Infomaster: An Information Integration System*

Michael R. Genesereth
Computer Science Dept.
Stanford University
genesereth@cs.stanford.edu

Arthur M. Keller
Computer Science Dept.
Stanford University
ark@cs.stanford.edu

Oliver M. Duschka
Computer Science Dept.
Stanford University
duschka@cs.stanford.edu

Abstract

Infomaster is an information integration system that provides integrated access to multiple distributed heterogeneous information sources on the Internet, thus giving the illusion of a centralized, homogeneous information system. We say that Infomaster creates a virtual data warehouse. The core of Infomaster is a facilitator that dynamically determines an efficient way to answer the user's query using as few sources as necessary and harmonizes the heterogeneities among these sources. Infomaster handles both structural and content translation to resolve differences between multiple data sources and the multiple applications for the collected data. Infomaster connects to a variety of databases using wrappers, such as for Z39.50, SQL databases through ODBC, EDI transactions, and other World Wide Web (WWW) sources. There are several WWW user interfaces to Infomaster, including forms based and textual. Infomaster also includes a programmatic interface and it can download results in structured form onto a client computer. Infomaster has been in production use for integrating rental housing advertisements from several newspapers (since fall 1995), and for meeting room scheduling (since winter 1996). Infomaster is also being used to integrate heterogeneous electronic product catalogs.

1 Introduction

In recent years, there has been a dramatic growth in the number of publicly accessible databases on the Internet, and all indications suggest that this growth will continue in the years to come. Access to this data presents several complications.

The first complication is *distribution*. Not every query can be answered by the data in a single database. Useful relations may be broken into *fragments* that are distributed among distinct databases. Database researchers distinguish among two types of fragmentation. In *horizontal* fragmentation, the rows of a database are split across multiple databases. For example, GM will maintain its own catalog of cars separately from Ford's catalog of cars. In *vertical* fragmentation, the columns are split. For example, while the basic description of each car model is consistent, the price of the same model car may vary from dealer to dealer. Car model descriptions should come from the manufacturer's database, while price may come from the dealer's database. Distributed databases can exhibit mixtures of these types of fragmentation.

A second complication in database integration is *heterogeneity*. This heterogeneity may be notational or conceptual. Notational heterogeneity concerns access language and protocol. One source is a Sybase database using SQL while another is an Informix database using SQL and a third is an Object Store using OQL. This sort of heterogeneity can usually be handled through commercial products (such as the Sybase OpenServer). However, even if we assume that all databases use a standard hardware and software platform, language and protocol, there can still be a conceptual heterogeneity, i.e., differences in relational schema and vocabulary. Distinct databases may use different words to refer to the same concept, and/or they may use the same word to refer to different concepts. Reassembling the distributed fragments of a database in the face of heterogeneity is doubly difficult.

Infomaster is an information integration tool that solves these problems. It provides integrated access to distributed, heterogeneous information sources, thus giving its users the desirable illusion of a centralized, homogeneous information system. Infomaster effectively creates a virtual data warehouse of its sources. The user does not have to be expert in accessing the diverse databases and yet the data does not have to be copied into one central location. (Data may however be cached for performance reasons.)

The next section gives some technical details about Infomaster. Section 3 is a detailed example of translation of heterogeneous sources by Infomaster. Section 4 is the conclusion.

2 Technical Details

The core of Infomaster is a facilitator that determines which sources contain the information necessary to answer the query efficiently, designs a strategy for answering the query, and performs translations to convert source information to a common form or the form requested by the user. Formally, Infomaster contains a model-elimination resolution theorem prover as a workhorse in the planning process. Figure 1 illustrates the architecture.

*This work was partially supported by CommerceNet and DARPA.

SIGMOD '97 AZ,USA

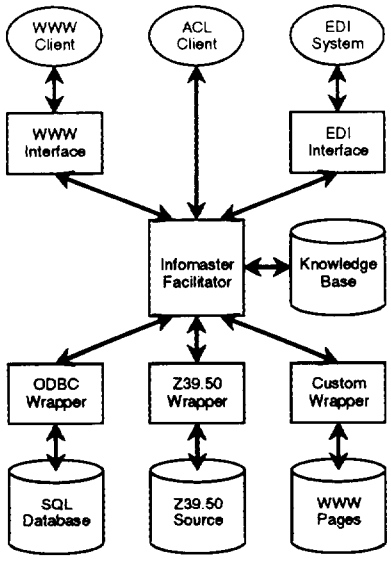

Figure 1: Infomaster Architecture showing the Infomaster Facilitator integration engine, wrappers for ODBC, Z39.50 and custom sources, and user interfaces for WWW, EDI, and ACL.

There are wrappers for accessing information in a variety of sources. For SQL databases, there is a generic ODBC wrapper. There is also a wrapper for Z39.50 sources. For legacy sources and structured information available through the WWW, a custom wrapper is used. For example, we use a custom wrapper to access housing rental advertisements from several San Francisco Bay Area newspapers on the WWW. The advertisements are then accessible through Infomaster using a forms-based WWW interface that supports structured queries.

