A model for intuitive Knowledge Sharing

Pedro Furtado and H. Madeira
pnf@dei.uc.pt

Abstract

This paper proposes a system which eases the job of entering and sharing expert analysis on a database system, with emphasis to pictorial and document information. With this system a medical doctor expert on the analysis of certain imagery can not only do the normal browsing/querying but also interact with the images in a human-like fashion storing his analysis to be shared in a PACS system. Knowledge information is organized in dynamically configurable nested collections hierarchy with categories and registries that allow a web-based intuitive navigation and interaction with the system. The system combines hyperlink navigation applied at all levels with user-friendly advanced visual query facilities. It also includes facilities for extended semantics. It is cooperative, helping the user to choose the next step. The intelligent conjunction of all those features creates a very useful system. We conceive it as a possible add-on to any existing database. It is called KnowShareSys or KSS, synonyms for Knowledge Sharing System .

1 Introduction

Cases usually provide the basis for teachers to present a subject and for students to learn that subject. Learning, improving skills, and sharing knowledge are usually processes that go through many stages of data relating and partial deductions and conclusions. Analysis sharing and discussion forums are an important add-on to archival databases such as picture archiving and communication systems (PACS). Visual information management systems (VIMS) should have those facilities. Image management and its relation to the rest of the information is a very important issue today, and in medical informatics standards are actively pursued by HL7 Image Management SIG (IMSIG) in the framework of health information standards. Todays' hypermedia systems organize and access sparse information in a very intuitive way, linking all types of information, while database systems are widely used and offer efficient data storage and query capabilities for large repositories of data but lack the semantic linking capabilities of hypermedia. Database browsers usually allow comprehensive object browsing and advanced query capabilities, but often require the user to be aware of at least the objects/entities. The access mechanism offered by the system is limited in what concerns knowledge representation, archiving and sharing. There is a need for an integrated system that adds hypermedia-like object relating and accessing in a database.

The goal of our system is to provide a most intuitive graphical user interface and extensible database organization that can be used in any system to allow the sharing of knowledge associated with pictorial analysis and document information in a specific professional domain. The new architecture gives emphasis to some simple principles: A visual most intuitive cooperative interface which helps the user categorize and organize analysis information to be shared, gives generalized access to allowed users, web-like navigation and wide applicability. It extends the database schema to include a layer of referencing objects and a layer of knowledge information in the form of abstract concepts, simple links and also generalized relations between any number of objects. Our system creates and semi-automatically maintains dynamic registries and directories to effectively organize all the objects to be shared and allow other users to navigate that knowledge. Hyperlink jumping is used not only between explicit links but also between any two database objects that are related and also to view objects (where object can be of any kind, including images, regions-of-interest within images, etc). Complex database queries can also be used. This web-based interface means a universally accessible system which supports distributed database servers. We integrate all these features in one system in a way that turns it into a very innovative and complete system. We are building a prototype for a medical imagery system in which the facilities will be applied.

The rest of this paper is organized as follows. In section 2 we briefly review related subjects like user interfaces, pictorial database systems, issues in semantic data models and hypermedia to point out the importance of our system. Section 3 introduces the functionality of the system. Section 4 shows how the system works to allow such functionality. Section 5 briefly clarifies some implementation issues, Section 6 discusses future work and Section 7 concludes.

2 Related work

In this section we review some trends in the related areas of user interfaces, pictorial database systems, semantic data models and hypermedia, and highlight the relevance of the KSS system.

The importance of graphically-oriented specialized end user-interfaces is emphasized in [Ston 93]. Some paradigms, languages and systems appeared first for relational systems, like QBE [Zloo 77], SDMS [Herot 80], Timber [Ston 82], Formanager [Yao 84], LID [Fogg 84], FADS [Rowe 85], ISIS [Goldman 85] and proved to be more productive than SQL. [Batini 91] offers a survey on systems implementing visual query languages. Such languages usually express queries by following diagram links, filling forms or by using icons. Today many of them are object-oriented. Examples of visual interfaces of those kinds are KIVIEW [Mottr 88], Pasta-3 [Kunt 89], OdeView [Agra 90], PESTO [Carey 96]. In [Catarci 96] a multiparadigmatic approach with several visual representations is used to access databases according to the user needs.

In [Catarci 95] 4 concept types are identified to represent information through the data model: Objects of the real world and their properties, classes of objects, relationships among objects and relationships among classes of objects. Semantic Data Models (SMD) [Hammer 81] aim at efficiently representing knowledge information and can be implemented in an Entity-Relationship model [Hull 87] [Nijssen 89] [Peckham 88]. Usually semantic information is added as a layer of abstraction on top of the relational model. [Mohan 93] points out the expressive power of any query language is limited by the semantic depth of the underlying data/knowledge model, and it is argued that the semantic knowledge captured in a traditional relational database is limited due to its tabular structure [Codd 79], [Peckham 88]. The paper on GUIDANCE [Haw 94] also reviews some problems with user interface and database systems, specially in what concerns the expression of semantic knowledge. Users access a flat table structure and are required to reconstruct the semantic information using their understanding of the context. Instead, the system should be able to conceal the relational structure to the user accessing knowledge information and present that information in an intuitive way. Rich models as object-oriented and semantic databases are used to capture further semantics from the application domain. But then the number of objects and links can become a very complex web and intuitive access becomes more difficult.

Visual query systems were also developed for pictorial databases in ways similar to QBE and QBForm and picture object retrieval techniques were also developed. These Content Based Image Retrieval techniques (CBIR) include Query by Sketch (QBS) found in QBIC [Faloutsos 93] [Flickner 95], or Query By Pictorial Example (QBPE) in Virage system [Gupta 95]. These and other systems like Berkeleys Digital Library project retrieve objects based on shape, color, texture or spatial relations [Carson 96].

