DTL's DataSpot: Database Exploration as Easy as Browsing the Web...

Shaul Dar Gadi Entin Shai Geva Eran Palmon

Data Technologies Ltd. 3 Tevuot Ha'aretz St., Tel Aviv Israel 69546, +972-3-6471661, {dar,gadi,eran,shai}@dtl.co.il

1. Abstract

DTL's DataSpot is an advanced, programming-free tool that lets Web designers and database developers automatically publish their databases for Web browser access. DataSpot enables non-technical end users to explore a database using free-form plain language queries combined with hypertext navigation, in a fashion similar to using search engines such as Alta Vista to search text files on the Internet.

DataSpot is based on a novel representation of data in the form of a schema-less semi-structured graph called a Web View. The DataSpot Publisher takes one or more possibly heterogeneous databases, predefined knowledge banks such as a thesaurus, and user-defined associations, and creates the Web View. The DataSpot Search Server, which connects to any standard HTTP server, performs searches and navigation against the Web View, generating dynamic HTML pages that are returned to the user. The presentation and navigation of answers are controlled by templates that can be modified by the data provider.

The DataSpot product has been successfully deployed in diverse Internet and Intranet application areas, including electronic catalogs, yellow pages, classified ads, help desks and finance.

1.1 Keywords

Internet, Database Publishing, Search, Navigation.

2. Introduction

Database publishing on the Web is a major issue in today's world. The reasons are clear: the vast amounts of important business information that resides in corporate databases, the explosion of the Internet and in particular the familiarity of Web browsers as standard user interfaces. A prime example of an Internet application is an electronic catalog. A prime example of an Intranet application is a help desk, where a company's support representative receives customer inquiries and must quickly locate relevant information in an appropriate database. These examples demonstrate the need for a standardized, intuitive and friendly way for users to search online databases (see e.g. [3]). It is this need that DTL's DataSpot seeks to address.

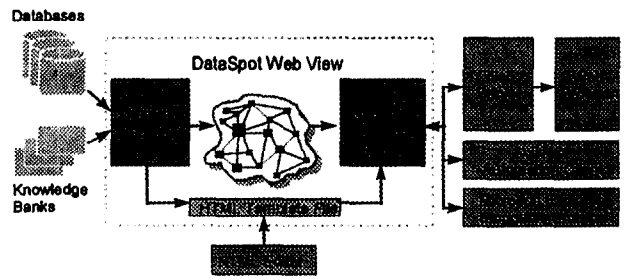

Figure 1: The DataSpot Architecture

3. An Overview of DataSpot

The DataSpot system provides end users with the capability of exploring databases using free-form queries and navigation. DataSpot does not provide this capability by a "thin" interface layer. The DataSpot Publisher actually translates the source data into a novel, schema-less representation called a Web View, which in turn may be queried efficiently by the DataSpot Search server (see Figure 1). The Web View is a universal representation of data as a graph of associated elements, tailored to support free-form query algorithms. This representation lends itself naturally to a hypertext presentation and navigation via the mapping of associations to links. The model is general and could represent different data models. In particular, the relational model is represented through associations such as foreign keys, associations between a record and its fields and between a field and its data values. Linguistic data, e.g. morphology and thesauri,

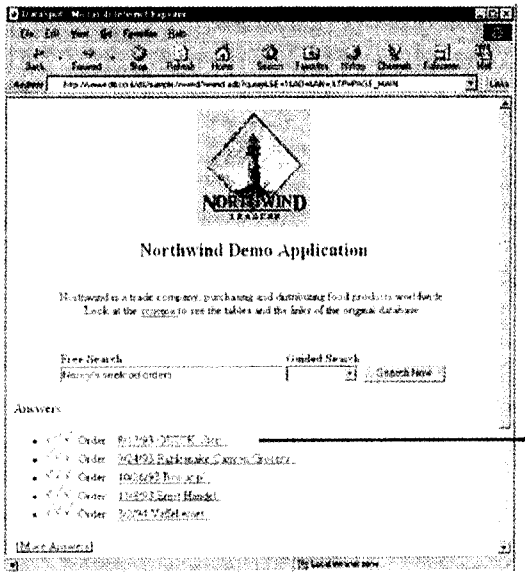

Figure 2: The Query "Nancy's Seafood Orders"

