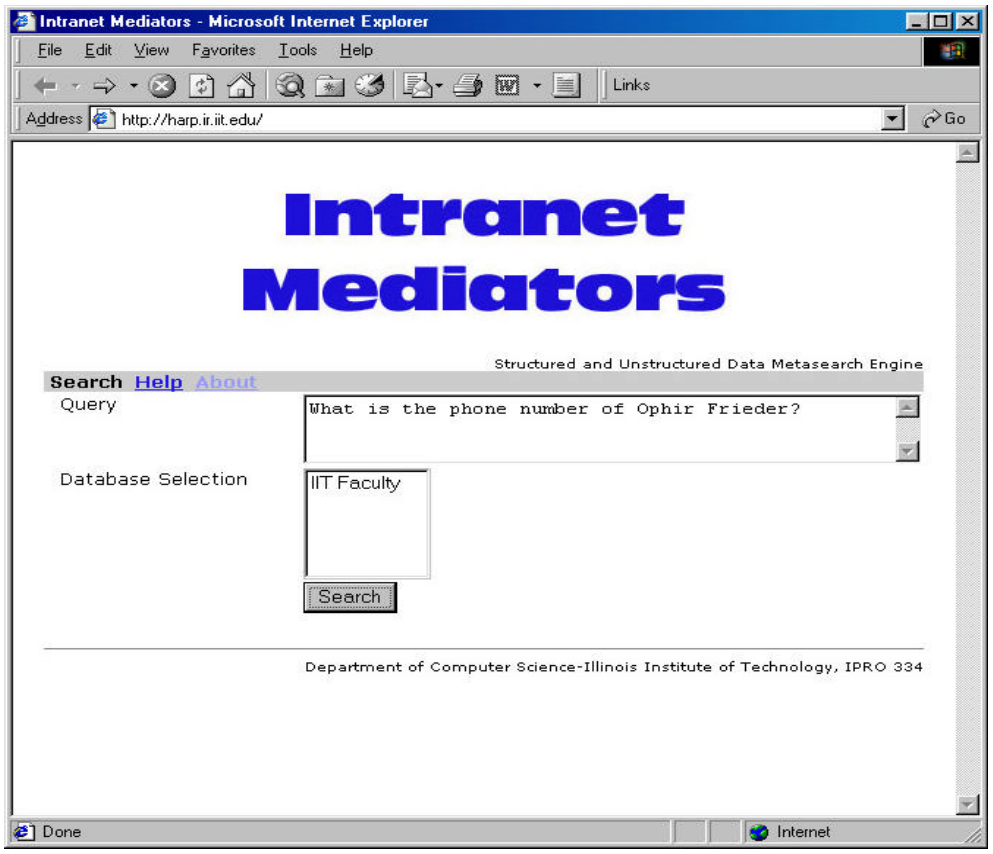
Numerous mediators take different approaches. Inquirus asynchronously displays results [Glov99, Glov00]. This is effective in the sense that users do not have to wait for a result, but it has the potential for sub-optimal relevance ranking. Some documents may not arrive until after some ranking was done. To alleviate this problem, an improved prototype, Inquirus2, has two windows, and one window is constantly re-ranking documents. This avoids the problem of latearriving good document, but now presents a confusing view to the user. Documents are re-ranked in front of their eyes, making it difficult to comprehend the ranked list of results.

To achieve an optimal time/ranking tradeoff, we wait for some, small fixed time period of *t* seconds for some documents to arrive. After *t* seconds, we rank the results we have and display them. Simultaneously, we continue retrieving documents in asynchronous threads of execution. As documents arrive they are stored in a cache. When the user hits the "next" button, we now remove duplicates and show the documents that arrived while the user was viewing the first set.

User Interface

There are two key issues with the user interface: display of structured and unstructured answers on the same page and display of a single answer that the mediator believes is correct. First, if a structured source is available, and it returns a single tuple as a result, our user interface displays "we think the answer is" and gives the result. However, we know the answer may be wrong so we also display a "related information" section. The effect is that the user can quic kly see the answer to a question like *"What is the phone number of Ophir Frieder?"* and, if all goes well, will see related links to other unstructured information like John's home page.

A second problem is display of a set of structured results. A query such as "*What are the phone numbers of all Computer Science faculty?*" may well retrieve a set of tuples. We present these as a single link for the user to click. The link is ranked and displayed along with several other unstructured links. The user interface, which accepts the query and results screens, is shown below.


