

**European Community IV Framework
Esprit Project 20671**

TOOBIS

**Temporal Object-Oriented Databases
within Information Systems**

**Final Project Report
Deliverable T12R.3**

June 1999

Table of Contents

<i>Table of Contents</i>	<i>i</i>
1. Introduction	1
2. Project overview	2
3. Objectives	4
3.1. General Objectives	4
3.2. Partner Roles and Specific Objectives	5
3.2.1 01 Pliroforiki - Prime Contractor	5
3.2.2 Matra Systemes et Informations	5
3.2.3 Ardent Software	6
3.2.4 Delta S.A.	6
3.2.5 GlaxoWellcome	7
3.2.6 University of Sorbonne – Paris 1	7
3.2.7 University of Athens	7
4. Results	8
4.1. Technical achievements	8
4.2. The applications	10
4.2.1 Delta application	10
4.2.2 GlaxoWellcome application	10
4.3. Dissemination	11
4.3.1 Seminars/Conferences	11
4.3.2 White Papers	11
4.3.3 Publications	12
4.3.4 TOOBIS CD	13
4.3.5 HTML pages	13
4.4. Deliverables	13
5. Detailed description of the TOOBIS platform	15
5.1. Temporal Object Oriented Methodology - TOOM	15
5.1.1 Objectives of the TOOBIS methodology	15
5.1.2 Overview of the OO method (OOM)	15
5.1.2.1 Structural properties	15
5.1.2.2 Behavioural properties	16
5.1.3 Overview of the temporal extension (TOOM)	17
5.1.3.1 Calendar definition	17
5.1.3.2 Extension of the basic domains to temporal domains	18
5.1.3.3 Extension of object classes to time dimensions	18
5.1.3.4 Extension of constraints applied on object classes to time dimensions	21
5.1.3.5 Temporal extension of events	21
5.1.4 Conclusion	22
5.2. Temporal Object-Oriented DBMS (TOODBMS)	23
5.2.1 Temporal Object Data Model - TODM	23
5.2.1.1 Time Model and Manipulation	23
5.2.1.2 Temporal Data within Objects	24
5.2.2 Temporal Object Definition Language - TODL	27
5.2.2.1 TODL module architecture	29
5.2.3 Temporal Object Query Language - TOQL	30
5.2.3.1 Data types for time representation	30
5.2.3.2 Referencing Temporal Data in Queries	31
5.2.3.3 Selecting Information from Temporal Data	31

5.2.3.4	Partitioning and Aggregating on the Temporal Dimensions	33
5.2.3.5	Advanced Features of TOQL	33
5.2.3.6	TOQL module architecture	33
6.	<i>Overview of the Pilot Applications</i>	35
6.1.	The Delta S.A. DSS Application	35
6.2.	The Glaxo Wellcome Clinical Trial Management Application	38
7.	<i>Future Exploitation</i>	42
7.1.	01 Pliroforiki	42
7.1.1	TOOM Methodology for Consultancy, Training and Development activities	42
7.1.2	Pilot application development for customised applications	42
7.1.3	Object Technology	43
7.1.4	The TOOBIS platform	44
7.2.	Matra Systemes et Informations	44
7.2.1	Interest of TOOBIS in MS&I activities	44
7.2.2	Market analysis	45
7.2.3	MS&I offer	46
7.2.4	Expected MS&I business	47
7.3.	Ardent Software Inc.	47
7.4.	University of Sorbonne	48
7.5.	University of Athens	48
7.6.	Delta Dairy S.A.	49
7.7.	GlaxoWellcome	49
8.	<i>References</i>	50
	<i>Annex: Market Analysis</i>	52
	<i>Project Roster</i>	55

1. Introduction

In the last two decades, researchers have addressed the issue of *temporal data*, i.e. data varying over time. In the real world, entities evolve over time; thus their respective representation in information systems must change. Most computerised information systems (and DBMSs, in particular) nowadays retain only the *current state* of real-world entities, providing no support for keeping past values or storing predicted future values. Some information systems allow for retaining a log of past states for each entity, but these facilities mostly serve auditing and archiving purposes, providing only minimal support for information retrieval and analysis. An additional time-related requirement for information systems is to store the time that each piece of information was stored in the information repository and the time it was (logically) removed. Technically, the time that each piece of information is true in the real world is termed *valid time*, while the time from the insertion of a piece of information in a DBMS to its (logical) deletion is termed *transaction time*. Depending on the support they provide for the two time dimensions, DBMSs are classified into four categories:

- *Snapshot DBMSs*, which includes “traditional” DBMSs that provide no support for either time dimension.
- *Valid time (or historical) DBMSs*. This category includes DBMSs that support only the valid time dimension.
- *Transaction time (or rollback) DBMSs*. This category includes DBMSs that support only the transaction time dimension.
- *Bitemporal DBMSs*, which includes DBMSs that support both time dimensions.

Temporal information systems are inherently more complex than *snapshot* information systems (i.e. information systems storing only the current state of the modelled universe), since an extra dimension (the time dimension) is added. In this respect, along with the tools to support the storage and manipulation of time-varying information, a methodology is also needed to enable analysts and designers to capture, represent and analyse the semantics and requirements of temporal information systems.

The TOOBIS project has addressed both issues, by providing both a methodology and a complete DBMS, building thus a platform to support the complete life cycle of temporal information systems. The platform has been validated by modelling and implementing two end-user applications, a DSS system for optimising the production and loading of dairy products and a medical application for patient monitoring and therapeutical protocol assessment.

2. Project overview

The TOOBIS project developed a platform to support the full life cycle of temporal information systems. This platform includes an innovative conceptual modelling methodology, which allows for capturing the semantics, analysing the requirements and formulating the design of temporal information systems. Additionally, the core functionality of an OO-DBMS has been extended, resulting in a Temporal Object-Oriented Database Management System (TOODBMS), which provides the necessary basis for implementing and maintaining temporal applications.

The TOODBMS is positioned towards the market of Temporal Object-Oriented DBMSs, which is a segment of the OODBMS market. Object-oriented DBMSs will be increasing their market share in the forthcoming years, as the degree of data complexity that must be managed by the applications is increasing too.

Many of these applications are dealing with time and dates, while there is an increasing need for maintenance and flexible management of evolving information. In this respect, the temporal extensions implemented in this project exploit the power and expressiveness of the OODBMS technology to support existing and emerging data management requirements. The applications currently running or moving towards on OODBMSs cover the following areas:

- **Telecommunications:** network management, administration, simulation, planning, etc.
- **Aerospace industry:** technical data and technical documentation management.
- **Defence:** commandment systems, mission preparation, simulation, war games, etc.
- **Bank and insurance:** back and front office system, trade house system, customer information systems.
- **Services:** electricity distribution, transportation network management, cartography, and utilities.
- **Commercial:** (DSS applications on finance, banking, insurance).
- **Health:** (simulation, medical operations, disease management, pharmacology).

The benefits of TOODBMS technology are:

- The ability to analyse and design, the temporal aspects of static and dynamic behaviour of an application domain.
- The reduction of temporal applications design, development, maintenance and evolution costs.
- The improvement of the performance of the resulting applications.
- The provision of a flexible framework for storing temporal data and a powerful query language to retrieve them.

Using the results, the expected benefits can be the formalisation of temporal object oriented data, efficient historical data management, data synthesis and analysis through temporal information retrieval.

DELTA S.A., the largest fresh-products producer in Greece, is using the TOODBMS as the core of a DSS, which predicts the next day milk consumption, based on historical information. Thus an optimal production, loading and distribution plan can be formulated, in order to minimise the amount of expired milk product returns.

GLAXO Wellcome, one of the biggest pharmaceutical companies is using the TOODBMS for managing the data of a medical application. In the scope of this application, patients suffering from asthma are monitored and the evolution of their symptoms is recorded, together with the therapeutical protocols employed. This information is then used to assess the efficiency of the various therapeutical protocols, which usually involve new medication.

3. Objectives

3.1. General Objectives

TOOBIS aim was the development of a competitive platform to support the complete life cycle of temporal applications. Its objectives were the following:

1. Provide a temporal analysis and design methodology, by extending and adapting existing methodologies.
2. Extend the core functionality offered by the O₂ OODBMS, by supporting both valid and transaction time. These extensions should be compatible with the ODMG standards and take into account the forthcoming SQL-3 standard definitions.
3. Comply with evolving ODBMS standards, participate and influence standards definition groups.
4. Enrich the functionality of the O₂ OODBMS, providing the data model constructs for creating and manipulating temporal objects.
5. Enhance the standard Object Definition Language (ODL).
6. Extend the standard high-level Object Query Language (OQL).
7. Validate the approach and TOOBIS products (TOODBMS and Methodology) by developing two demonstrator applications.
8. Gain a competitive advantage with respect to both core ODBMS functionality, and industrial size applications in the areas of MIS-EIS and health applications involving temporal and GIS functionality.
9. Generalise applications of TOOBIS and investigate marketing potential in other industrial sectors.

The TOOBIS project has developed an innovative conceptual modelling methodology for temporal applications. In that respect TOOBIS contributes to the ease of capturing, representing and analysing the semantics and the behaviour of temporal information systems. TOOBIS also extends the core functionality offered by the O₂ OODBMS, by supporting both valid and transaction time, being compatible with the ODMG standard, and taking into account the forthcoming SQL-3 standard definitions (SQL/T).

The TOODBMS provides appropriate Data Model constructs for creating and manipulating temporal objects, enhances the O₂ OODBMS with a Temporal Object Definition Language (TODL), which is an extension of ODMG ODL, and extends the standard high-level Object Query Language (OQL). The approach and TOOBIS products (TOODBMS and Methodology) have been validated via the development of two demonstrator applications.

Furthermore, the TOOBIS consortium, mainly through Ardent Software, participates in standards definition groups and influences the directions of standards developments. One of the goals of the consortium is to include elements of the TOODBMS in a future version of the ODMG standard that will address temporal support.

