DATABASE GROUP AT ISS

Institute of Systems Science
National University of Singapore
Heng Mui Keng Terrace, Kent Ridge, Singapore 0511

The database group is headed by Dr. A Desai Narasimhalu. It comprises nine staff members, together with a collaborative member and three undergraduate students from the Department of Information Systems and Computer Science. The group's main research areas are listed below.

A. Past Work

1. Data Model For Object Oriented Databases

Member Dr. A Desai Narasimhalu

Objective To develop a data model that can support aggregations and classifications for multimedia data.

Description Multimedia objects need both classification and aggregation to support animation and other applications. Also, such data if stored on optical disk will consist of both current and historical data. Hence, the data model should support temporal information. Temporal information can be either of the periodic type such as time series or non-periodic. Furthermore, the data model also incorporates the concept of reference object and target object in a user-defined relationship. Our present projects make frequent use of this concept.

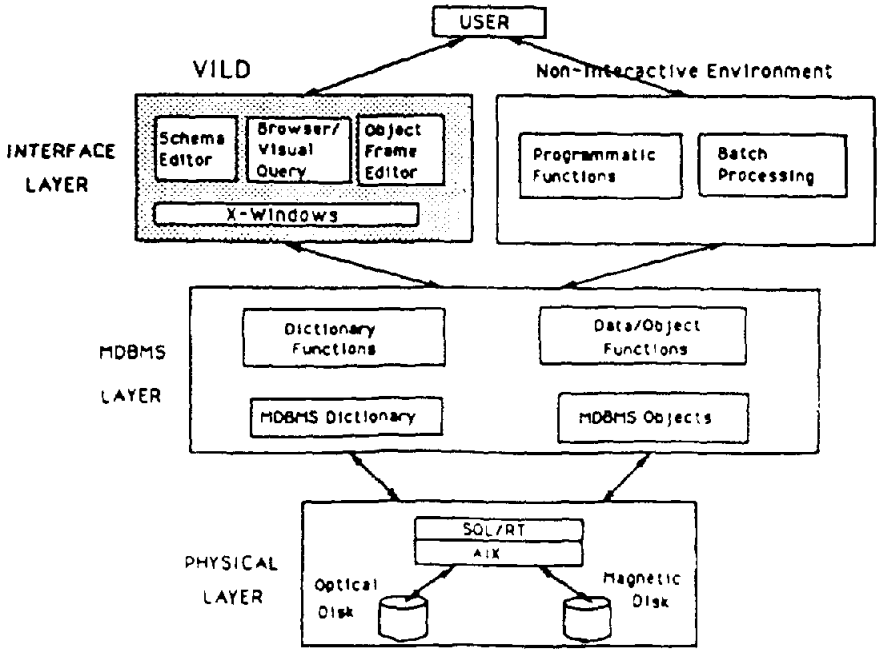
Status The data model, together with a set of algorithms for the design of databases using this data model, have been completed [Na88].

2. MDBMS - An Object Oriented Multimedia Database Management System

Members MunKew Leong, Rosanne Price, BengChong Ooi, Dr. A Desai Narasimhalu
Collaborator Singapore Telecoms

Objective To design and develop a Multimedia Database Management System with an interface to a write once optical disk.

Description MDBMS extends the range of data types handled by current database systems. New data types handled by this system are Image, Matrix (made up of image data so often used that are hence assigned a code), Graphics and Multilingual text. In addition, it has two other data types. A 'file' data type to store data that does not have to be interpreted by the DBMS and an extensible data type that can be used to



MDBMS Architecture

define any other data type required by the user. The MDBMS supports the definition of primitive objects, sub/superclasses, composite objects and other relationships between objects. In addition, it provides temporal data management. It is implemented as a layer on top of an existing relational DBMS. The large amount of image data are stored on optical disks.

Status An optical disk drive has been integrated to the MS-DOS operating system which provides transparent access modes similar to magnetic disk drives [Oo87, Oo88]. This work has been ported to the IBM RT/PC for the AIX environment. A prototype of the MDBMS has been implemented on this hardware configuration. It has been used in ISS to support the development of a prototype of a hotel information system. This system presents information about hotels in Singapore in the form of English text, Chinese descriptions, road maps and pictures. Some vendors have expressed interest in developing systems based on this prototype. Also, a visual interface (VILD) is being built for MDBMS.