Semantic modeling and queries with pictures are used in VIMSYS [Gupta 91], the PICQUERY+ system [Cardenas 93], and KMed [Chu 94] . In those systems picture objects and relationships are important semantic entities. The VIMSYS model defines three layers: The image representation layer with the image matrix, the image objects layer which individualizes the image objects, and the most abstract domain knowledge layer. [Goble 96] also focuses on the problem of describing and classifying pictorial information. The authors point out that the description of image contents must incrementally evolve to link the image instances with associated descriptions and so must not be predefined. Semantic annotations link objects to collection of domain concepts, bringing media to world concepts translation.

Besides the user interface and the functionality of the system, distributed storage and access are another important issue in database systems, posing new challenges [Dayal 84], [Batini 86], [Sheth 90], [Ahmed 91], [Zhou 94]. Hypermedia systems combine multimedia richness in datatypes with an elegant way of navigating through data in a content-based manner in a decentralized environment [ACM HyperMedia 94], and even address the semantics of time and context [Hardman 94]. Many Hypertext systems exist that provide the user with the ability to create, manipulate and examine a network of information-containing nodes such as NoteCards [Halasz 88], KMS [Akscyn 88], DHT [Noll 91], InterMedia [Haan 92]. Multiple servers contain nodes linked among them and often use a database to store links and node locations [Noll 91] . [Vazirgiannis 95] presents a model that integrates semantics and multimedia information. Hypermedia systems focus on linking diverse information in a distributed environment while database systems focus on the efficient access to large repositories of data.

In our system Knowledge coexists in regular documents, images, sketches and also as semantic database concepts and objects. We include some picture object outlining functionality as graphical objects description and analysis are a very important feature of the system. We concentrate on the structure of the system and the user-interface. The system will progressively include more features of pictorial information management and transparent semantic data handling that will be added to the basic functionality.

3. Functionality of the System

The model is useful in domains that deal with knowledge information of many kinds, from abstract concepts to documents, images or sketches, specially when the contents of the pictorial information is important. In the case study we had the help of potential users from the medical professional domain who helped us identifying some important knowledge data and interactions together with the prescribed database schema. In addition to the usual database facilities the KnowShareSys should offer at least the following functionalities:

- Allow the individualization of regions-of-interest (ROI) in imagery and the definition of objects for those ROI, with fingerpointing for the points of interest/objects from textual or pictorial data and allowing to enter definitions, analysis and relationships for all the data;
- Allow the user to see information related to an image, possibly superimposed on the image (for instance, retrieving the diagnosis information);
- Allow the user to define new categories of concepts, like anatomical parts, to relate them to other objects;
- Keep dynamically updatable directories of documents and tutorials related to the information in the database and somehow link them;
- Keep directories to categorize the information in collections - by subject, by type, by user, etc;
- Keep registries of every item in the extended database. Those registries should be easy to browse;

- Allow the experts to store drawings and sketches that they may relate to information so they can complement their analysis and tutorials;

These topics suggest the user needs a wide range of facilities that go much beyond the functionality of a database system. Web-browsers offer the facilities to allow the editing of documents in different formats through the invocation of the required application, so KnowShareSys takes advantage of those facilities and focuses on adding loose knowledge from many sources to a large database in an organized way and providing the basic knowledge access and management functionality that the user requires, all dynamically scaleable.

Hypermedia services aim at allowing the complete linking functionality of different media by using links and anchoring. The Standard Generalized Markup Language - SGML or the within component layer of the Dexter hypertext reference model [ACM HyperMedia 94] are designed to access the structure of documents or other media, but for anchoring and linking the media editors used must cooperate with the system. It is worth noting that for a deeper functionality every document/image of any format should be decomposable into objects of any required depth into its structure for database awareness (for instance, to find the place where a practical case is described) and there could be some way of relating document items to the extended database objects. The access and analysis of images is an important part of our system, so we allow the user to point an image from the database repository, define regions and objects related to that image, relate a document or simple text to the image or to an object in the image. In this first phase we do not try to implement the full functionality of hypermedia systems or content-based image retrieval (CBIR) as our system is centered on the database functionality and on keeping registries of information related to the database.

4.Components of the System

Every item in KSS is an object and objects can be referenced, take part in any relation and be linked to any other object. Figure 1 shows a diagram of the system centered around an all-objects view. We have seen that he VIMSYS model defines three layers for image management. In KSS we can also consider three layers for convenience: raw data refers to indistinct data in the database and corresponds to schema data objects in Figure 1 (table/object data). The second layer consists of referencing objects as seen in Figure 1. These objects either "point" to objects or individualize parts of objects, either schema data objects or any other type of object. Any raw data object can be registered to KSS, creating a referencing object. Finally the knowledge layer consists of relations, links, categories and registries respectively relating, linking, categorizing and grouping any object. This layer is expressed in Figure 1 through the category and registry objects and also with the help of some referencing layer objects such as relationObj and linkObj.

One of our concerns in this system is to allow concept types in an intuitive manner using a visual interface that helps the user with his queries and browsing. Objects, classes of objects and any concept are defined and shared through the registry and category objects, while general-purpose relations are registered in the relationObj object. Sometimes the user desires to access the database in a "flat way", that is, accessing the objects/tables explicitly, but frequently the user is interested in "talking to familiar concepts". So, the database and user interface should provide both worlds in an integrated fashion.