3.2. Partner Roles and Specific Objectives

<i>Name of participating organisation</i>	<i>Admin. Role C/P/A</i>	<i>Funct. Role S/U</i>	<i>Effort in Person Years</i>	<i>Country</i>
01 PLIROFORIKI S.A.	C	S	6.25	Greece
DELTA S.A.	P	U	0.71	Greece
GLAXO WELLCOME	P	U	0.48	France
Matra Systemes & Information	P	S	3.75	France
University Paris 1–Pantheon–Sorbonne	P	S	1.58	France
Ardent Software	P	S	1.04	France
University of Athens	A	S	3.33	Greece

to 01 PLIROFORIKI

C = Co-ordinating Partner, P = Partner, A = Associate Partner

3.2.1 01 Pliroforiki - Prime Contractor

A Greek SME specialising in IT Consultancy and Development. 01-P participated in specifications and development of the definition and query languages of TOOBIS in collaboration with University of Athens (UoA). It also participated in the Development of the Pilot Application of TOOBIS for DELTA

Address	Contact Person
01 PLIROFORIKI S.A 438 Aharnon Street 111 43 Athens Greece Phone: +30 1 21 82 500 Fax: +30 1 25 33 161 Web: http://www.01p.gr/	Dr. John Kozatsas E-mail: jkoz@01p.gr

3.2.2 Matra Systemes et Informations

Matra developed the Temporal Object Data Model of TOOBIS. In addition MS&I developed, in collaboration with GlaxoWellcome, a pilot application for the evaluation of the TOOBIS platform.

Address	Contact Person
MATRA Systemes & Information S.A. Parc D'affaires des Portes BP 613 27106 Val-de-Reuil Cedex – FRANCE Tel.: +33.2.32.63.40.45 & 23 Fax: +33.2.32.63.42.00 Web: http://www.matra-msi.com	Dr. Yann Pollet Email: pollet@vdr14.mcs-vdr.fr

3.2.3 Ardent Software

Westboro, Mass., Ardent Software is a leading global data management software company. Ardent develops and markets embedded databases and software tools for enterprise-scale applications and data warehouses. Ardent is all about data management - storing and retrieving complex data, as well as accessing and moving among diverse environments and reorganising it along the way. Its family of database systems and warehouse development tools addresses the spectrum of data management issues, including all aspects of the mapping, modelling, manipulation and movement of data.

The two principal product lines are embedded databases and application development tools, and data warehouse generation, administration and management tools. Ardent's products are sold world-wide through a direct sales force and are also sold by many partners who use the technology to power the application solutions they deliver to their end-user customers.

Ardent is headquartered in Westboro, Massachusetts and has 12 field offices across the United States, as well as offices in Europe, Africa and Asia. We offer a full range of support, consulting, and education services for all of our products, delivered by an international team of experts dedicated to customer service and support. With world-wide sales and service operations, and 1,000 resellers in more than 50 countries, Ardent is one of the top software companies in the world.

Exchange / Symbol: NASDAQ:ARDT

Web Server for information: <http://www.ArdentSoftware.com/>

Ardent Software (former O₂ Technology) is a member of the ODMG committee. Ardent Software led the exploitation task and was responsible for the optimisation of the temporal extension to the OODBMS and the quality monitoring of the TOODBMS development.

Address	Contact person
Ardent Software 7 rue du Parc de Clagny 78035 Versailles Cedex - France Tel.: +33.1.30.84.77.77 Fax: +33.1.30.84.77.90 Web: http://www.ArdentSoftware.fr/	Mrs Catherine Lanquette E-mail: Catherine.Lanquette@ArdentSoftware.fr

3.2.4 Delta S.A.

DELTA is a User organisation. DELTA is interested in the development of a Decision Support Application that will enable the minimisation of milk-products returns. The application has significant time handling requirements, since the prediction of dairy products consumption –whose accuracy is critical for the application's goals– is heavily based on the existence of historical data and the facilities for retrieval, manipulation and analysis.

Address	Contact Person
DELTA DAIRY S.A. 3 Kerkyras Street 177 78 Tavros Athens - GREECE Tel.: +30.1.3494.000 Fax: +30.1.3494.040 Web: http://www.delta.gr	Mr. Chrysanthos Pavlou E-mail: cpavlou@delta.gr

3.2.5 GlaxoWellcome

GLAXO is a User organisation. GLAXO participated in the development of an application that enables the monitoring of the clinical trials of pharmaceutical products.

Address	Contact Person
Laboratoires GLAXOWELLCOME S.A. 23 rue Lavoisier 27000 Evreux FRANCE Tel.: +33.2.32.23.55.00 Fax: +33.2.32.23.55.50	Mr. Hakim Zeghib Email: hz29043@ggr.co.uk

3.2.6 University of Sorbonne – Paris 1

University of Sorbonne developed the Temporal Object Oriented Methodology (TOOM) which was applied and evaluated in its use to the End User Applications.

Address	Contact Person
Universite Paris I - Pantheon - Sorbonne UFR 06 90 Rue de Tolbiac 75013 Paris FRANCE Tel.: +33.1.40 77 46 34 Fax: +33.1.45.86.76.66	Dr. Carine Souveyet Email: souveyet@univ-paris1.fr

3.2.7 University of Athens

The UoA was engaged in the task of specifying and developing the definition and query languages for the TOODBMS.

Address	Contact Person
University of Athens - Department of Informatics Panepistimiopolis - TYPA Buildings 157 71 Athens Tel.: +30.1.7257.560, 7275.220 Fax: +30.1.7275.214 Web: http://www.di.uoa.gr/	Dr. Costas Vassilakis E-mail: costas@di.uoa.gr toobis@di.uoa.gr

4. Results

4.1. Technical achievements

The TOOBIS project has developed a platform to support the full life cycle of temporal information systems. This platform includes the following components:

- 1) The *Temporal Object-Oriented Methodology* (TOOM). TOOM is used in the analysis and design phases of temporal information systems' lifecycle. TOOM, along with being capable of capturing and modelling the “traditional” needs of an information system using UML-compliant processes and notations, provides the necessary constructs which enable analysts and designers to capture and formally model the information systems' temporal needs. TOOM provides constructs to model both the structural and behavioural aspects of information systems. In all the steps of these phases, analysts and designers are aided by a comprehensive guideline suite, which enables them to determine which TOOM constructs are most suitable for representing the entities of the modelled universe.
- 2) The *Temporal Object-Oriented DBMS* (TOODBMS). The TOODBMS may be used to define, store and query temporal data, providing thus the necessary infrastructure for the implementation and operation of temporal information systems. The TOODBMS comprises of three distinct, inter-operating modules that deliver the temporal functionality to applications and users, as illustrated in Figure 1. These modules are:

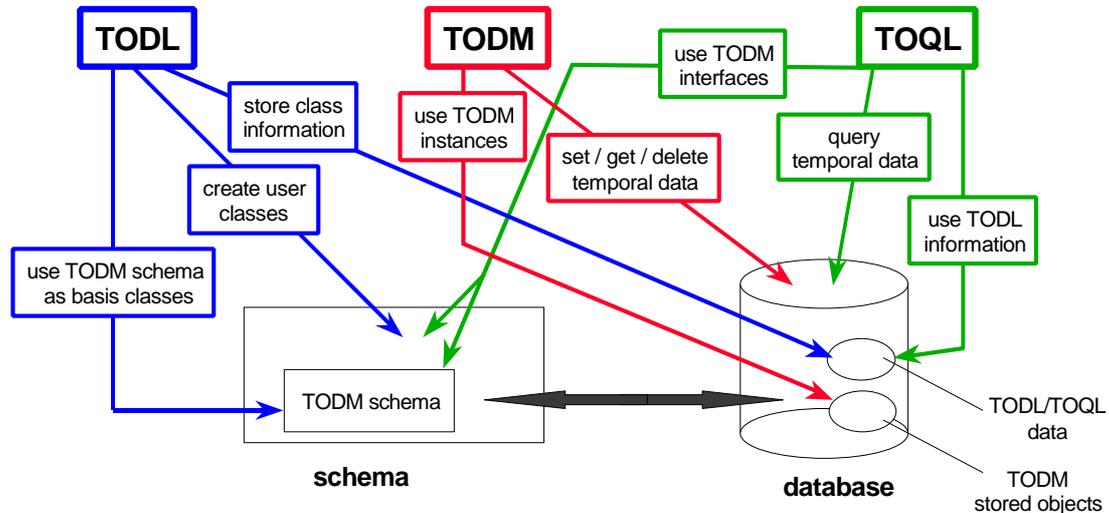


Figure 1 – Architecture of the TOODBMS

- i) The *Temporal Object Data Model* (TODM). This module provides the necessary facilities for the creation, storage and application of operations on temporal and non-temporal objects. Effectively, TODM extends the ODMG Object Data Model (ODM), by including a temporal counterpart for each data type supported by ODM, and by providing the necessary facilities to add temporal characteristics to user-defined data types. The operations provided by TODM include the addition and deletion of temporal information to (from) temporal objects, extraction of temporal information pertaining to

specific time point or time periods, combining information from various temporal objects etc.

- ii) The *Temporal Object Definition Language* (TODL). This module provides a high-level facility for defining objects with temporal characteristics. TODL extends the ODMG *Object Definition Language* (ODL), allowing for the definition of objects having temporal characteristics and methods operating on such objects. Additionally, the TODL module of the TOODBMS maintains a dictionary (*meta-data*), through which users can query the available types and determine their properties.
- iii) The *Temporal Object Query Language* (TOQL), a high-level level query language for retrieving information from temporal databases. TOQL is an extension of the ODMG *Object Query Language* (OQL), introducing methods for accessing whole object histories or specific portions of them, computing aggregate values over the time axis (e.g. departments' expenses per year), combining information pertaining to the same time periods (through *temporal joins*) etc. Besides providing augmented functionality to applications written having the temporal dimension in mind, TOQL delivers the maximum degree of compatibility for legacy applications, i.e. applications that were designed to operate on a database containing only the current information.