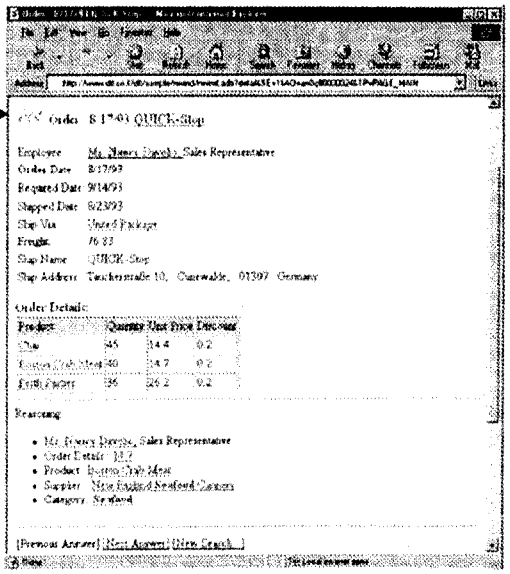

Figure 3: An Answer

is likewise represented by associations. Within the relational framework free-form queries can be understood as follows. The input to a query is a set of words. The search process finds answer records that are related to the query words through relational or linguistic associations. The quality of an answer is based on the strength of the associations used to derive it. These associations may optionally be displayed as justification to the answer. Given an answer record the user may navigate to related records. The user may also submit continuation queries whose input consists of words and of nodes in the graph corresponding to the current answer record or set of answer records.

As a simple example consider Microsoft's Northwind Traders database, a demo corporate database consisting of eight tables, namely employees, orders, order details, customers, products, suppliers, shipper sand categories tables (for the schema and a demo see www.dtl.co.il/dtl/sample/nwind/nwind.adb). The query "Nancy's seafood orders" returns (as an HTML page) several records from the orders table (see Figure 2). The first record, for example, represents an order for a customer named QUICK-stop, processed by the employee Ms. Nancy Davolio, where one of the products in the order, Boston Crab Meat, is of category Seafood, and is also supplied by a supplier named New England Seafood Cannery (see Figure 3). Note that finding this answer required five relational associations ("joins") that are shown under Reasoning at the bottom of Figure 3, as well as linguistic associations used e.g. to connect "Nancy's" with "Nancy" (linguistic associations may also include related words and concepts associated via a thesaurus). Note also that the words "Nancy's" and "Seafood" relate to values in the database while the word "orders" is related to a metadata element (the order table), but the DataSpot user need not be aware of this distinction.

The first commercial release of DataSpot, DataSpot 1.1, has been available since April 1997 on Windows NT/95 platforms. It supports all major database products as well as flat files. DataSpot 2.1, currently in pre-release state, offers several major enhancements: an online update capability to automatically keep the Web View synchronized with the source databases, an object interface to allow developers to integrate DataSpot searches into their applications, support for all major European languages, and major performance enhancements including a multi-threaded engine. As of this writing, several dozen DataSpot 1.1 and 2.1 systems have been deployed in varied application areas including electronic catalogs, yellow pages, classified ads, help desks and finance.

More information about DTL and DataSpot can be found at www.dataspot.com. In particular, see the DataSpot applications page for examples of applications built using DataSpot.

4. Related Work

Many products address the problem of providing a friendly way for users to retrieve information from databases. In general, these products fall into three categories: text search engines, form-based interfaces and natural language interfaces.