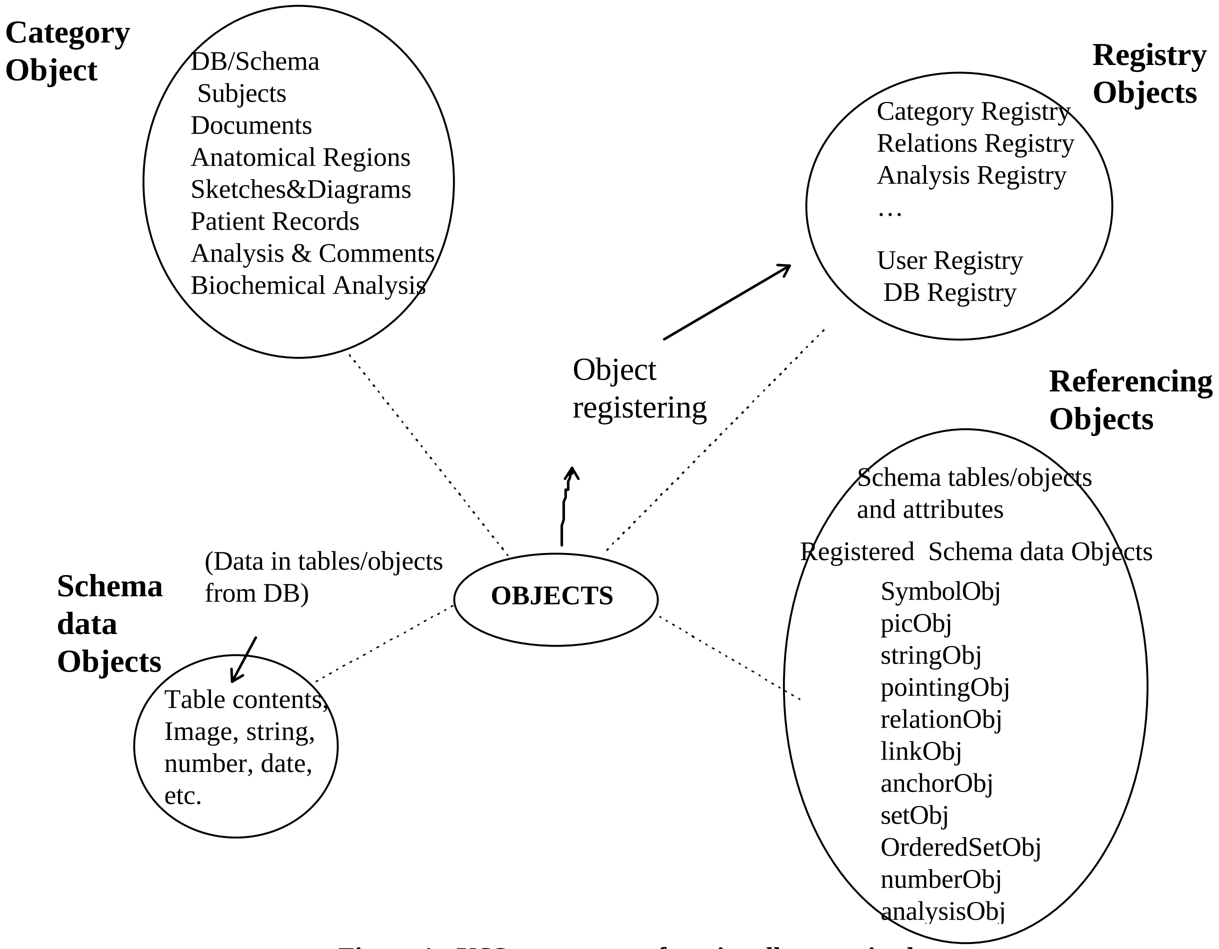

Figure 1 - KSS components functionally organized

4.1 Objects

Schema data objects are the data in the objects/tables in the database. The other objects belong to the dynamic schema extension. Schema extension refers to objects that implement a more complex semantics or help on that task. Referencing objects possess a specific data structure to denote a part of an image, to hold a string or to implement any other extended functionality. LinkObj objects keep simple links between two objects and relationObj objects express relations of any degree of complexity between any number of objects. Registries keep track of all the objects in the system and at the same time group them according to their type and functionality. Specific features for handling objects of a given type are encapsulated in the procedures that access each registry. Categories are dynamically configurable directories of every possible item in the system. This way users can organize their important multimedia and semantic objects in easy to browse directories. As seen in figure 1 Categories can include Subject and Document subdirectories for instance. Each individual document node will be represented through a referencing object registered in the registry object. That registry includes an URL to the document or the reference to the object in the database schema if it is there. A Subject is an abstract concept represented as a symbolObj that will be used to aggregate the data in the categories subdirectories.

4.2 User Interface and object exploration

The user interface is a very important component of our system because it is targeted at people with no expertise in complex database access languages. The user will quickly learn that if he wants to do something with an object anywhere he will most probably have to click on it for a start and then relations and links having that object will be suggested to him through a registry window or otherwise an object will be shown. Users need multiparadigmatic access [Catarci 96] to all kind of objects, because they have several different objectives. Figure 2 shows a simplified structure of the interaction with the user. He can choose different ways to start his search, switch among them, jump to another window at any time by clicking on an object or link or have multiple windows simultaneously. In Figure 2 Schemas allow the user to access the database by choosing one schema table/object to view the corresponding data objects. Registries allow him to search in a particular type of referencing object while categories allow him to navigate through a directory of relevant objects. In all the windows many objects/data values retrieved in the tables are also hyperlinks to further refinement. This happens when links or relationships were registered for the object, or if it is a referencing object (when the user doubleclicks on an image reference in an object window the corresponding image is displayed). Schema defined relationships like primary/foreign keys are used to automatically establish links and the user can establish other links and relationships manually between any kind of objects. Category and registry items also link automatically to the referenced objects (the user can also bookmark URLs for direct access). These cooperative features allow any user to be successful even without any experience using the system. We present similar user interfaces for the access to any object, including Registries. This interface presents each Object in tabular form, with the attributes as columns (Figure 3), allowing selective searches. All semantics and objects can be available through directories in the categories window, or in tabular form organized by type in the registry window. The user can use advanced queries using the optional Filtering popups and Chosen values table (Figure 3). Filtering popups restrict the values shown for each attribute and have the options - none, all, by keyword, pick from the existing ones, similar-to. After choosing some data in the filtering popup, the user simply clicks on the chosen values table to activate the chosen filter component. The user can dismiss the Filtering popups and Chosen values table at any time if he doesn't need them. He can also establish a visual join between any two object tables based in some linking condition.