All modules of the TOODBMS have been implemented on top of the O₂ OODBMS. The specifications of the TOODBMS have taken into account existing and emerging standards in the areas of Object-Oriented DBMSs and Temporal DBMSs. In particular, the TODL and TOQL modules are fully syntactically and semantically conformant to the ODMG standards of ODL v. 2.0 and OQL v 1.2, respectively. Furthermore, the semantics SQL/T (the temporal part of the forthcoming SQL-3 standard) have been taken into consideration. The TOOBIS consortium, mainly through Ardent Software, participates in standards definition groups and influences the directions of standards developments.

The TOOM component of the platform is linked to the TOODBMS through a set of *mapping guidelines*. Once the design step for a temporal information system is complete, designers and implementers may use this set of guidelines, in order to choose the most appropriate data model constructs to use, in order to represent the entities of the modelled universe within the TOODBMS.

Both the methodology and TOODMBS have been validated in the context of the TOOBIS project via the development of two pilot applications, a DSS system for optimising the production and loading of dairy products and a medical application for patient monitoring and therapeutical protocol assessment. These applications are briefly described in the following paragraphs, while the overall system architecture is illustrated in Figure 2.

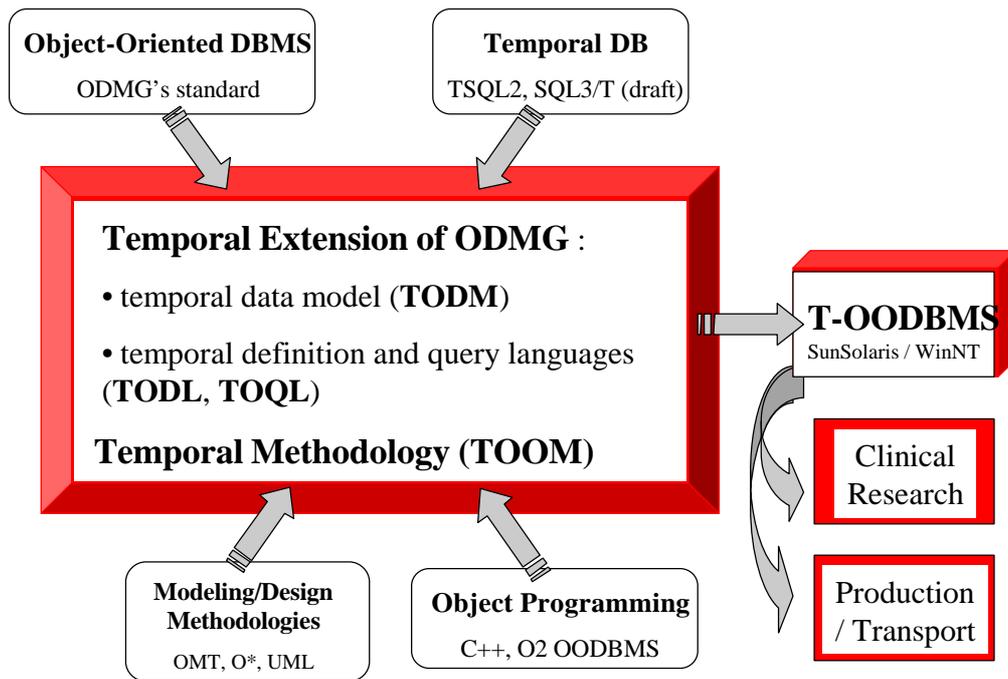


Figure 2 – Overall Architecture of the TOOBIS platform.

4.2. The applications

4.2.1 Delta application

The pilot application for DELTA S.A. is a scheduling system that monitors the daily loading and returns of products having limited shelf-life (2-3 days), in order to offer advice on production planning and fleet loading. The pilot offers the following functionality:

- Efficient management of daily trucks loading and daily returns of goods historical information,
- Prediction of next day consumption for returns minimisation,
- Historical information of factors affecting the returns of goods, on a daily basis (weather conditions, production breakdowns etc.), which have been proved to be critical factors,
- Generation of complex quality reports based on temporal information.

The DSS application does not only address data with temporal dimensions, but handles snapshot data as well, constituting thus a full-scale DSS application, which caters for all the company's needs related to optimal product manufacturing and fleet loading.

4.2.2 GlaxoWellcome application

The pilot application for GlaxoWellcome S.A. allows the management of data resulting from Clinical Trials: acquisition, clarification process and preparation of biostatistic analyses. The challenge is to take into account all the temporal aspects in order to monitor the history and make a complete assessment of the patient's medical record and the therapeutical protocols' efficiency.

The goal of the application is to enable the monitoring of the evolution of symptoms of a patient and evaluate the therapeutical protocols' efficiency. Another function of the application is to provide a generic level in the management of the different clinical trials, in order to minimise and to automate new developments required each new trial.

4.3. Dissemination

4.3.1 Seminars/Conferences

1. 01P, the Department of Informatics of the University of Athens and the University of Sorbonne organised a seminar entitled: "Object oriented Methodologies Software Development and their Time Extensions". The seminar was hosted at the "01 PLIROFORIKI-EKPEDEFTIKI" Vocational Training Centre on the 26th of May, 1997. More than 35 participants from both industry and public sector attended.
2. Banktech Exhibition in Athens, Greece on the 20th – 21st of May 1997.
3. Very Large Data Bases '97 Conference in Athens, Greece on the 26th-29th of August 1997. 01P had an exhibition of the O2 ODMG Compliant Database System and supplied dissemination material on the TOOBIS TOODBMS and methodology.
4. "Object Oriented Methodologies Software Development and their Time extensions" Seminar took place on the 26th of May 1998, organised by 01P in collaboration with the UoA and Sorbonne. The seminar was hosted by the University of Patras. More than 36 participants from Research (Academics & Postgraduates) and Industry.
5. The consortium has actively participated in the OOIS'98 conference. In a session organised by the University of Sorbonne and lasting 1:30', the TOOBIS project was presented by 01P, Ardent and MS&I. The presentation was entitled "Temporal requirements for database applications, ODMG and the TOOBIS approach". The conference was held on the 9th – 11th of September 1998 in Paris, France.

4.3.2 White Papers

1. The TOOBIS methodology by Carine Souveyet and Rebecca Deneckere.

The concept of a database application with an object- and event-driven method can be resumed in the definition of the pertinent classes of the application in describing how these objects can evolve. Focusing on the temporal aspects of a database application comes to determine:

- *which are the temporal units that rhythm the organisation life,*
- *which is the semantic and the temporal domain of each attribute,*
- *what are the data with an evolution that is pertinent to keep and to manage in the database,*
- *which are the constraints to define in order to assure the coherence of the past, present and future states of the temporal data.*

These problems are dealt with in the temporal extension of the method, by the introduction of concepts for the measure, the representation of time and the historical management. These concepts allow to take into account the temporal aspects of an application at the conceptual level.

2. ODMG Meets Time: The TOOBIS approach by Michael Souillard

In the past years a number of temporal extensions to the different database models have been proposed. Extensions to the relational model have been following the different SQL standards, while no attempts have been made to extend the OO-databases' standard, defined by ODMG. In this paper we present a temporal extension to the ODMG-93 standard, as this has been specified in the TOOBIS project. A Temporal Object Data Model, a Temporal Object Definition Language and a Temporal Object Query Language have been specified and have been proposed as extensions to the ODM, ODL and OQL of ODMG-93. This extension is under implementation on top of a commercial OODBMS.

3. A Flexible Framework for Managing Temporal Clinical Trial Data by Michael Souillard, Carine Souveyet, Costas Vassilakis and Anya Sotiropoulou.

Clinical trials are processes that produce large volumes of complex data, with inherent temporal requirements, since the state of patients evolves during the trials, and the data acquisition phase itself needs to be monitored. Additionally, since the requirements for all clinical trials have a significant common portion, it is desirable to capture these common requirements in a generalised framework, which will be instantiated for each specific trial by supplementing the trial-specific requirements, using a temporal object-oriented methodology to capture and model the requirements, a temporal OODBMS for data storage and a generalised template application, through which trial-specific applications may be generated.

4. Benchmarks for TOQL by Costas Vassilakis and Anya Sotiropoulou.

The temporal research community has formulated a number of benchmarks, which may be used to assess the functionality offered by temporal query languages. In this white paper the functionality of the TOQL language is evaluated against two such benchmarks, namely the Kalhua and Robertson benchmark and the TSQL2 benchmark. TOQL was able to express all queries, while other temporal query languages have been found to provide more limited expressive power.

4.3.3 Publications

The following publications, addressing technical and application aspects of the TOOBIS results, have been published in journals and conference proceedings:

- Michael Souillard and Yann Pollet, "Toobis: une approche pour la gestion de donnees evolutives et temporelles dans la Recherche Clinique", in Proceedings of 6th International Conference Interfaces'97, Montpellier, France, pp. 221-225.
- Carine Souveyet and Rebecca Deneckere, "Conception de bases de donnees: Aspects temporels" published in "Techniques de l'ingenieur, traite informatique" May 1998. Its reference is H3 268.
- Michael Souillard, Costas Vassilakis and Anya Sotiropoulou, "TOOBIS: application de la gestion de Donnees Temporelles dans El domain de la Recherche Clinique", in Proceedings of XVI Congres INFORSID 12-15 of May 1998, Montpellier, France, pp. 147-165.
- Anya Sotiropoulou, Michael Souillard and Costas Vassilakis, "Temporal Extension to ODMG", Proceedings of the Third Biennial Conference on Integrated Design and Process Technology, Berlin, Germany, Vol. 2, Issues and Applications of Database Technology, pp. 304-311.
- Yann Pollet and Michael Souillard, "Une approche generique pour la gestion du temps dans les Systemes d'Information basee sur des types abstraits de donnees", in BDA 98, Tunisie.