Benefits and Applications of Results No other single existing DBMS can currently handle the complete range of temporal, object-oriented, multimedia, and multilingual data in an integrated fashion or provide a visual interface. The concepts and techniques illustrated by MDBMS represent a significant contribution to the continuing effort to meet the database requirements of the next generation of computer users.

B. Present Work

1. Database Support For Hypermedia Systems

Members Dr. BengChin Ooi, LimSoon Wong, HweeHwa Pang, Shawn Sam, MunKew Leong, Dr. A Desai Narasimhalu

Objective To develop a back-end database server for hypermedia systems.

Description Hypermedia is an excellent tool that allows flexible browsing through a large collection of information. Present hypermedia systems are generally small to medium in size. When larger systems are built, the information has to be organized for efficient storage and retrieval. Database support will provide this efficiency.

As these systems become large, they will become multi-user systems that may also be distributed over many sites. Such systems need to provide for concurrency control and consistency among distributed information. The support for these features will be provided by a database server.

Status The features of the database support have been identified. The project group has decided to adopt object oriented database (OODB) systems as our basis in the construction of intelligent databases. The high level architecture of the OODB management system has been designed and we are looking at the possibility of enhancing an existing OODB to include the additional functionality required. Please refer to the next page for the architecture.

Benefits and Applications of Results The database support will allow the development and maintenance of large hypermedia systems that are used for applications such as electronic publications and national archives.

2. VILD - Visual Language Interface To Multimedia Databases

Members JyhJang Lim, MunKew Leong, Shawn Sam, Dr. A Desai Narasimhalu *Collaborator* Dr. HongJun Lu *Other participants* SokKim Goh, ChinBoon Yeap and SooKian Wong.

Objective To build visual interfaces to object oriented databases using graphical icons.

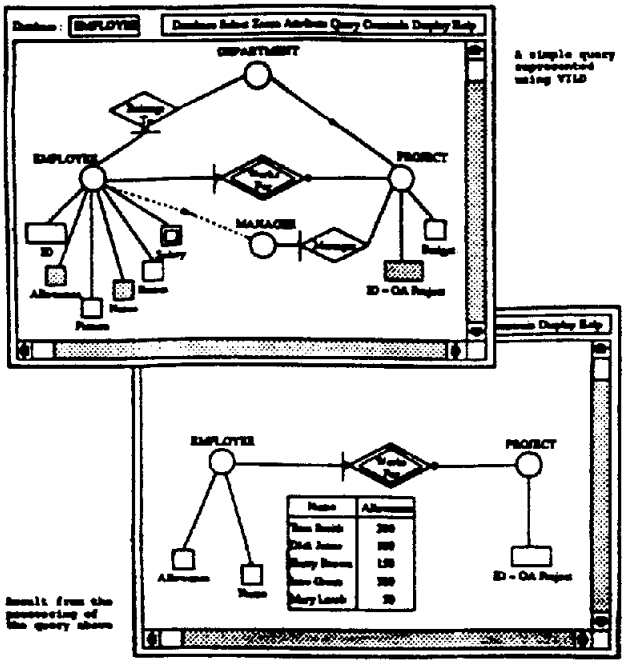
Description This project develops a visual interface to an object-oriented database system. Such an interface will allow a user to use graphical icons for the definition of a database schema for an application. The schema can be used to create the required database. The system allows a person to call up an object through a datatype-sensitive editor to either input or modify attribute values of the object.

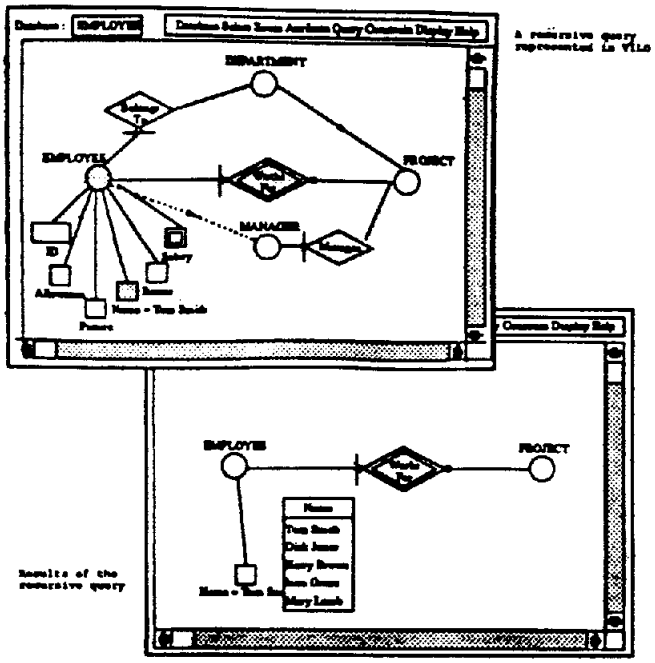
The graphical icons used for the definition of a schema can also be used for browsing and querying the database. Visual interfaces will minimize the amount of training needed to use a database system. It eliminates much of the programming normally required in using conventional data definition, data manipulation and query languages.