Text-search engines (e.g. AltaVista, Yahoo!, Excite, InfoSeek, Lycos, Verity). These products provide retrieval of structure-free, text-based information. The criteria for the search are usually that the requested words appear in the same proximity in the document. While such systems could in principle be used to search a copy of a database exported into one or more text files, this translation loses the semantics embedded in database tables and relationships. Consider again the query "Nancy's seafood orders". Furthermore, text-search engines do not support navigation between related data elements.

Form-based interfaces (e.g. Saphire/web, Cold Fusion, NetDynamics, dbWeb). These products offer a user interface based on structured forms. This approach has several major drawbacks. First, it requires a programming effort on the part of the data provider: forms must be created, translated to SQL, and Web enabled. Second, it limits the user to requesting data via a particular set of queries. For example, the query "Nancy's seafood orders" would require a specific form where an employee's name and a product category may be entered, and where matching records are projected on some fields of the orders table. Third, the input must be specified exactly, e.g. linguistic associations would be difficult to incorporate.

Natural language Interfaces (e.g. English Wizard). These products offer a natural-language interface to the user and then translate the query into SQL statements. This approach looks similar to ours in that it supports free-form queries and it uses an external index on the published data values (e.g. to guide the translation of the user's query), but it passes query evaluation to the database query processor. The DataSpot Web View representation provides important advantages, such as the ability to integrate heterogeneous databases, extendibility (e.g. non relational data, additional languages and associations), better performance (e.g. joins are "pre-computed" and search algorithm is specialized), ability to justify answers (as opposed to just showing the generated SQL) and navigation capability.

Representation of unstructured or loosely structured data has received much attention in the database research community in recent years (see e.g. [1], [6], [8]). For lack of space we mention only a few salient points regarding the relationship of the DataSpot data model (the Web View) to these proposals. First, the emphasis of the DataSpot representation is to enable free-form queries, while most of the above mentioned research work is focused on formal query languages for semistructured data. Second, the DataSpot model is different than the above proposals, and it addresses some important theoretical and practical questions such as the representation of cyclic data and the integration of information from heterogeneous data sources, including relational and non-relational databases, a thesaurus and user-defined associations. Third, a lot of attention has been paid in DataSpot to the efficient implementation of the graph, to the acceleration of the bulk load process that creates it and to the tuning of the query process that traverses it, including support for multi-threaded queries concurrent with updates.

Other work in the database research community addresses ways to organize and query data on the Web (see e.g. [2], [3], [5], [6], [7]). These studies are complementary to our work.

5. Acknowledgments

We are grateful to Divesh Srivastava and S. Sudarshan for providing valuable feedback on this work.

6. References

References to commercial products are omitted. Information can be found at the respective Web sites.

[1] P. Buneman, S. Davidson, G. Hillebrand and D. Suciu. A Query Language and Optimization Techniques for Unstructured Data. SIGMOD 1996.

[2] Mary Fernandez, Daniela Florescu, Jaewoo Kang, Alon Levy, Dan Suciu. System Demonstration - Strudel: A Web-site Management System. SIGMOD 1997.

[3] Maurice Frank, Editor. Searching Text and Tables. Internet Systems (DBMS Magazine supplement), October 1996.

[4] D. Konopnicki and O. Shmueli. W3QS: A Query System for the World Wide Web. VLDB 1995.

[5] Alon Y. Levy, Anand Rajaraman and Joann J. Ordille. Querying Heterogeneous Information Sources Using Source Descriptions. VLDB 1996.

[6] J. McHugh, S. Abiteboul, R. Goldman, D. Quass, and J. Widom. Lore: A Database Management System for Semistructured Data. SIGMOD Record, 26(3): pgs. 54-66, September 1997.

[7] A. Mendelzon, T. Milo. Formal Models of the Web. PODS 1997.

[8] Y. Papakonstantinou, H. Garcia-Molina and J. Widom. Object Exchange across Heterogeneous Information Sources. ICDE 1995.