- Rebecca Deneckere and Carine Souveyet, “Patterns for Extending an OO Model with Temporal Features”, in Proceedings of 5th International Conference on Object Oriented Information Systems, 9-11 September 1998, Paris, France.

4.3.4 TOOBIS CD

Taking into account the reviewer's request, a CD was prepared to package TOOBIS dissemination material. OIP having the active support of all TOOBIS partners implemented this task. The material gathered includes user manuals, specification documents, white papers, and all presentations. The presentations included in the CD are in the form of ‘definitive’, involving all presented material in dissemination and training activities as well as into technical presentations and the project reviews of TOOBIS. Additional presentations targeting to offer a better overview on specific topics, involving speaker notes have been drafted. A presentation involving the use of speech-sound and animation has been prepared as a ‘self-running’ TOOBIS overview.

The CD is intended to be used as the means to address the wider market, starting from the established clients of Ardent Software Inc. This may result to a number of pilot projects to further test, target and improve the TOOBIS core functionality. To this end, the TOOBIS OODBMS, as well as the two applications will be packaged as self-running demos, using a commercial demo-writing tool such as the Demoshield™.

4.3.5 HTML pages

The HTML pages for the project were designed by the University of Athens. The WWW page for the project is available in English and Greek at: <http://www.di.uoa.gr/~toobis>

TOOBIS is in co-ordination with the PROSOMA activity for disseminating European Projects events and prepared the material to be used for dissemination via the PROSOMA web site.

4.4. Deliverables

	Deliverable no.	Types of deliverables	Description of deliverables (Title)	Availability C-R-P	Responsible/Involved partner
1	T11AR.1	Document	Project Initiation	P	01-P
2	T11AR.2.x	Document	Project Plan Update	P	01-P
3	T11AR.3.x	Report	Six Monthly Cost Statements and Progress Report	P	01-P
4	T11AR.4.x	Report	Three-Monthly Management Report	P	01-P
5	T12TR.1.x	Report	Phase Technical Summary	P	01-P
6	T12TR.2.x	Report (file)	Project Quality Plan	P	01-P
7	T12R.3	Report	Final Report	P	01-P

	Deliverable no.	Types of deliverables	Description of deliverables (Title)	Availability C-R-P	Responsible/Involved partner
8	T21D.1.x	Documents	User Training Material Update	P	01-P, MS&I, Ardent
9	T22TR.1	Document	Analysis of User Requirements	P	01-P, MCS
10	T23D.1.1	Document (draft)	Temporal Object Oriented Methodology	P	SORB
11	T23D.1.2	Document	Temporal Object Oriented Methodology	P	SORB
12	T23D.2	Manual	Guidelines for developing Temporal Applications	P	SORB
13	T31TR.1	Document	TODM Specifications And Design	P	MCS
14	T32TR.1	Document	TODL Specifications and Design	P	UoA
15	T33TR.1	Document	TOQL Specifications and Design	P	UoA
16	T34D.1.x	Software	TODM module	R	MCS
17	T34D.2	Document	TODM Manual	P	MCS
18	T35D.1.x	Software	TODL & TOQL module	R	UoA
19	T35D.2	Document	TODL & TOQL manual	P	UoA
20	T36D.1.x	Software	TODL & TOQL module	R	UoA
21	T36D.2	Document	TODL & TOQL manual	P	UoA
22	T37D.1.x	Software	TOOBIS S/W	R	01-P
23	T38D.1.1	Software	Optimised TOODMBS	R	O2
24	T41TR.1	Internal	Status Report	P	01-P, MCS
25	T42TR.1	Document	Evaluation Procedure Specifications	P	01-P
26	T43TR.1	Document	Design Documents of the Pilot Applications	P	01-P, MCS
27	T44D.1.x	Software	Pilot Applications	R	01-P, MCS
28	T45D.1	Document	Evaluation Document	P	MCS
29	T51D.1.x	Document	Exploitation Plan Update	P	All
30	T52D.1.x	White Papers, Brochures, Posters, HTML docs	Dissemination Material update	P	All

5. Detailed description of the TOOBIS platform

5.1. Temporal Object Oriented Methodology - TOOM

5.1.1 Objectives of the TOOBIS methodology

The conception of a database application with an object- and event-driven methodology can be resumed in the definition of the pertinent classes of the application in describing how the objects can evolve. Focusing on the temporal aspects of a database application, the following must be determined:

- Which are the temporal units that govern the organisation life.
- Which is the semantic and the temporal domain of each attribute.
- What are the data whose evolution is pertinent to keep and manage in the database.
- Which are the constraints to define in order to assure the coherence of the pasts, present and futures states of the temporal data.

These problems are dealt with in the temporal extension of the methodology by the introduction of concepts for measuring and representing time, and concepts for historical management. These concepts allow the modelling of the temporal aspects of an application at the conceptual level.

5.1.2 Overview of the OO method (OOM)

The basic OO method uses only one model, which deals with three abstractions: structure, function and behaviour. The main concepts of the model, that are used to formalise real world phenomena, along with their underlying relationships are shown in Figure 3.

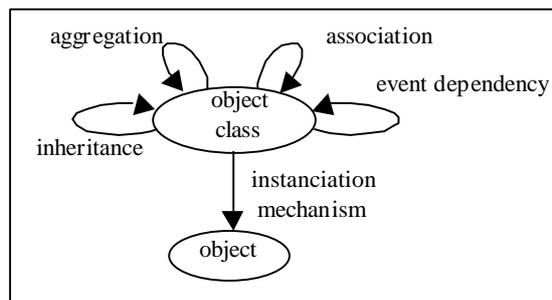


Figure 3 - Main concepts of the OO Model.

5.1.2.1 Structural properties

Two basic concepts of the model, namely object and object class, are adopted from the object-oriented paradigm. The main goal of the model is to adopt the concept of object from the analysis phase right through the implementation phase. An **object** represents real phenomena of the application's universe. An **object class** describes and groups a set of objects of the same nature in terms of the structural and

behavioural characteristics of the objects (for instance, the *patient Smith*, the *GlaxoWellcome company*, the *truck number 4477 GB 75* are objects).

Moreover, the model also takes into account the life cycles of objects for the determination of structural links (static links) between object classes, besides the inheritance links of the object-oriented paradigm. Besides the inheritance links, two types of structural links can be distinguished: **associations** and **aggregations**. An association or aggregation link may be defined between two object classes, after identifying the dependency between the life cycles of objects in these classes.

An **attribute** is a characteristic of a class. The value set of an attribute is defined with the notion of *domain* (for instance, *name*, *address*, etc. are examples of attributes of the *Customer* class. *STRING* is the domain of the *name* attribute whereas *ADDRESS* is a structured domain associated to the *address* attribute).

The structural description of an object class is completed by a set of constraints. **Constraints** may be classified into three categories:

1. *Inheritance constraints*, which model partitioning, covering and disjunction.
2. *Attributes constraints*, which restrict either the attribute values of an object, or the cardinalities of the attributes, defining related objects or domains according to a certain criterion. For instance, the *weight date* of a *patient* must be greater than the *birth date* is specified in the object class *Patient*.
3. *Uniqueness constraints*, which are defined between a set of attributes and a collection of objects. For instance, *the order number is unique in the collection of Orders*.

5.1.2.2 Behavioural properties

Besides the structural aspects, the model also considers the concepts of *event* and *operation*, so as to take into account the behavioural aspects of information systems at the conceptual level. The causal dependency of events is related to the cause and effect in the occurrence of events, and is identified by the fact that the occurrence of an event is due to the previous occurrence(s) of some event(s).

Three groups of events are distinguished: **external events**, **internal events** and **temporal events**. External events correspond to the events occurring in the environment outside the information system. Internal events correspond to the internal state changes or, rather, to the system responses to external changes (for instance, *out of stock* is an internal event detected when *an object of the stock class becomes less than 10*). External events interact with the information system through messages initiated by an actor of an **actor class** (for instance, *arrival of an order* is an external event coming from a *customer*). A **temporal event** is a particular state of the clock.

Each external event is attached to an actor class, while all temporal events are attached to the *calendar class*. Internal events are attached to object classes.

Operations are defined in object classes. They are classified under *basic operations* and *queries*. A basic operation defined in an object class represents an action that will act on an object of this class, causing a state change to that object. The execution of *basic operations* is triggered by events. Queries merely provide access services to other objects. Derived attributes are realised as queries without input parameters.

Figure 4 summarises the integration of the different abstractions (structure, function & behaviour).

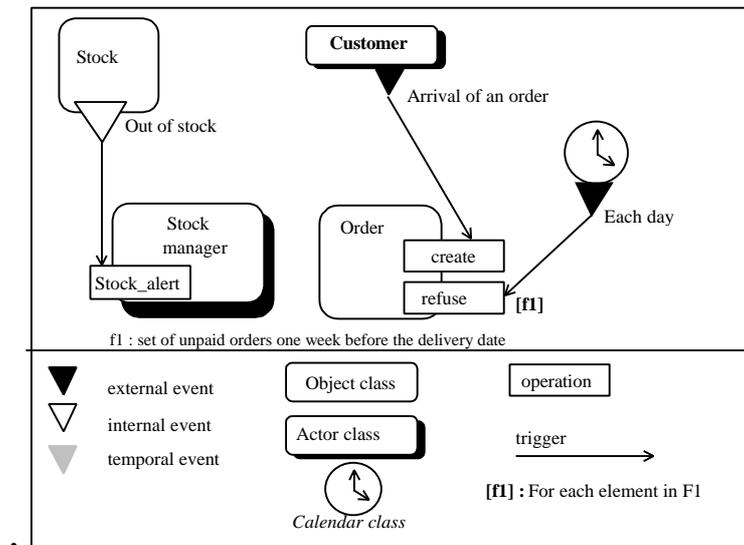


Figure 4 - Example of a dynamic schema

A **dynamic transition** [Rolland82] is made up of an event, a set of basic operations triggered by the event and a set of persistent objects manipulated by these operations. A dynamic transition is analogous to an elementary transaction of databases, that is, a unit of atomicity, consistency and integrity. In a wider perspective, a dynamic transition may be defined globally and explicitly by the sequence of event occurrences that are caused by the occurrence of an external or a temporal event. The execution of a dynamic transition must lead the database from a coherent state to a coherent state.