Status The definition, edit, and browse capabilities of VILD have been completed by BoonSiong Choo, ChunHong Kok, and JyhJang

OODB Architecture - Definition
Define Data Model
Create Dictionary
Data Model
Dictionary
Database Structures
Access Methods Information
Select Implementation
Create Tables
Index Tables
Access Tables
Access Mechanisms
Create Files
Database Files
Storage Structures
Persistent Object Language Source Code
OODB Definition
optional
Semantic Constraints
optional
Optimizing Information
Semantic Precompiler
Optimized Database Program
Database Program
Transformation
Program with rules and persistent vars
Persistent Var Management
Persistent Variable File
Persistent Variable Data

OODB Architecture - Runtime
File System
Object Physical Memory
Machine Virtual Memory
Object Virtual Memory
Partitioned Cache
UNIX File Management
Page Management
Cache Management
Buffer
ADT Management
Object Management
View Management
Method Management
Runtime Optimizer
Query Language
Database Programs
Index Management
Index
Statistical Dictionary
Data Dictionary





Lim [Le89]. The graphical query language is being implemented by SokKim Goh, ChinBoon Yeap, and SooKian Wong. A working prototype is expected by end of July. Additionally, a visual programming environment is being developed for those applications that need interface to databases. This will be built around VILD.

Benefits and Application of Results Visual interfaces will allow a larger number of people to use database systems. It also improves the usability of database systems thereby resulting in improved organizational productivity.

3. Inferential Object-Oriented Programming For Databases

Members LimSoon Wong, Dr. A Desai Narasimhalu

Objective The objective is the integration of the logic programming and object oriented programming paradigms. The integration should be sensible and the two paradigms should be mixed orthogonally.

Description Object oriented systems, though close to modern data models such as [Na89], lacks declarative description of complex object relationships. The motivation of this part of the project is to make such descriptions possible. We have taken a novel approach by using a concept we call "rule obedience", which we introduced in [Wo89a]. Its semantics is given in [Wo89b]. An implementation suitable for separate compilation has been suggested in [Wo89b].

Essentially, range restricted functorless datalogs are employed to express object relationships or rules. In a rule such as the classic TheAncestorRule, there are two roles: Ancestor and Descendant. Object classes are given the ability to declare their obedience or disobedience to a rule in a certain role. Once a class obeys TheAncestorRule as Ancestor, then every object of this class (barring non-monotonic inheritance) can be used to instantiate variables corresponding to Ancestor in TheAncestorRule. Furthermore, a rule once declared becomes universally understood, i.e. every object in the system knows about it. Thus we can direct queries involving the rules to any object. To make query answering natural and sensible, methods known as "foci" are used [Wo89a, Wo89b]. A query such as "who is the ancestor of whom?" is formalized as a focus. The previous focus is in third person. First person foci such as "Are you David's descendant?" are provided too. Once a rule enters the system, its associated foci immediately become methods of all objects in the system.

The semantics is clean and orthogonal. Any object can play any role in any rule. There is an additional semantic benefit. Because only objects playing a particular role in a particular rule can be used for instantiation in that part of the rule (given that the number of objects in a database is guaranteed finite), we think we can handle negation in all possible modes.

Status The main idea in the suggested implementation in [Wo89b] is compile time

transformation from nondeterministic rules into relative set abstractions. It is currently being improved so that goal directed top-down but nonredundant evaluation is achieved based on a simple memoization technique [Wo89c].

Benefits and Application of Results Our research can be applied to any object oriented programming system, whether it is used for databases or not. From now on, in order to have a complex rule such as TheSameGenerationRule (another classic) in an object oriented system, we need not write the (n+1)! methods necessary to reply to every queries concerning it; all we need to do is just write down a logic expression and all the methods should be generated instantly, correctly, and efficiently.