Infomaster uses rules and constraints to describe information sources and translations among these sources. These rules and constraints are stored in a knowledge. For efficient access, the rules and constraints are loaded into Epilog, a main memory database system from Epistemics. Examples of the internal forms of these rules and constraints are given in the next section.

Infomaster includes a WWW interface for access through browsers such as Netscape's. This user interface has two levels of access: an easy-to-use, forms-based interface, and an advanced interface that supports arbitrary constraints applied to multiple information sources. However, additional user interfaces can be created without affecting the core of Infomaster.

Infomaster has a programmatic interface called Magenta, which supports ACL (Agent Communication Language) access. ACL consists of KQML (Knowledge Query and Manipulation Language), KIF (Knowledge Interchange Format), as well as vocabularies of terms.

Figure 2 shows the internal form of two of the rules that translate between the virtual data warehouse on top and the source data below.

Harmonizing n data sources with m uses does not require $n \times m$ sets of rules, or worse. By providing Infomaster with a reference schema, we allow database users and provides to describe their schemas without regard for the schemas of other users and providers. This strategy is shown in Figure 3. Translation rules describe how each source relates to the reference schema. These translation rules are bidirectional whenever possible, so information stored in one source's format may be accessed through another source's format. The same type of rules are used to describe how clients want to access data. These rules are combined and interpreted by Infomaster during query optimization and query processing. Because these translations reference each other essentially through the reference schema, entry and maintenance of translation rules is enhanced.

	Make	Doors	Seats	Range	MSRP
190e	Mercedes	4	4	400	35500
450sl	Mercedes	2	2	336	65500

```
(<= (doors ?x 4) (mercedes-type ?x sedan))
(<= (seats ?x 4) (mercedes-type ?x sedan))
```

	Mercedes Type	Mercedes Fuel	Mercedes MPG	Mercedes MSRP
190e	Sedan	16	25	35500
450sl	Sport	14	24	65500

Figure 2: Translation Rules showing translation of sedan in source database into 4 seats and 4 doors in virtual data warehouse.

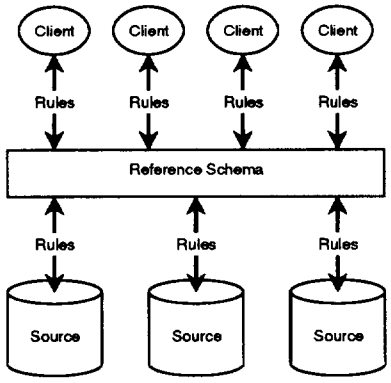

Figure 3: Use of Reference Schema to support efficient and maintainable translation rules harmonizing heterogeneous information sources and providing information as needed by each user.

The example in the next section shows how Infomaster can integrate heterogeneous structured information sources. However, Infomaster actually uses a peer-to-peer connection architecture. The rules can be used to direct updates from the clients to the relevant information sources, when permitted. Clients can register their interests and be notified when an information source changes in a way that intersects the registered interest.

3 Example of Harmonizing Source Heterogeneity

In this section, we illustrate the use of Infomaster with an example of four heterogeneous databases describing cars, and a virtual database integrating these four databases. The four databases are General Motors (GM) cars, Japanese cars, Mercedes cars, and Motorsport cars. Table 1 shows the virtual database combining all four databases. The columns are the car model, car make, number of doors, number of seats, driving range on a tank of gas, and the manufacturers suggested retail price.

	Make	Doors	Seats	Range	MSRP
Jimmy	Chevrolet	2	5	400	25500
Nova	Chevrolet	4	6	467	14999
Grandam	Pontiac	4	6	410	15999
Cutlass	Oldsmobile	4	6	440	17900
Accord	Honda	4	5	382	10364
Civic	Honda	2	4	410	8909
Corolla	Toyota	2	4	364	7636
Camry	Toyota	4	4	330	13636
190e	Mercedes	4	4	400	35500
450sl	Mercedes	2	2	336	65500
318i		2	4	640	25500
Targa		2	2	490	55500

Table 1: Virtual Cars Database

Table 2 shows the source database for GM cars. The attribute names need to be mapped to the virtual database. The values do not need to be mapped.

	GM Make	GM Doors	GM Seats	GM Range	GM MSRP
Jimmy	Chevrolet	2	5	400	25500
Nova	Chevrolet	4	6	467	14999
Grandam	Pontiac	4	6	410	15999
Cutlass	Oldsmobile	4	6	440	17900

Table 2: GM Cars Database

The following rule shows how the GM Make attribute of the GM cars database is mapped into the Make attribute of the virtual cars database.

```
(<= (make ?x ?y) (gm-make ?x ?y))
```

Table 3 shows the contents of the Japanese car database.

	Japanese Make	Japanese Doors	Japanese Seats	Japanese Km	Japanese Yen
Accord	Honda	4	5	612	1140000
Civic	Honda	2	4	656	980000
Corolla	Toyota	2	4	582	840000
Camry	Toyota	4	4	528	1500000

Table 3: Japanese Cars Database

Note that driving range is expressed in kilometers in the attribute Japanese Km, while the virtual cars database describes it in miles. The following rule does this conversion.

```
(<= (range ?x ?y) (japanese-km ?x ?km)
    (round ?km 1.6 ?y))
```

Also the manufacturer's suggested retail price must be converted from yen in the Japanese car database to dollars in the virtual car database. The following rule does this conversion.

```
(<= (msrp ?x ?y) (japanese-yen ?x ?yen)
    (convert ?yen yen ?y dollars))
```

Table 4 shows the Mercedes car database.