5.1.3 Overview of the temporal extension (TOOM)

The temporal extension aims at introducing at the analysis level, the concepts required to manage the temporal aspects of a database application. The temporal extension of the OO methodology includes the following aspects:

- definition of one or more calendars for an application,
- extension of basic domains to temporal domains,
- extension of object classes to time dimensions,
- extension of constraints applied on object classes,
- temporal extension of events.

5.1.3.1 Calendar definition

A **calendar** is a metric system to apply on the time line. The OO methodology proposes the concept of calendar class only for defining temporal events of the application. In the basic OO methodology, the Gregorian calendar is used. At the object level, the calendar class has only one instance: the clock of the Gregorian calendar.

We extend the definition of the calendar class for defining any calendar useful for the application. A calendar is described by:

- the calendar's name,
- the calendar's origin (instant),
- a basic unit (granule),
- an ordered list of granules with their finer and coarser conversions,
- an operation of translation from an instant to a period,
- an operation of conversion between granules of the same calendar and,
- two conversion operations, from and to the Gregorian calendar (these two operations facilitate conversions between calendars).

5.1.3.2 Extension of the basic domains to temporal domains

New temporal domains are used to represent time into database. There are three of them: **instant**, **period** and **interval**. An instant represents a point on the time line (e.g. *1996-12-12*). A period represents the quantity of time between two instants (e.g. *[1996-12-12, 1996-12-15)*¹). An interval, on the other hand, is an unanchored quantity of time (*3 months*). Each instance of these three types is expressed in a granularity of a calendar. Consequently, the granularity and the calendar are the instance's parameters. These generic temporal domains (*INSTANT<calendar, granule>*, *PERIOD<calendar, granule>* and *INTERVAL<calendar, granule>*) complement the classical types: *date*, *time* and *datetime* found in classical DBMSs. The domains *INSTANT-A<calendar, granule>*, *PERIOD-A<calendar, granule>* are used to specify absolute time domains. The relative time is introduced by two new temporal domains: *INSTANT-R<calendar, granule>* and *PERIOD-R<calendar, granule>*. The relative time appears as a temporal derived attribute (instead of storing the temporal value, an expression used to derive its value is stored).

5.1.3.3 Extension of object classes to time dimensions

A temporal DBMS is able to manage different kinds of objects: snapshot, historical, rollback and temporal. Snapshot objects are also managed in non-temporal DBMS (without time management). Historical, rollback and temporal objects are specific to temporal DBMS (with specific time management). Valid time is managed in historical and temporal objects whereas transaction time is included in rollback and temporal objects.

This section presents how the valid and the transaction time dimensions are introduced in object classes, either basic or derived. Two sub-classes are introduced: *snapshot class* and *temporal class*.

5.1.3.3.1 Snapshot class

A **snapshot class** is a class that does not require time management. The properties of such classes are not related to time (for instance, The *Order class* is a snapshot class because the evolution of its properties is not relevant). The following hold for snapshot classes:

¹ [1996/12/12, 1996/12/15) expresses a period including the beginning of the period but *not* including the end of the period.

- A *property* of a snapshot class can be a relationship to another class or an attribute, as it is defined in the object class.
- A *constraint* defined for a snapshot class can be an attribute constraint or an inheritance constraint, as it is described in the object class.
- *Methods* of a snapshot class can be queries or basic operations, and they can be applied at object level or class level. The types of basic operations are *create*, *update*, *delete* or *erase*. Because of the temporal consistency in the temporal database, the delete operation applied on a snapshot object is a logical deletion. The physical deletion is materialised by the erase operation.
- *Internal events* can be defined on snapshot classes.

5.1.3.3.2 Temporal class

A **temporal class** is a class with time management (cf. historical, rollback and temporal objects—for instance, the *salary of an employee* is a temporal class because its evolution is required to be kept in the database). The temporal class permits to integrate the valid and/or the transaction time management into the class definition.

The *state* of an object is composed of a set of values (a value for each property) at a given point of time. A state is associated to a timestamp (valid and/or transaction times).

A state is called *valid* when it is associated to the valid time, whereas it is called *database state* when it is related to the transaction time. *Bi-temporal states* of objects integrate their valid and its transaction times.

A *temporal class* is used to group properties that evolve synchronously along the time axis and that are linked to the same temporal dimension. It describes a temporal variation of the objects of a snapshot class. In the OO paradigm, there is at least one property that never changes, i.e. the identity of the object. Consequently, a temporal class is a temporal variation of a snapshot class. Moreover, a snapshot class can have more than one temporal variation. The link between a temporal class and its snapshot class is called a «temporal variation». For instance, an employee can be characterised by some non time-varying properties and some properties that evolve in the course of time and for which their value history must be maintained. Figure 5 depicts the fact that an employee has two temporal variations: *salary* and *family status* (these temporal variations are distinct because they evolve independently, rather than synchronously).

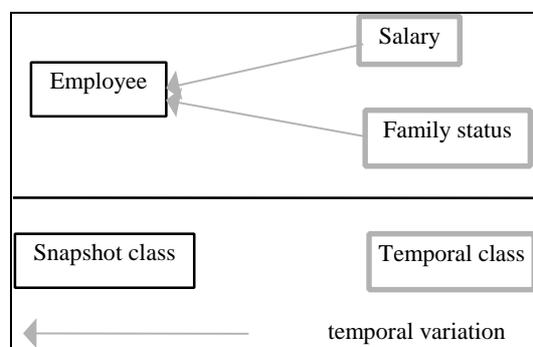


Figure 5 - Example of temporal classes

The notion of *history* is used when all states of an object must be kept, in order to retain its evolution over time. The *valid history* of an object reflects its evolution from the real world point of view (valid time management). The *database history* of an object represents its evolution according to the database viewpoint (transaction time management). And finally, a *bitemporal history* of an object combines the real world and database viewpoints.

A temporal class is characterised by the time dimension to manage. For each time dimension, two possibilities are offered: *last state* or *history*. The possible combinations from the *temporal class* concept are summarised in Table 1. Note that a class with no time dimension to manage is not handled by the *temporal class* concept but by the *snapshot class* concept.

		<u>transaction time</u>		
		none	state	history
<u>valid time</u>	none	no time management (snapshot class)	last database state of an object	database history of an object
	state	last state of an object with its valid time	last bitemporal state of an object	database history of an object and its last valid state
	history	valid history of an object	valid history of an object and its last database state	bitemporal history of an object

Table 1 - Time management defined in a temporal class.

According to the information must be retrieved, the characteristics of each time dimension that needs to be managed must be defined appropriately.

- 1) Definition of the valid time. If valid time is required, several characteristics for it have to be determined:
 - the nature of the valid time (instant or period),
 - the granularity and the calendar of the valid time timestamps,
 - the type of management (state or history),
 - the completeness of the history (complete or partial),
 - the extrapolation function if the history is partial. When the history is partial, information for some time instants may be missing from the database, and the extrapolation function is used to approximate the values for the missing instants, based on the values stored in the database for the “neighbouring” instants.
- 2) Definition of the transaction time. If transaction time is required, the type of management (state or history) must be determined.

5.1.3.4 Extension of constraints applied on object classes to time dimensions

5.1.3.4.1 Definition of a framework for classifying constraints

1. In the active OO DBMS such as [ODE], constraints are classified in two groups: *intra-object* and *inter-object* constraints. This distinction allows us to determine when they have to be verified.
 - An intra-object constraint is defined locally to an object and can be checked at the end of each method execution (for instance, *the weight of a patient can not exceed 200 kilos*).
 - An inter-object constraint uses several objects; thus it has to be checked at the end of the database transaction (for instance, a *referential constraint*).
2. The second classification comes from the temporal database field and groups the constraints in two main categories: *intra-time* and *inter-time* constraints.
 - An *intra-time* constraint is only dependent on an instant t . Each instant t should satisfy the constraint. Three flavours of constraints may be defined, corresponding to the time dimensions under consideration: intra-VT, intra-TT and intra-Bi constraints. For instance, *for each week declared in the Weekly effort, an assignment of this employee for this project must exist* is an intra-VT constraint; *the name of a company must be unique in each database state* is an intra-TT constraint; and *the valid time of a weight value cannot be greater than its transaction time* is an intra-Bi constraint.
 - An *inter-time* constraint is defined by using information available or valid at different instants. Each instant t should satisfy the constraint. Three flavours of constraints may be defined, corresponding to the time dimensions under consideration: inter-VT, inter-TT and inter-Bi constraints. For instance, *the valid timestamp must be unique in the temporal class with history<VT>* is an inter-VT constraint, *the name of a person cannot change more than three times* is an inter-TT constraint and *the weight of a patient for a specific valid time cannot be corrected more than three times* is an inter-Bi constraint.

5.1.3.4.2 Introduction of specific constraints to temporal features

In the OO methodology, the constraints are defined firstly on object classes: uniqueness constraint, inheritance constraint and attribute constraints. Constraints are also embedded in the definition of aggregations and associations such as cardinality constraint and referential constraints. The extension of these constraints to include temporal aspects affects only referential, uniqueness and attributes constraints.

5.1.3.5 Temporal extension of events

The *event* is the concept used in the OO method to define the dynamics of the application. Three kinds of events are defined:

- An **external event** models the emission of a message by an actor. The trigger part of this event represents the actions to perform when such messages arrive to the system. This action is the response of the information system to this message. The valid time of an external event is

the valid time associated to the message. External events may be classified into three categories, with respect to their valid times:

An **event in time** is an event whose valid time is the present time instant.

An **“a priori” event** has a valid time in the future (allows triggering of actions in advance).

An **“a posteriori” event** has a valid time in the past.