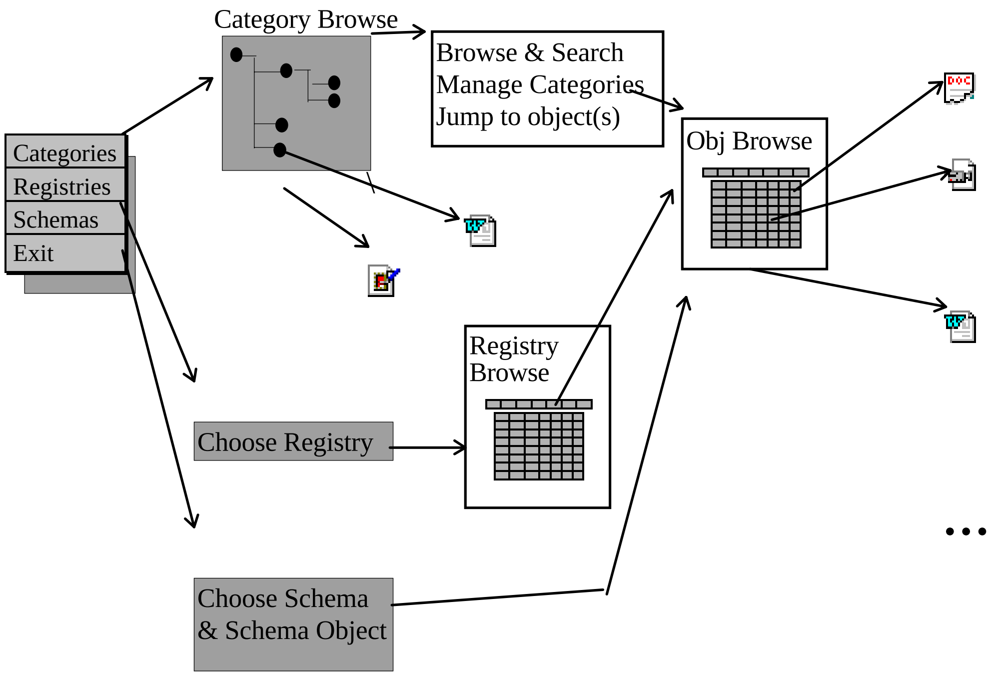

Figure 2 The KSS access mechanism

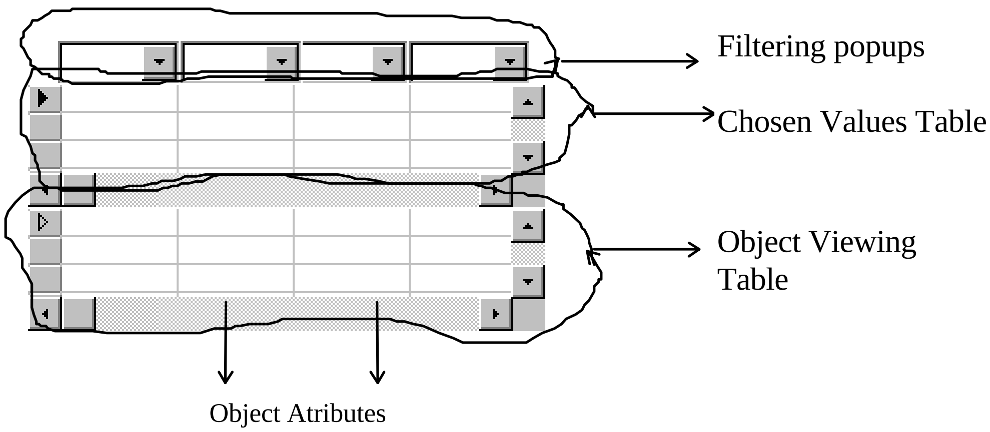

Figure 3 KSS Object windows User-Interface parts

4.3 Extended components

4.3.1 Registries

Registries are at the heart of the system. They are entities that keep track of all the registered objects in KSS and assign a unique OID to each. Every referencing object is automatically registered when it is created. Each Registry accesses base tables containing the registered items of a datatype. A Registry determines the functionality of its objects and keeps specific data needed to handle those objects. For instance, in Figure 4 the database registry contains information necessary to access different database nodes. In that Figure the user has chosen some filters in the chosen values table using the filtering popups (for instance, only drivers including the letters ORA and 73 or SQL are shown). At any time the user can jump directly to objects, categories, another registry or elsewhere, optionally keeping the restrictions he had imposed. To see objects related to the active one the relations and link registry windows are automatically called showing only those that include the object.

The schema is automatically registered to the system. Regular schema data items are registered to the system once they become of interest through the indication of the user. Extended semantic objects are registered immediately when they are created. There is a categories registry also.