Note that there is no Make listed. It is assumed to be Mercedes for all cars in the database. The Mercedes car database has a type code, which is mapped into doors and seats in the virtual car database using the following 6 rules.

	Mercedes Type	Mercedes Fuel	Mercedes MPG	Mercedes MSRP
190e	Sedan	16	25	35500
450sl	Sport	14	24	65500

Table 4: Mercedes Car Database

```
(<= (doors ?x 2) (mercedes-type ?x sport))
(<= (doors ?x 2) (mercedes-type ?x coupe))
(<= (doors ?x 4) (mercedes-type ?x sedan))
(<= (seats ?x 2) (mercedes-type ?x sport))
(<= (seats ?x 4) (mercedes-type ?x coupe))
(<= (seats ?x 4) (mercedes-type ?x sedan))
```

The Mercedes car database separately shows fuel capacity and miles per gallon, and these are mapped into driving range in the virtual car database using the following rule.

```
(<= (range ?x ?y) (mercedes-fuel ?x ?f)
    (mercedes-mpg ?x ?m)
  (* ?f ?m ?y))
```

The Motorsport car database, shown in Table 5, consists of sporty German cars.

	Motorsport Doors	Motorsport Seats	Motorsport Range	Motorsport MSRP
318i	2	4	640	25500
Targa	2	2	490	55500

Table 5: Motorsport Car Database

The make of the car is not listed in the Motorsport car database. However, it is known that the database consists only of cars from BMW and Porsche. This fact is documented in the last two rules below. The following are the 5 rules describing where the Make attribute of the virtual car database gets its values.

```
((<= (make ?x ?y) (gm-make ?x ?y))
 (<= (make ?x ?y) (japanese-make ?x ?y))
 (<= (make ?x mercedes) (mercedes-msrp ?x ?y))
 (<= (make ?x bmw) (motorsport-msrp ?x ?y)
     (not (make ?x porsche)))
 (<= (make ?x porsche) (motorsport-msrp ?x ?y)
     (not (make ?x bmw))))
```

Because Infomaster does not know whether any particular Motorsport car is a BMW or a Porsche, Infomaster lists the Make in the virtual car database as blank. However, a query to obtain the cars made by BMW, Mercedes, or Porsche will result in Table 6, because Infomaster can determine that the Motorsport cars must be one of these.

	Make	Doors	Seats	Range	MSRP
190e	Mercedes	4	4	400	35500
450sl	Mercedes	2	2	336	65500
318i		2	4	640	25500
Targa		2	2	490	55500

Table 6: Result of Query for BMW, Mercedes, and Porsche Cars

The rules given above are in their internal forms as interpreted by Infomaster. System maintainers are expected to use a GUI to enter their rules using a spreadsheet metaphor.

While the last set of five rules are shown together, they are actually entered separately and incrementally with each data source. Infomaster assembles these into an efficient knowledge base for interpretation when a query is handled.

4 Conclusion

Infomaster is an information integration system developed at the Center for Information Technology of Stanford University.

Infomaster has been in use since fall 1995 for searching housing rentals in the San Francisco Bay Area, and since winter 1996 for room scheduling at Stanford.

Infomaster is the basis for the current Stanford Information Network (SIN) project that is integrating numerous structured information sources on the Stanford campus.

Infomaster is also the basis for the Housewares Virtual Catalog, a proof of concept with participants from Corning, Mirro, Regal, Sears, GE Information Services, National Housewares Manufacturers Association, National Retail Federation, Stanford University, and Epistemics.

Stanford's Infomaster service can be found at `http://infomaster.stanford.edu`

Infomaster is now being commercialized by Epistemics. Epistemics can be reached at `info@epistemics.com` or `http://www.epistemics.com`

Acknowledgments

The authors would like to thank the members of the Center for Information Technology for their assistance and feedback in developing Infomaster.

References

[1] Oliver M. Duschka and Michael R. Genesereth, "Query Planning in Infomaster," *1997 ACM Symp. on Applied Computing*, February 1997.

[2] Donald F. Geddis, Michael R. Genesereth, Arthur M. Keller, and Narinder P. Singh, "Infomaster: A Virtual Information System," *Intelligent Information Agents Workshop*, at CIKM '95, December 1995.

[3] Arthur M. Keller, "Smart Catalogs and Virtual Catalogs," *Readings in Electronic Commerce*, Ravi Kalakota and Andrew Whinston, eds., Addison-Wesley, 1997, pp. 259–271.

[4] Arthur M. Keller and Michael R. Genesereth, "Multivendor Catalogs: Smart Catalogs and Virtual Catalogs," in *EDI Forum, The Journal of Electronic Commerce*, Vol. 9, No. 3, September 1996, pp. 87–93.