- An **internal event** acknowledges a noteworthy state change of an object. The valid time of an internal event is restricted in TOOM to the “event in time”. An event based on the state change of an object is called **event on object**. The extension of object classes to temporal classes with history implies a new class of events based on the state of the history instead of the state of an object. For instance, *If an employee has increased his salary four times during the last three years*, the staff department is alerted. This event is called **event on history**.
- A **temporal event** is a particular state of the clock. Temporal events are modelled via temporal expressions based on one of the calendars of the application. If the application requires several calendars, temporal events are bound to a specific calendar.

Three types of temporal events are defined: *absolute*, *relative* and *periodic* events. For instance, the “*25th of December 1996*” is an **absolute event**, “*each 25th of December*” is a **periodic event** and “*1 month after the expiration of the loan*” is a **relative event**.

In the temporal extension the relative event is refined according to the relative time domains. It has a relative time expression as predicate. Two kinds of temporal marks are possible: **temporal attribute** or **time occurrence of event**. For instances, “*1 month after the expiration of the loan*” is an *event relative to an object* of the loan class and the temporal mark is the attribute *end of loan*; “*15 days after the arrival of an order*” is an *event relative to the event* “*arrival of an order*”.

5.1.4 Conclusion

The TOOBIS methodology helps the analysts to describe the temporal aspects of database applications and supports the mapping of the conceptual specification to the TOOBIS TOODBMS with its definition language TODL.

The temporal extension tackles the following aspects:

- the measurement of time through calendar definition,
- the representation of time through new temporal domains,
- timestamping and keeping a history of data with the temporal class, and its new static links,
- Definition of the integrity of temporal data with the extension of static constraints.

In addition, behaviour in temporal applications is an important aspect to study. This is why we choose for OOM an event-oriented approach and a strong integration of behavioural modelling with the two other perspectives.

In the OOM approach, the behaviour of the application is described in terms of events and triggering rules. An event is a particular state change of objects, which plays the role of a stimulus implying the system reaction. The reaction of the system to an event is expressed through triggering rules in terms of operations to perform on objects. This is a causal approach of dynamics in the sense of cybernetics and causal theory of organisations. It has numerous advantages among which are (a) the possibility of both a global and a local study of the system behaviour and (b) a tight coupling of the three perspectives. The important principle of minimality (minimum set of concepts for a maximum representability) which is a well-established principle in science is fully followed.

5.2. Temporal Object-Oriented DBMS (TOODBMS)

The TOODBMS comprises of three modules, namely the *Temporal Object Data Model* (TODM), the *Temporal Object Definition Language* (TODL) and the *Temporal Object Query Language* (TOQL). These modules are described in the following paragraphs.

5.2.1 Temporal Object Data Model - TODM

TODM is an extension of the Object Data Model –ODM– on top of which the ODMG-compliant OODBMSs are built. By extending this model, TODM aims at being portable on any of these OODBMSs.

Before introducing temporal data within object-oriented concepts, the time model used by TODM is described. Then the structures and interfaces used for dealing with temporal data will be presented.

5.2.1.1 Time Model and Manipulation

TODM is based on a classical and almost standard time model. It is a temporal and linear structure where a total order is defined, using the “inferior to” operator. TODM handles a discrete view of this model. The time axis can be divided in a finite number of smaller segments called *granules*. The smallest granule is called *chronon* and its size is implementation-specific. To manipulate time quantities the following entities are defined:

- an *instant* is a time point on the time axis, e.g. “1997-09-16”;
- a *period* is a quantity of time between two instants, called boundaries, e.g. “[1997-09-01, 1997-10-01)”;
- an *interval* is a duration of time with known length, but without specified boundaries –e.g. “1 month”– which may appear as a time window along the time axis.

TODM provides full support for the standard Gregorian calendar with its standard granules –year, month, day, hour, minute and second– as well as provision for multi-calendar support. Each time quantity, anchored on not, is expressed in a given calendar and at a specific granularity. For the Gregorian calendar, leap years and seconds, and time zone specifications are supported. For instance, “1997-09-16 15 MET DST”, is an instant expressed in the Gregorian calendar, at hour-sized

granularity, representing the sixteenth of September 1997, at 3:00 p.m., expressed in the Middle European Time time zone, using Daylight Saving Time - GMT²+2.

A set of arithmetical and comparison operations, such as *precedes*, *meets*, and so on, is provided for instant, period and interval manipulation. Relative time, as opposed to absolute dating, is also implemented within TODM time support. A form of late binding is used to represent specific instants such as *Now*, *Beginning* (the smallest instant on the time axis), *Forever* (the greatest instant on the time axis) etc.

5.2.1.2 Temporal Data within Objects

The ODM of ODMG-93 defines the characteristics of objects and how they can relate to each other. The basic primitive is the object, which has a unique identifier (object identifier-OID) constant over time. Its state is defined by the values carried by the instance properties –attributes and relationships– and its behaviour is defined by a set of operations. An attribute is of one type, whereas a relationship is defined between two types, which must have instances referencable by OIDs. Objects are entities whose values, i.e. states, can evolve over time. TODM is designed to maintain such evolutions over the valid and/or transaction time dimensions.

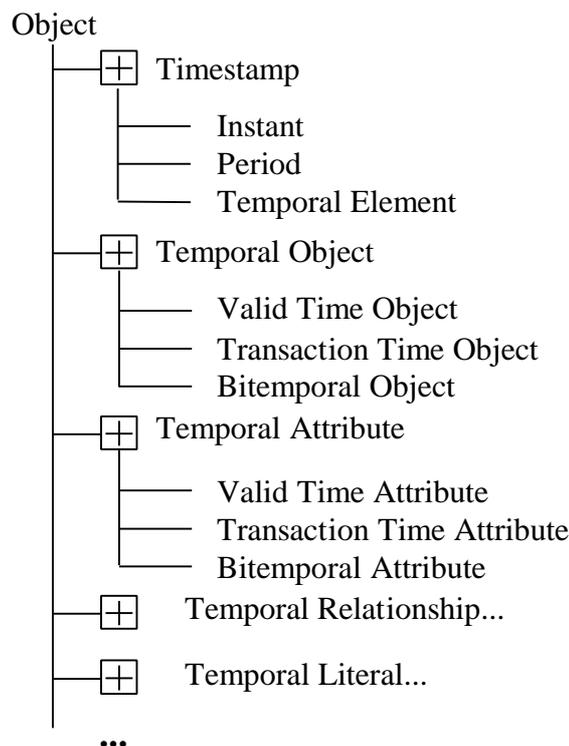


Figure 6 – Extensions to the ODMG type hierarchy

TODM supplements the type hierarchy defined by ODMG, introducing types for storing and manipulating temporal data. For each type defined by ODMG, temporal counterparts are defined by TODM, allowing for evolution along the valid time axis, the transaction time axis or both time axes. Figure 6 depicts a portion of the

² GMT: Greenwich Meridian Time

extensions to the ODMG type hierarchy that are introduced by TODM. These extensions are provided both at instance property level and at object level by introducing sub-types of attribute, relationship and object types. Temporal features can not be nested. On one hand, an object that varies over one or both of the time dimensions can evolve as a whole, i.e. the evolution of all its instance properties which form its state will be maintained. On the other hand, an object can have instance properties, which can vary independently over one or both time dimensions. In this case, the temporal characteristics are defined on the specific instance properties and not at the object level.

Figure 7 illustrates the difference between temporal instance properties and temporal objects. The value of each temporal entity at a given instant (or period) along with the applicable timestamps is called *a variant*.

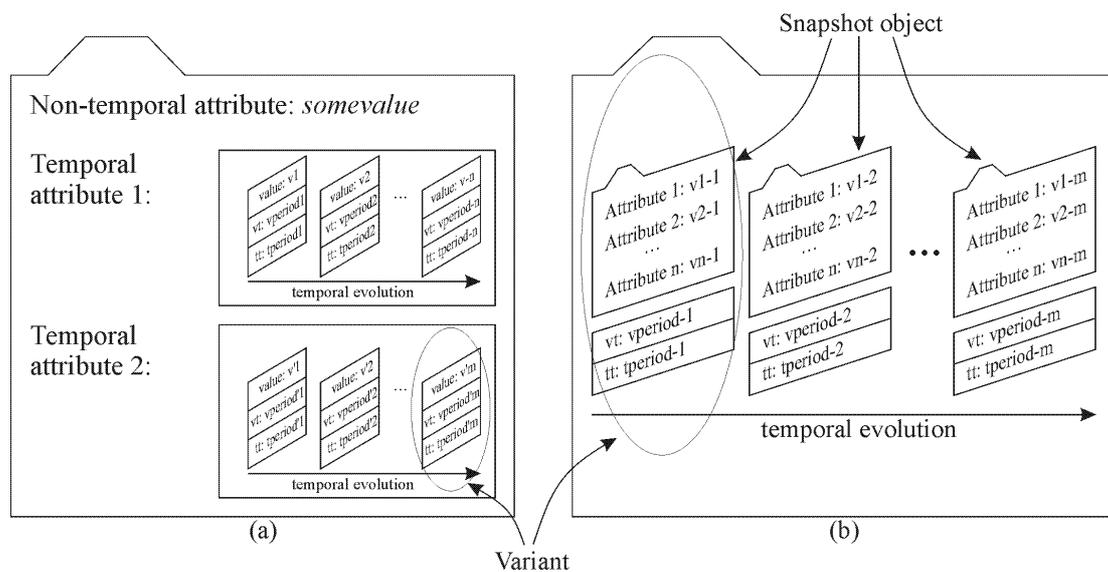


Figure 7 - Temporal instance properties vs. temporal objects

The way to identify an object is always via its unique OID, regardless of whether the object is snapshot or temporal. However, to access a value of a temporal object, i.e. one of its states, a timestamp has to be added to this OID. Depending on the time dimensions handled, this timestamp will be an instant or a period. Accessing temporal data for temporal instance properties is done in a similar way. The value of an attribute or a relationship is accessed using the selected object and the name of the instance property. To access a value of a temporal attribute or temporal relationship, a timestamp must be added to the (object, instance property name) couple. To support upwards compatibility with snapshot objects, the result of accessing a temporal entity without any timestamp argument, is the current value of the entity: the valid and current in the database value at the access execution time.