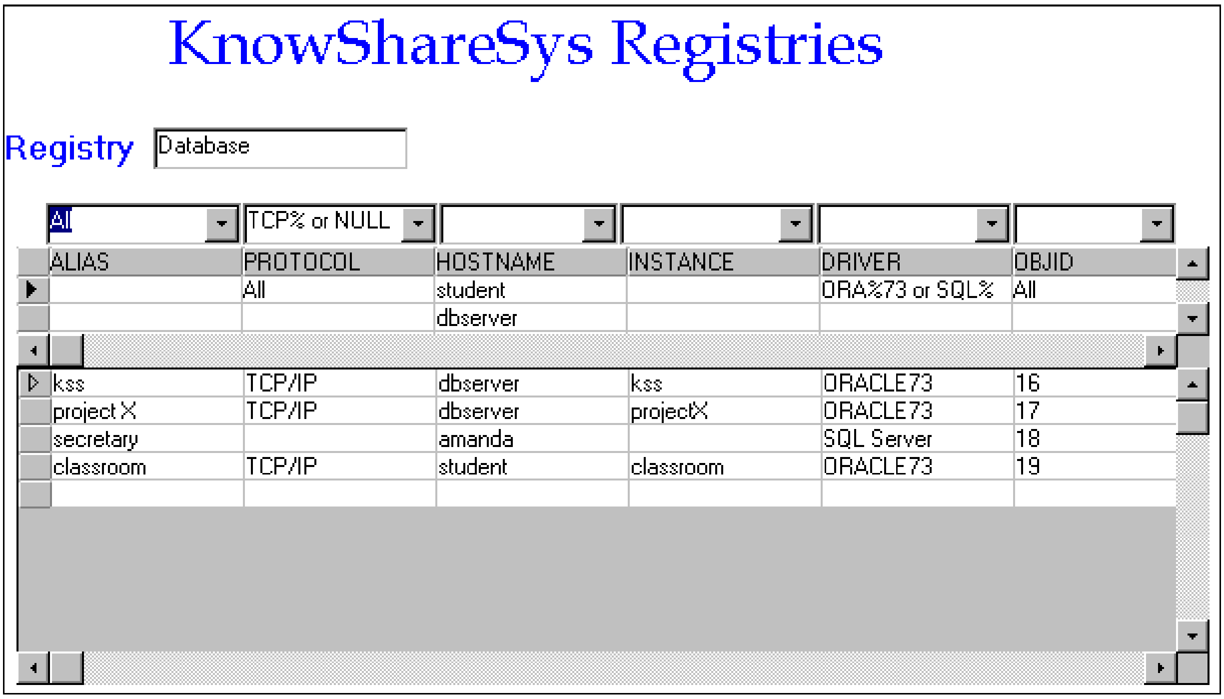

Figure 4 Database Registry Window showing the user interface in the KnowShareSys

4.3.2 Categories and Category directory

Figure 5 shows the category directory. The directory is organized by levels. In the screen shown the user chooses what to see using the popups and then transfering the chosen filter to the chosen values table. At each category level the user can restrict the subcategories to see expanded. The Category object keeps a structured directory tree of every desired item (either a subcategory or an end-object, for instance documents, subjects, anatomical parts, abstract concepts, etc). The specific categories depend on the application domain. Categories don't include only objects residing in the local database but also objects from other database instances and pointers to files in the file system. Schema definition objects/tables and attributes are automatically included and correctly indented in the category tree, because they are preexistent concepts. A menu adds all the required functionality to allow a user to use and dynamically manage the Category tree allowing creation, deletion, renaming and modification of any category (depending on privileges), optionally creating an associated schema object if desired. As with all the other object types, there is also a categories registry with the information necessary to manage the tree. When the user creates a new category he can either search for the object to be referenced or else create a new symbol corresponding to a new concept. In the second case a new symbolObj object is created with an OID assigned to it. In both cases the category registry keeps track of the object through the OID and position in the directory tree. The user can navigate KSS through the category tree at any time, easing the job of finding a single object or a group of objects. As with the rest of the user-interface, advanced search is also available. When the user clicks on a sub-category or object he jumps to the target chosen (viewing the image, document, table/object, attribute or abstract concept properties). Using links and relations he can then access other objects.