Intranet Mediators - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Links
Address
Go
Search
Done
Internet

http://harp.ir.iit.edu/

Intranet Mediators

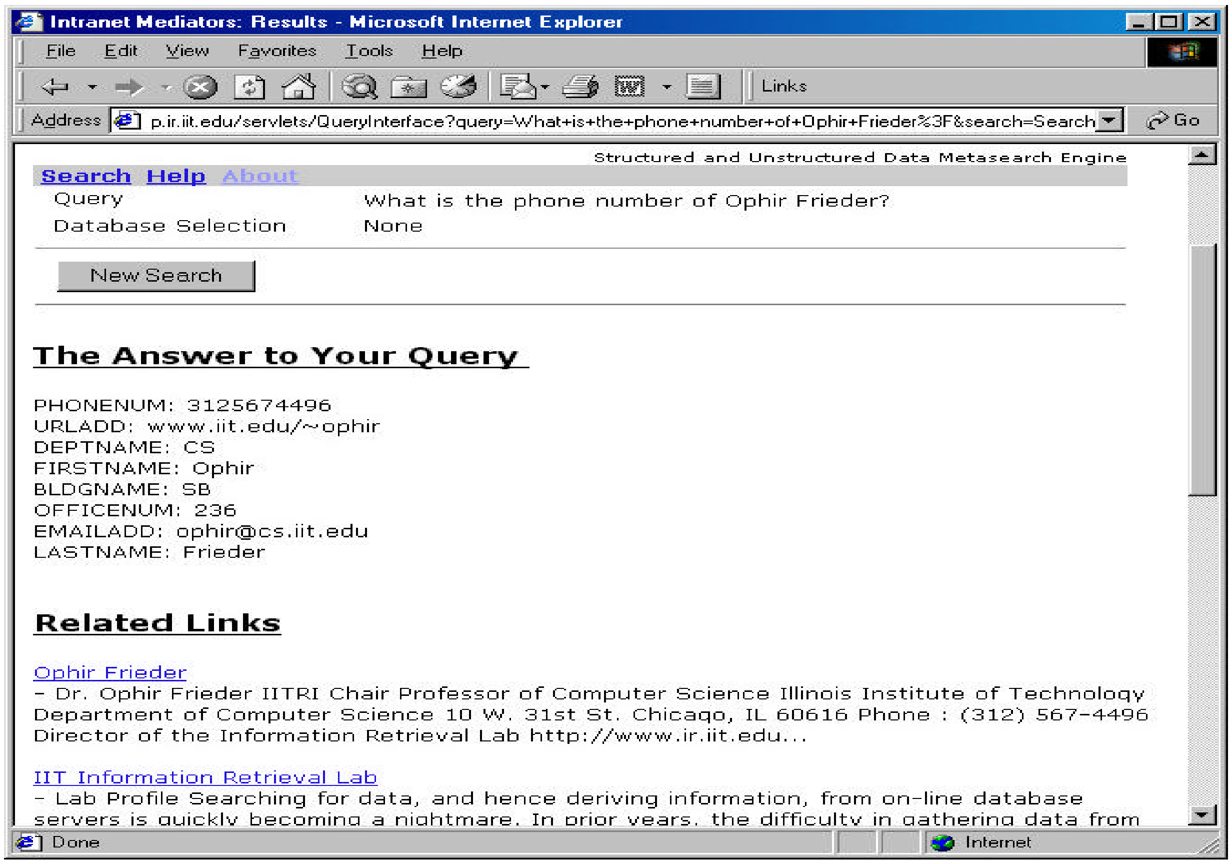
Structured and Unstructured Data Metasearch Engine

Search Help About

Query: What is the phone number of Ophir Frieder?

Database Selection: IIT Faculty

Department of Computer Science-Illinois Institute of Technology, IPRO 334


Intranet Mediators: Results - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Links
Address p.ir.iit.edu/servlets/QueryInterface?query=What+is+the+phone+number+of+Ophir+Frieder%3F&search=Search
Go
Search Help About
New Search
Done
Internet

Structured and Unstructured Data Metasearch Engine

Query What is the phone number of Ophir Frieder?
Database Selection None

The Answer to Your Query

```
PHONENUM: 3125674496
URLADD: www.iit.edu/~ophir
DEPTNAME: CS
FIRSTNAME: Ophir
BLDGNAME: SB
OFFICENUM: 236
EMAILADD: ophir@cs.iit.edu
LASTNAME: Frieder
```

Related Links

Ophir Frieder
- Dr. Ophir Frieder IITRI Chair Professor of Computer Science Illinois Institute of Technology Department of Computer Science 10 W. 31st St. Chicago, IL 60616 Phone : (312) 567-4496 Director of the Information Retrieval Lab http://www.ir.iit.edu...