A relationship, which is always defined between two instanciable types, can connect either temporal or non-temporal types, since the OID is always used to access any object. A relationship, whether it has temporal characteristics or not, can be defined between types which may or may not have temporal characteristics. However, some restrictions are imposed for inverse links, due to time dimensions support. For example, a non-temporal relationship between two temporal types, the one evolving over valid time and the second over transaction time, can not have an inverse link, since the objects evolve on different time axes with no common portion. The inverse

relationships, when allowed, can be classified as symmetrical or asymmetrical ones. A symmetrical relationship is one between two temporal types evolving over the same time dimension. An asymmetrical one appears when a temporal relationship (which can only be defined in a non-temporal type) points towards a temporal type evolving on the same time dimension. The restrictions imposed on the existence and type of inverse relationships, along with the information needed to model each kind of relationship are illustrated in Table 2.

Table 2 – Temporal Relationships and Inverses

From / To		Snapshot Object: S2	Historical Object: H2	Rollback Object: R2	Bitemporal Object: B2
Snapshot Object: S1	Snapshot Relationship	OID-S1, OID-S2 inverse: snapshot, symmetrical	OID-S1, OID-H2 No inverse	OID-S1, OID-R2 no inverse	OID-S1, OID-B2 no inverse
	Historical Relationship	OID-S1, VT, OID-S2 inverse: historical, symmetrical	OID-S1, VT, OID-H2 Inverse: asymmetrical	OID-S1, VT, OID-R2 no inverse	OID-S1, VT, OID-B2 no inverse
	Rollback Relationship	OID-S1, TT, OID-S2 inverse: rollback, symmetrical	OID-S1, TT, OID-H2 No inverse	OID-S1, TT, OID-R2 inverse: asymmetrical	OID-S1, TT, OID-B2 no inverse
	Bitemporal Relationship	OID-S1, BT, OID-S2 inverse: bitemporal, symmetrical	OID-S1, BT, OID-H2 No inverse	OID-S1, BT, OID-R2 no inverse	OID-S1, BT, OID-B2 inverse: asymmetrical
Historical Object: H1		OID-H1, VT, OID-S2 inverse: historical, asymmetrical	OID-H1, VT, OID-H2 Inverse: symmetrical	OID-H1, VT, OID-R2 no inverse	OID-H1, VT, OID-B2 no inverse
Rollback Object: R1		OID-R1, TT, OID-S2 inverse: rollback, asymmetrical	OID-R1, TT, OID-H2 No inverse	OID-R1, TT, OID-R2 inverse: symmetrical	OID-R1, TT, OID-B2 no inverse
Bitemporal Object: B1		OID-B1, BT, OID-S2 inverse: bitemporal, asymmetrical	OID-B1, BT, OID-H2 No inverse	OID-B1, BT, OID-R2 no inverse	OID-B1, BT, OID-B2 inverse: symmetrical

Legend: VT: Valid time timestamp
 TT: Transaction time timestamp
 BT: Bitemporal timestamp

In ODMG, relationships are defined between two types. An 1-1 relationship from an object A to an object B, for instance, can be represented using a pointer from A to B. The pointer can be implemented using the object identifier of B. By introducing temporal objects, which can be seen as collections of states of objects, TODM also introduces a new kind of relationship: **state relationships**. A state relationship does not point towards an object, but towards a specific state of a temporal object. In this case, the OID of the target temporal object is not sufficient to model the state relationship. Instead, a Temporal-OID (TOID) –the OID of the target temporal object plus the timestamp associated with one of its states– should be used, which allows to precisely select the state of the temporal object involved in the state relationship. Figure 8 illustrates traditional relationships and state relationships.

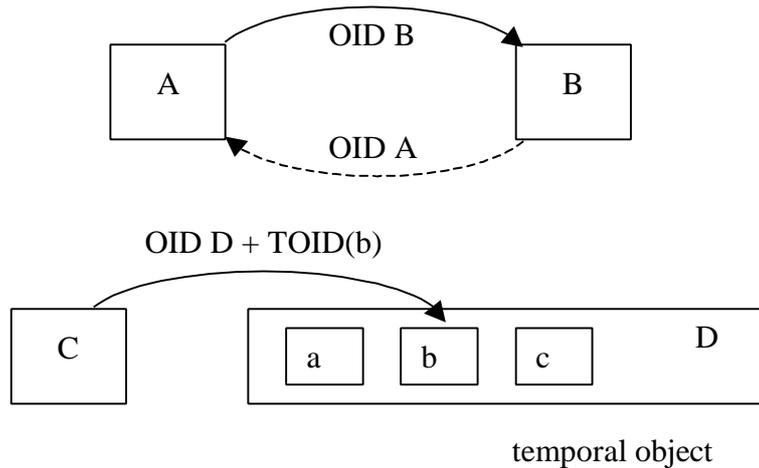


Figure 8 - Relationships and state relationships

5.2.2 Temporal Object Definition Language - TODL

TODL is a definition language that enables user to define temporally enhanced interfaces. TODL is in line with the ODMG's ODL concepts since:

- it supports all semantic constructs of the underlying data model (TODM).
- it is a definition language for object specifications.
- it is programming language independent.

TODL deals with the definition of interfaces and instance properties for Temporal Objects. It allows the definition of temporal data on both instance property and object level, as it is supported by TODM. Figure 9 illustrates some of the features supported by TODL. In this example *Patient* is an interface containing two temporal instance properties, namely *room* and *doctor*, evolving along the valid time axis and both time axes, respectively. For temporal properties evolving along the valid time axis, the user may specify the granularity and the calendar used for storing the valid time timestamps, whereas no such specifications may be used for the transaction time axis, since transaction time timestamps are always represented at the finest granularity of the Gregorian calendar. Temporal instance properties within an interface may evolve independently along the temporal axes. Additionally, the *stay* instance property models *user-defined* time, providing the place for storing a *Period*-type datum that represents the hospitalisation period of the patient. User-defined time may be expressed using any type for time representation supported by TODM (*Interval*, *Instant*, *Period*, *Period_Set*) and the user may always specify the granularity and calendar that will be used with the timestamps.

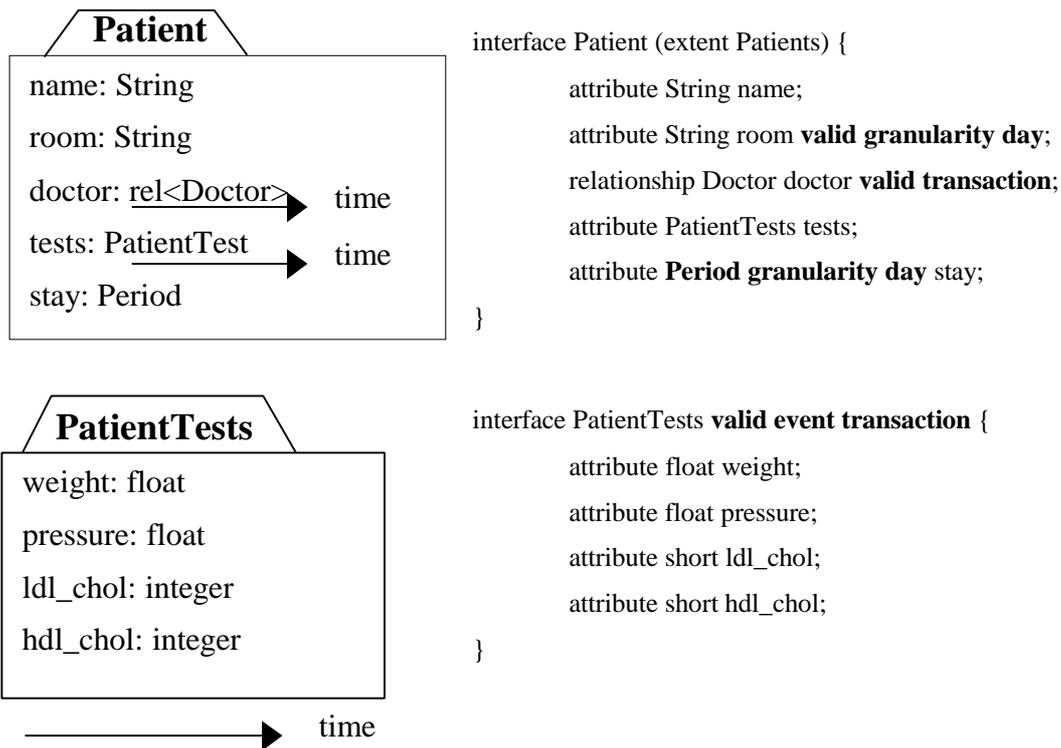


Figure 9: TODL definition of a temporal object

On the other hand, *PatientTests* is an interface evolving along both time axes as a whole. Effectively, this means that *all* instance properties of the interface evolve along the time axes, and this evolution is synchronised.

```

calendar Gregorian {
    origin "0001-01-01, 00:00:00";
    granules
        Year Month: 12;
        Month Day: 30;
        Day Hour 24;
        Hour Minute 60;
        Minute Second 60;
        Second chronon;
}

```

Figure 10 – Calendar definition in TODL

Finally, TODL provides facilities for calendar definition. Users may define custom calendars so as to meet the temporal support needs presented by the applications, e.g. financial applications may need to operate using the fiscal calendar, whereas an application for an academic institute may need to express time in the Academic calendar (year of two semesters, with the beginning of the year set in September). An example of calendar definition is presented in Figure 10. For each calendar, the user needs to specify a name, the earliest time point that can be expressed in the calendar (the calendar *origin*), the granules which are valid for this calendar and mappings between the granules (i.e. the number of lower-level granules comprising one higher-level granule). When this mapping is not constant (e.g. mapping between days and months), the user needs to supply a method for performing the appropriate

computations. Finally, the user must supply the format that will be used to display timestamps to the users.