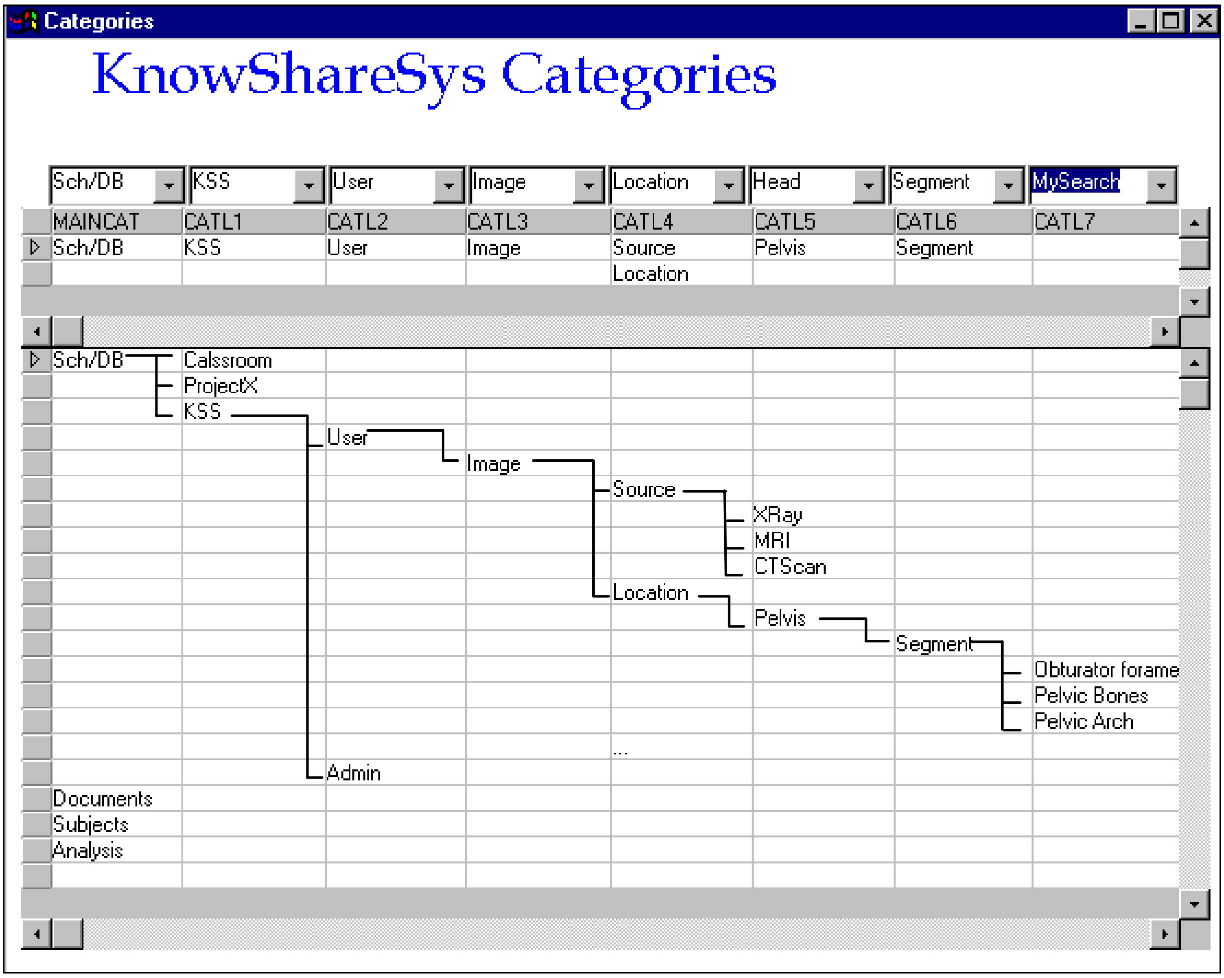

Figure 5 Categories Window in the KnowShareSys

4.3.3 Relationships and links

Database systems and semantic models are based on data objects and relations among those objects. Registries and Categories keep track of the objects in a satisfactory way. Relationships keep track of any relations between any number of objects through the relations registry. Our system allows dynamic definition of any relationships between any number of objects of any type. In Figure 6 the user is viewing the relations containing the words beginning in “Obturator” or “Acet” (in chosen values table, under TITLE). A relation includes any number of objects. In Figure 6 the “obt_in_pelvis” relation has the title “Obturator foramen in Pelvis”, includes the objects “Obturator foramen” and “Pelvis”, has a cardinality 1x1 and belongs to a more general class of relations called “is-part-of”. The user can access relationships directly through the Relations registry or he can ask for the relations and related objects when he is viewing one particular object. In the first case he simply uses the relations registry or the category registry to retrieve a relation. In the second case, to ask for related objects when viewing one, the relations registry opens for those relations containing the current object. This way the user can choose the desired relations or object(s) to browse and then jump to them by clicking on the object. Presently, links are simple relations between two objects allowing jumping, but we wish to extend the concept to include anchors and more advanced hypermedia features.