4. Semantic Query Optimization In OODB

Members HweeHwa Pang, Dr. A Desai Narasimhalu *Collaborator* Dr. HongJun Lu

Objective To build a semantic query optimizer for object oriented database systems.

Description In order to maintain the overall efficiency of the optimizer, it is necessary to perform only those transformations that may lead to a query that is less costly to execute. However, as the number of semantic constraints and object classes grows, it becomes increasingly costly to determine which constraints are relevant to a query and to decide which transformations are potentially profitable.

A feature of this semantic query optimizer that distinguishes it from most other related work is a scheme to classify constraints. The idea underlying the classification scheme is to identify the host objects of each semantic constraint and store the constraint with these host objects. During optimization only constraints stored with objects involved in the query need to be retrieved and considered. The resulting reduction in overhead costs thus derived can be substantial especially when there are many object classes and semantic constraints.

We have included in our optimizer automatically derived rules that reflect the current state of the database, which was introduced by Siegel [Si88]. Examples of such rules might be "All employees live in Singapore", or "No employee earns more than $10,000". Because these states may not always be true of the database, they will not be specified as semantic constraints. However, knowledge of these states might be very useful for optimization purposes. A mechanism which monitors the performance of these rules are incorporated into the optimizer to help the database administrator maintain the rules.

In this optimizer, instead of embedding the transformation rules in program codes, we allow the database designer to input them as data to the optimizer. As the specification of the conditions under which a transformation rule is applicable can be quite complicated, we express the rules procedurally (i.e. using C functions). For instance, the conditions of a rule might be "we will only introduce an index if the range ratio is small and the index is clustered".

The optimizer also makes use of AI techniques to automatically fine-tune the system. Cost statistics are maintained to enable the optimizer to determine the profitability of each potential transformation. The cost statistics must be adjusted to reflect changes in access patterns and past cost savings. A set of learning algorithms are provided to update these statistics. A good part of system tuning is thus done automatically instead of depending on a database administrator.

Status The design of the optimizer has been completed and published as a working paper [Pa89]. We are currently implementing a prototype of the optimizer to test its performance.

Benefits and Application of Results The various features we provide improve the effectiveness of the optimizer and make the optimizer a useful component of an OODB.

Further Information

Enquiries may be addressed to either of the following through BITNET.

HweeHwa Pang - ISSPHH@NUSVM
Dr. A Desai Narasimhalu - ISSAD@NUSVM

References

[Le89] MunKew Leong, Shawn Sam, Desai Narasimhalu, "*Towards a Visual Language for an Object-Oriented Multi-Media Database System*", Proceedings of the IFIP TC 2/WG 2.6 Working Conference on Visual Database Systems, Tokyo, Japan, April 1989.

[Na88] A. Desai Narasimhalu, "*A Data Model for Object Oriented Databases with Temporal Attributes and Relationships*", Technical Report 88-04-0, Institute of Systems Science, National University of Singapore, 1988.

[Oo87] B.C. Ooi, A.D. Narasimhalu, K.Y. Wang, I.F. Chang "*The Design of a Multimedia File Server Using Optical Disks for Office Applications*", IEEE Computer Society Symposium on Office Automation 87, Gaithersburg MD, USA.

[Oo88] B.C. Ooi, A.D. Narasimhalu, I.F. Chang, "*Integration of Write Once Optical Disk with Multimedia DBMS*", RIAO 88 Conference, Massachusetts Institute of Technology, Cambridge USA.

[Pa89] HweeHwa Pang, "*Semantic Query Optimization in OODB*", Working Paper, Institute of Systems Science, National University of Singapore, 1989.

[Si88] M.D. Siegel, "*Automatic Rule Derivation for Semantic Query Optimization*", Proceedings of the Second International Conference on Expert Database Systems, Virginia, April 1988.

[Wo89a] LimSoon Wong, "*Inference Rules in Object Oriented Programming Systems*", Technical Report 89-04-0, Institute of Systems Science, National University of Singapore, 1989. Submitted to the 1st International Conference on Deductive Object Oriented Database Systems, Kyoto, Japan, 1989.

[Wo89b] LimSoon Wong, "*Inferential Object Oriented Programming for Databases*", Working Paper, Institute of Systems Science, National University of Singapore, 1989.

[Wo89c] LimSoon Wong, "*Datalog Evaluation Without Magic*", Working Paper, Institute of Systems Science, National University of Singapore, 1989.