QBI: Query By Icons

*Antonio Massari** *Stefano Pavani** *Lorenzo Saladini** *Panos K. Chrysanthis***

* Dipart. di Informatica e Sistemistica, Universita' di Roma "La Sapienza" Via Salaria, 113 - 00198 Roma, Italy

** Dept. of Computer Science, University of Pittsburgh, Pittsburgh, PA 15260, U.S.A.

Email: massari@infokit.dis.uniroma1.it, panos@cs.pitt.edu

1 Overview

QBI is an icon-based query processing and exploration facility for large distributed databases [3]. As opposed to other interactive query interfaces, it combines (1) a *pure iconic specification*, i.e., no diagrams of any form, only icon manipulation, with (2) *intensional browsing* or *metaquery* tools that assist in the formulation of complete queries without involving path specification or access to the actual data in the database.

Path expressions are automatically generated by QBI and irrespective of their length, represented by a single icon, allowing for better use of the screen. It requires no special knowledge of the content of the underlying database nor understanding of the details of the database schema. Hence, QBI is domain independent and equally useful to both unsophisticated and expert users.

1.1 User's Perception of the Database

In QBI, the concepts of *class of objects* and *attribute of a class* exclusively form the external representation of the database structure. QBI visualizes the entire database within each object, expressing object classes as *generalized attributes* (GAs) of the object. That is, GAs encapsulate both implicit and explicit relationships among objects within each object. Thus, in QBI, each object can be thought of as an independent "universal relation" [2]. QBI uses icons to represent both classes and GAs. The implicit ambiguity of iconic representation is resolved by automatically generated natural language descriptions.

1.2 QBI's View of the Database

Internally, QBI uses the Binary Graph Model [1, 3], a semantic data model, for capturing the structure of the underlying database. A GA is strictly related to the concept of *path* connecting nodes representing classes of objects in a typed-graph that represents the internal schema. Since not all paths are equally meaningful, and in order to cope with cyclic paths when generating GAs, QBI defines a *semantic distance function* on paths which returns a value for each path representing the meaningfulness of the corresponding GA for the specification of a query. The semantic distance is expressed in terms of various aspects of the structure of the GA such as its length, the number of cycles, inclusion/exclusion of specific paths, cardinality constraints as well as statistical information.

1.3 Query Language

The query language of QBI is based on the select-project paradigm: a query is constructed by first choosing a class node whose GAs are used to define the selection condition; then composing the selection condition (predicate) by appropriately connecting their corresponding icons; and finally specifying those GAs that are going to be part of the output result.

Every query results in a subclass derived from the chosen class and as such it is added to the typed-graph to be used subsequently in the composition of more complex queries.

2 The Prototype

The prototype of QBI to be demonstrated in SIGMOD 95 has been developed in *C* for MS-Windows using the XVT toolkit, and it is tailored to provide access to a radiological database containing data about doctors, patients, hospitals, lab tests, radiological images, etc. It consists of four modules: The *Presentation Manager* which structures the interaction with a user around three windows, each dedicated to a specific aspect in the specification of a query (the user interacts with the system mainly by dragging and arranging icons); the *Query Manager* which supports the specification of queries; *GA Evaluator* which computes the generalized attributes and their associated semantic distance value; and *Database Access Manager* which is responsible for any access to the database as well as managing the metadata.

Currently, the GA evaluator of the prototype is being enhanced with the goal to evolve QBI into a query interface for mobile database applications [4].

References

[1] T. Catarci and G. Santucci. Fundamental Graphical Primitives for Visual Query Languages. *Information Systems* , 18(3), 1993.

[2] D. Maier and J.D. Ullman. Maximal Objects and the Semantics of Universal Relation Databases. *ACM Trans. on Database Systems*, 8(1):1-14, 1983.

[3] A. Massari and P.K. Chrysanthis. Visual Query of Completely Encapsulated Objects. *RIDE-DOM 95.*

[4] A. Massari and P.K. Chrysanthis. An Iconic Query Processing Facility for Mobile Users. *UPitt CSD TR-95-10.*

SIGMOD ' 95,San Jose , CA USA