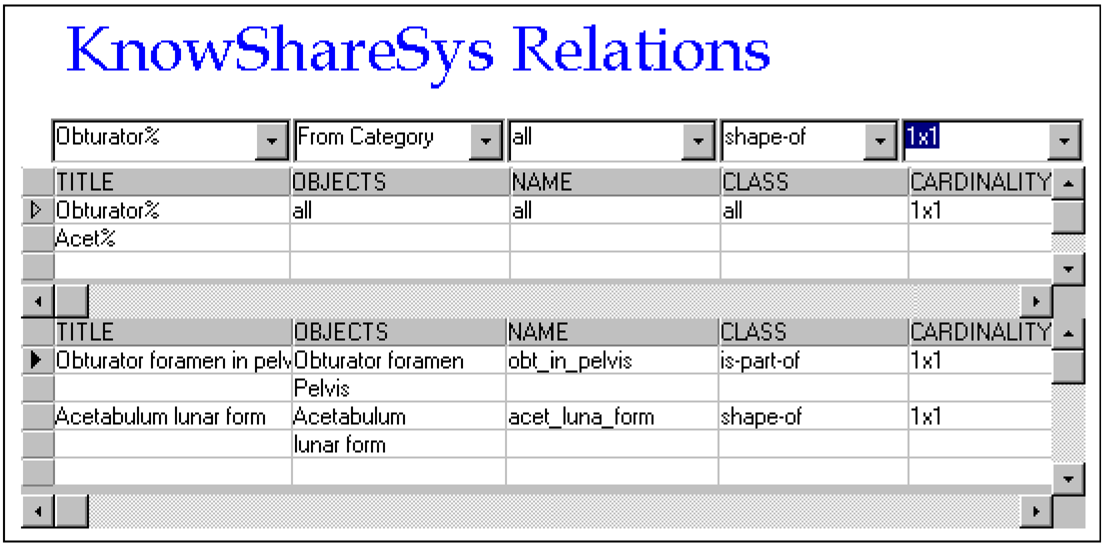

Figure 6 Relations Window in the KnowShareSys

5. Implementation issues and specific features

We are implementing this system with the basic functionality using Oracle Webserver and Java. We also use Power Objects for quick prototyping. Although some attributes of the database registry, categories and relations can be seen in figures 4, 5 and 6, we do not explicitly define the attributes we have chosen for the extended objects in this paper as we think the elicitation of the model is more important. Our model builds a framework on which many features can be used flexibly. When the user jumps, the previous window can be maintained if desired. Clicking on a number, a date, an image, a document, a video or anything else will jump to the object. Jumping to an object triggers a handling according to the type of the object. This could be the calling of the application to view the document, or another window to see the image. Image regions-of-interest can be registered as objects associated with their anatomical names, which will possibly appear in other images as well. It is important that some objects related with an image may be viewed together with it, possibly superimposed on the image. For pictorial objects the system being built allows the user to delineate an object and to define and relate that object with other objects, including things like the highlighting of the object, legend text pointing a place in the image, the zooming of the image, choosing the number of images to view simultaneously, etc. We will gradually add spatial search capacity to the images.

6. Future work on the model

The model can be implemented on relational, on object or both DBMS and it will become a distributed system. This raises interoperability issues. As shown in Figure 4, there is a registry for instances of databases. Presently the registries reside in one node. Another broad issue in this system is security and privileges. Who can access the system and what privileges will the system give to each user to modify directories dynamically? The system should have a privileges registry to allow the implementation of access policies.

In the extended schema there are types of objects to support the creation of new symbols, sets, relations, etc, and the system provides the facilities to link objects or to relate them through the relation object. These facilities and technical aid will both be instrumental in a medical imagiology environment to explore further the use of those semantics and to identify further needs.

We would like to explore further the hypermedia within component features of our model to allow a uniform access to parts of documents, videos, etc, and relating them to other objects.

7. Conclusions

We have presented KnowShareSys, a system for the sharing of integrated pictorial, document and semantic information within a professional domain, with a most intuitive interface which helps the user to categorize and organize analysis information to be shared, gives generalized access to allowed users, presents a web-like navigation and wide applicability.

We have presented the advantages and new features of the system. We use hyperlink jumping, not only for explicit links but also between any two database objects that are related and to view objects. We include extended semantics that allow the definition of any concept and keeps those concepts in a well organized structure through the use of category directories, registries, links and relationships. We present an intuitive web-based user-friendly interface where every operation can be done in a simple way but which also supports complex queries in a multiparadigmatic access and cooperates with the user to help him choose what to see next. This web-based interface means a universally accessible system and in the future we will turn the system into a distributed one by extending the concepts discussed here. The intelligent integration of all the objects and features makes this system very useful.

References

[ACM HyperMedia 94] Special Issue on HyperMedia in Communications of ACM Feb 1994, V 37, N.2.

[Agra 90] R. Agra al, N. Gehani, J. Srinivasan, Ode-View: The Graphical Interface to Ode," Proc. ACM SIGMOD Conf. , May 1990.

[Ahmed 91] Ahmed R., P. DeSmedt, W. Du, W. Kent, M. Ketabchi, W. Litwin, A. Raffi, M. C. Shan, "Pegasus heterogeneous multidatabase system", IEEE Computer, December 1991.

[Askin 88] Askin R., McCracken, D. L. and Yoder, E. A. KMS: A distributed hypertext for managing knowledge in organizations. Communications of the ACM July 1988, Vol 31, N. 7

[Batini 86] Batini C., M. Lenzerini and S. B. Navathe, " A comparitive analysis of methodologies for database schema integration", ACM Computing Surveys 18(4), December 1986.

[Batini 91] C. Batini et al, "Visual Query Systems", Tech. Rep.04.91, U. di Roma, Marc 1991.

[Cardenas 93] Cardenas A. F. et al, " The Knowledge-Based Object-Oriented PICQUERY+ Language", in IEEE Transactions on Knowledge and Data Engeneering, Vol. 5, n. 4, Aug 1993.

[Carey 96] Carey M., L. Haas, V. Maganty, J. Williams, "PESTO: An Integrated Query/Browser for Object Databases" in Proc. Intl Conf VLDB 1996.

[Carson 96] Carson C., Virginia E. Ogle, "Storage and Retrieval of Feature Data for a Very Large Online Image Collection" in IEEE Transactions on Data Engeneering, vol. 19, n. 4, Dec. 1996.

[Catarci 95] Catarci T., M. F. Costabile, S. Levialdi, C. Batini," Visual Query Systems for Databases: A Survey" , TR SI-95/17, September 1995.

[Catarci 96] Catarci T., M. F. Costabile, S. Levialdi, G. Santucci, "A Graph-Based Framework for Multiparadigmatic Access to Databases", IEEE Transactions on Know. and Data Eng., (8)(3), June 96.

[Cattel 88] Catell R., T. Rogers, "Entity-Relationship Database User Interfaces" in Readings in Database Systems, M. Stonebraker (ed.), Morgan Kauffman, 1988.

[CBR 95] IEEE Computer, September 1995, special issue on “Content Based Image Retrieval Systems”.

[Codd 79] Codd E. F. “Extending the database relational model to capture more meaning”, ACM Transactions on Dtaabase Systems, vol. 4, Dec 1979.

[Chu 94] Wesley Chu, I. T. Ieong, R. K. Taira, “A Semantic Modelling Approach for Image Retrieval by Content”, VLDB Journal, 3, 445-447, 1994.

[Dayal 84] U. Dayal and H. Y. Hwang, “ View definition and generalization for database integration in a multidatabase system”, IEEE Transactions on Software Engeneering 10 (6), 1984.

[Elmasri 85] Elmasri R., J. Larson, “ A graphical query facility for ER databases” in Proc. 4th Conf ER Approach.