IIT Information Retrieval Lab
- Lab Profile Searching for data, and hence deriving information, from on-line database servers is quickly becoming a nightmare. In prior years, the difficulty in gathering data from

Future Work

Numerous issues are yet to be resolved. The feasibility of a monolithic data warehouse even for an intranet must be ascertained. As the data volumes grow in this warehouse, the global schema becomes more complex and the decision-making component has a more difficult task. It is reasonable to expect that machine learning and data mining techniques may be used instead of handcrafted rules for each data source. The user interface is another area – we are unaware of user interfaces that tell the user "we think the answer is *x*" for question answering and also tell the user "for other related information, see *y*". This kind of user interface seems reasonable for a digital library, but we need to perform user studies to validate its effectiveness. Finally, improvements to the run-time efficiency of the decision maker as well as the results manager will be needed.

Acknowledgments

We gratefully acknowledge the work done by Michael Lee. His contribution enhanced our mediator prototype.

References

[Bart94]Bartell, B. T., G.W. Cottrell, and R.K. Belew. Automatic combination of multiple ranked retrieval systems. *Proceedings of the Seventeenth Annual ACM_SIGIR Conference on Research and Development in Information Retrieval (SIGIR '94)*, 1994.

[Baru98]Baru, C., A. Gupta, B. Ludascher, R. Marciano, Y. Papakonstantinou, P. Velikhov, V. Chu. XML-Based Information Mediation with MIX. *In Proceedings of the Special Interest Group on Management of Data (SIGMOD '98)*, 1998, pp. 597-599.

[Chu93]W. W. Chu, M. A. Merzbacher, L. Berkovich. The Design and Implementation of CoBase in *Proceedings of ACM SIGMOD '93*, Washington D.C., 1993.

[Drei97]Dreilinger, D. and A. E. Howe. Experiences with selecting search engines using metasearch. *ACM Transactions on Information Systems*. 15(3):195-222, 1997.

[Flor98]Florescu, D., A. Levy and A. Mendelzon. Database Techniques for the World Wide Web: A Survey. *SIGMOD Record*, 27 (3), pp. 59-74, 1998.

[Glov99]Glover, E. J., S. Lawrence, W. Birmingham, C. Lee Giles. Architecture of a Metasearch Engine that Supports User Information Needs. In *Proceedings of the Eighth International Conference on Information Knowledge Management (CIKM99)*, ACM, 1999.

[Glov00]Glover, E. J., S. Lawrence, M. Gordon, W. Birmingham, C. Lee Giles. Web Search - Your Way. Accepted for publication in *Communications of the ACM*.

[Gros94]D. Grossman, D. Holmes, O. Frieder, "A Parallel DBMS Approach to IR in TREC-3," *Overview of the Third Text Retrieval Conference*, December 1994.

[Gros97]D. Grossman, D. Holmes, O. Frieder, D. Roberts. "Integrating Structured Data and Text: A Relational Approach", *Journal of the American Society of Information Science*, February 1997.

[Hara00] S. Harabagiu, D. Moldovan, M. Pasca. R. Mihalcea, M. Surdeanu, R. Bunescu, R. Girju, V. Rus, and , P. Morarescu. Falcon: Boosting knowledge for answer engines. *Proceedings of the Ninth Text Retrieval Conference (TREC-9)*. November, 2000.

[Lund99]C. Lundquist, O. Frieder, D. Holmes, and D. Grossman, "A Parallel Relational Database Management System Approach to Relevance Feedback in Information Retrieval," Journal of the American Society of Information Science, 50(5), 1999.

[McCa00]M.C. McCabe, J. Lee, A. Chowdhury, D. Grossman, O. Frieder, "On the Design and Evaluation of a Multidimensional Approach to Information Retrieval," In *Proceedings of the Nineteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, July 2000.

[Inmo93]Inmon, W., Building the Data Warehouse, *John Wiley and Sons*, 1993.

[Kimb96]Kimball, R., The Data Warehouse Toolkit, *John Wiley and Sons*, 1996.

[Shaw95]Shaw, J. A. and E. A. Fox. Combination of Multiple Searches. *The Third Text Retrieval Conference (TREC 3)*, 1995. National Institute of Standards and Technology Special Publication.