[Flickner 95] Myron Flickner, Harpreet S. Sawhney, Jonathan Ashley, Qian Huang, Byron Dom, Monika Gorkani, Jim Hafner, Denis Lee, Dragutin Petkovic, David Steele, Peter Yanker: “Query by Image and Video Content: The QBIC System.” IEEE Computer 28(9): 23-32 (1995)

[Faloutsos 95] C. Faloutsos, Ron Barber, Myron Flickner, Jim Hafner, Wayne Niblack, Dragutin Petkovic, William Equitz: “Efficient and Effective Querying by Image Content.” Journal of Intelligent Information Systems 3(3/4): 231-262 (1994).

[Fogg 84] D. Fogg,, “Lessons from a “Living In a Database" Graphical Query Interface", Proc. ACM SIGMOD Conf., June 1984.

[Goldman 85] Goldman K. J., P. C. Kanellakis, S. A. Goldman and S. B. Zdonik, “ISIS: Interface for a semantic information system,” in Proc. ACM SIGMOD Conf. Management of Data,328-342, May 1985.

[Goble 96] Goble C., C Haul, S Bechhofer. Describing and Classifying Multimedia using the Description Logic GRAIL. SPIE Conf. on Storage and Retrieval of Still Image and Video IV San Jose, 1996.

[Gupta 91] Gupta A. , T. Weyworth, R. Jain, “Semantic Queries with Pictures: The VIMSYS model”, Proceedings of the 17th VLDB Conf., Barcelona, Spain, 1991.

[Gupta 95] Gupta A. , Visual Information Retrieval: A Virage Perspective, 1995.

[Halasz 88] Halasz, F. G., “Reflections on Notecards: Seven Issues for next generation of hypermedia systems” in Communications of the ACM July 1988 Vol. 31, N. 7.

[Hammer 81] Hammer, M. and McLeod D. (1981) : Database description with SDM: a semantic database model ACM TODS 6(3).

[Haw D. 94] Haw D, Goble CA, Rector AL. GUIDANCE: Making it easy for the user to be an expert. In Proceedings of 2nd Intl. workshop on User Interfaces to Databases. 13-15th July 1994, Ambleside K. pp 19-44. To appear in Workshops in Computer Science Springer-Verlag Series, 1994.

[Hardman 94] Hardman L., Bulterman D., Rossum G., “Adding time and context to the Dexter Model” in Special Issue on HyperMedia in Communications of the ACM Feb 1994, V 37, N. 2 .

[Herot 80] C. F. Herot, “Spatial Management of Data “, ACM Trans. on DB Sys. (5) pp 493-514, 1980.

[Hull 87] R. Hull and R. King: Semantic Modelling: Survey Applications and Research Issues. In: ACM Computing Surveys . 1987 pp. 201-260 .

[Haan 92] Haan B. J., P. Kahn, V. Riley, J. Coombs, N. Meyrowitz, “IRIS HyperMedia Services”, Communications of the ACM, January 1992, vol 35, nº1.

[Kunt 89] M. Kuntz and R. Melchert, “Pasta-3 Graphical Query Language: Direct Manipulation, Cooperative Queries, Full Expressive Power”, Proc 15th VLDB Conf., Aug. 1989.

[Laguna 88] The Laguna Beach Participants, Future Directions in DBMS Research - from [Ston 93].

[Mohan 93] Mohan L. and R. L. Kashyap, “A Visual Query Language for Graphical Interaction With Schema Intensive Databases”, IEEE Transactions on Knowledge and data Engeneering, October 1993.

[Motr 88] A Motro, A. D'Atri, and L. Tarantino, The of KIVIEW: An Object-Oriented Browser," Proc. 2ndInt'l. Conf. On Expert Database Sys. , April 1988.

[Nijssen 89] G.M. Nijssen and T.A. Halpin: Conceptual and Relational Database Design: a Fact Oriented Approach. Sydney: Prentice-Hall 1989.

[Noll 91] Noll J., W. Scacchi, “Integrating Diverse Information Repositories: A Distributed Hypertext Approach”, IEEE Computer December 1991.

[Peckham 88] J. Peckham and F. Maryanski: Semantic Data Models. In ACM Computing Surveys Sept. 1988, pp.153-189.

[Rowe 85] Rowe L. , “Fill-in-the-Form Programming," 1991 Proc. 11th VLDB Conference , Aug. 1985.

[Sheth 90] Sheth A. P., J. Larson, “Federated Database systems for managing distributed, heterogeneous and autonomous databases”, ACM Computing Surveys 22(3), September 1990.

[Ston 82] M. Stonebraker and J. Kalash,”TIMBER- A sofisticated relational browser”, Procs of the 8th VLDB Conf., 1982.

[Ston 93] M. Stonebraker Ed., Readings in Database Systems 2nd edition,1993 - Morgan Kaufmann.

[Vazirgiannis 95] M. Vazirgiannis, A. Kalousis, M. Hatzopoulos, “ Knowel: A Hypermedia Knowledge Editor”, Workshop of Database and Expert Systems Applications, 6th International Conference, DEXA ’95, London, United Kingdom, September 4-8, 1995.

[Yao 84] S. B. Yao et al, “FORMANAGER: An Office Forms Management System”, ACM Transactions on Office Info. Systems, 2(3), July 1984.

[Zloo 77] M. Zloof, “ Query By Example”, IBM Sys. J. 16, Dec. 1977.

[Zhou 94] Zhou G., R. Hull, R. King, JC. Franchitti, “Using Object Matching and Materialization to Integrate Heterogeneous DataBases”, Proc. Of the 3 Intl. Conf. On Cooperative Info. Sys., May, 1995.