5.2.2.1 TODL module architecture

TODL has been implemented as a modular processor, operating on top of the O₂ System, as illustrated in Figure 11. The processor accepts as input a file with TODL declarations, which is initially processed by the lexical and syntactical analyser modules, so as to determine whether the declarations are conformant to the languages syntax rules. Once this step is complete, another validation check is conducted, during which it is verified that all the types referenced in the declarations file are existing or newly defined types (data from the metaschema are used during this check), and that each type is used properly. Relationship constraints are imposed here too, since inverse relationships must be appropriately declared in both types, and inverse relationships between temporal entities are only allowed when certain preconditions are met (see Table 2).

The semantic analysis phase that follows, identifies the actions that must be performed. These actions are different in the case that an object type is defined for the first time and in the case that an object type is redefined. In all cases, the metaschema information is updated accordingly. Metaschema information need to be maintained by the TODL processor, since the O₂ system is not aware of the temporal semantics of the data, and thus the additional semantics introduced in the TDBMS cannot be handled by the O₂ system.

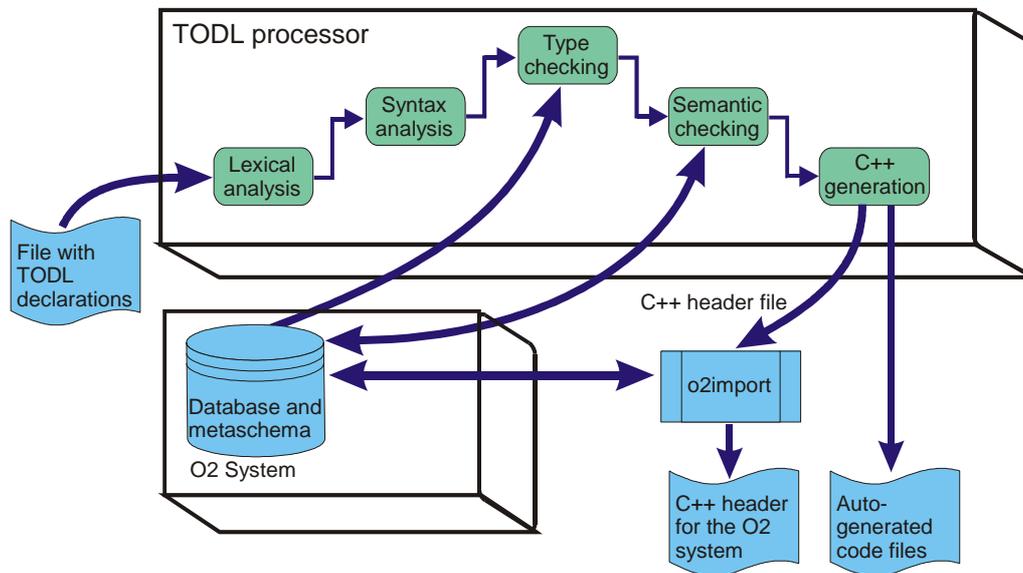


Figure 11 - TODL Processor Architecture

The final step of the TODL processor is the generation of C++ declarations for the classes defined (or redefined) in the TODL file. These declarations are fed to the *o2import* tool, which interacts with the O₂ DBMS for actually defining the new classes at the DBMS level.

Additionally, TODL relieves the programmers from the burden to write code supporting common operations on temporal objects, by automatically generating the appropriate pieces of code, which can be readily incorporated into the applications.

5.2.3 Temporal Object Query Language - TOQL

TOQL is a temporal extension to the OQL, the ODMG standard for querying object databases. The design objectives of TOQL were to formulate a temporal extension of OQL, delivering full temporal functionality, but yet remaining simple and treating uniformly temporal and non-temporal data. Another goal pursued in TOQL's design phase was the provision of the maximum degree of compatibility for legacy applications, i.e. applications that were written with a non-temporal schema in mind. The temporal features of TOQL are briefly discussed in the following paragraphs.

5.2.3.1 Data types for time representation

A TOQL query may contain constants representing time quantities, i.e. intervals, instants, periods and period sets. These constants may be used in any place a constant value is allowed in an OQL query. Table 3 presents some examples on constant time quantity specification in the context of a TOQL query.

Literal	Value
instant '1990' granularity year calendar Gregorian	An instant for the year 1990 of the Gregorian calendar.
instant 'now' granularity minute	The current date and time, expressed in a granularity of minute in the Gregorian (default) calendar.
instant 'Spring 1996' granularity semester calendar Academic	An instant corresponding to the Spring semester of the academic year 1996. It is expressed in the "Academic" calendar, having a granularity of "Semester".
interval '5' granularity year	An interval of 5 years in the Gregorian (default) calendar.
interval '2' granularity semester calendar Academic	An interval of two semesters in the academic calendar.
period '[1990, 1991)' granularity year calendar Gregorian	The year 1990 of the Gregorian calendar.
period '[now, 2000-01-01)' granularity day	A period starting from the current time and ending at the first day of year 2000. The period is expressed in the Gregorian (default) calendar.
period '[Winter 1996, Spring 1997)' granularity semester calendar Academic	A period starting at the Winter semester of the academic year 1996 and ending at the Spring semester of the academic year 1997.

Table 3 – Examples of time literal specification in TOQL

Additionally, TOQL provides predicates that may be used to test the relative position of time quantities (e.g. *instant1 precedes instant2* yields *true* if *instant1* occurs earlier on the time axis than *instant2*) and functions that operate on time quantities (e.g. *duration(interval1)* returns the duration of *interval1* in granules).

5.2.3.2 Referencing Temporal Data in Queries

When a temporal instance property (or an instance property of a temporal object) is referenced within a TOQL query, the language's default behaviour is to use the *current value* of the temporal instance property (the value of the *current variant* of the temporal object, respectively). The reason for this behaviour is twofold:

1. the majority of queries accesses the current value of temporal data, thus this behaviour facilitates the formulation of more compact queries.
2. legacy applications (i.e. applications developed on top of snapshot database schemata that were later enriched with temporal dimensions) can continue to operate, without any need for code modification.

The default behaviour may be overridden using the modifiers *valid*, *transaction* and *bitemporal*, which allow for retrieval of the valid time history, transaction time history and bitemporal history, respectively. When whole temporal objects, rather than specific instance properties, are referenced, no automatic conversion is performed. This does not jeopardise the compatibility for legacy applications, since once objects are brought into memory, applications interact with the objects through the available methods and do not directly access instance properties.

Examples

1. `select name, room from Patients`

This query retrieves the name and the *current* room for all patients (room is a valid time instance property). Note that this is the same query that would be used to retrieve the same information from a snapshot database.

2. `select name, valid room from Patients`

This query retrieves the name the whole history of room assignments for all patients.

5.2.3.3 Selecting Information from Temporal Data

TOQL treats temporal data as indexed collections, orthogonally to lists and arrays. Elements in these collections may be indexed by one of the following methods:

1. using a *timestamp* (instant or period), which selects the element (or elements) which contain data pertaining to the designated time instant (or period). For example, the query

```
select name, (valid room)[instant '1999-01-01'] as room
from Patients
```

selects, for each patient, the name and the room assignment on January 1st 1999, while the query

```
select name,
      (valid room)[period '[1999-01-01, 1999-02-01]'] as room
from Patients
```

retrieves, for each patient, the name and all his/her room assignments within January 1999. If the object evolves along both time dimensions, the subscript expression must designate the time axis on which information selection will occur.

2. using an *integer*, designating the *order* of the desired element within the temporal datum. Elements (variants) of temporal data are considered to be ordered in an ascending sequence with respect to their timestamps. Thus the query

```
select name, (valid room)[0] as room
from Patients
```

retrieves, for each patient, the name and his/her first room assignment, while the query

```
select name, (valid room)[3] as room
from Patients
where count(valid room) > 3
```

retrieves, for each patient having four or more room assignments, the name and his/her fourth room assignment. This selection mechanism may be used only when the temporal datum evolves along a single temporal axis.

3. using a *pair of integers*, designating the *first and last order* of the desired elements within the temporal datum. Elements (variants) of temporal data are considered to be ordered in an ascending sequence with respect to their timestamps, similarly to case (2). Thus the query

```
select name, (valid room)[0:3] as rooms
from Patients
where count(valid room) > 3
```

retrieves, for each patient having four or more room assignments, the name and his/her first four room assignments.

A query may also iterate through the elements (variants) of a temporal datum, in order to perform comparisons and/or calculations over the variants' value. The mechanism for defining variables iterating over the elements of a temporal datum is the same mechanism that enables iteration over a collection's (set, bag, list, array) elements, i.e. the variable may be defined in the *from* clause or in the universal and existential quantification clauses (*for all variable in datum*, *exists variable in datum*). If a variable *v* iterates over the variants of a temporal datum, the variable name *v* corresponds to the variant's value, whereas the notations *valid(v)* and *transaction(v)* may be used to access the valid time and transaction time timestamps of the variant, respectively.

Examples

1.

```
select name, r
from Patients as p, valid p.room as r
where r like '1*'
```

This query retrieves the patient names and the room assignments for all patients that were assigned to a room on the first floor.

2.

```
select name, r, valid(r) as when
from Patients as p, valid p.room as r
where valid(r) > instant '1999-01-01'
```

This query retrieves the patient names and the room assignments, and the room assignment periods for all room assignments made after January 1st, 1999.

5.2.3.4 Partitioning and Aggregating on the Temporal Dimensions

TOQL enables the partitioning of temporal data to *variant subsets*, with each variant subset pertaining to a specific portion of the time axis. For example, the valid time datum storing the room assignments for a hospitalised patient may be split into variant subsets, with each subset containing the room assignments for each calendric year. Each such variant subset may then be used for report formulation (e.g. produce a report on the room assignments of a specific patient for each year) or aggregate value computation (e.g. compute the number of room assignments for each year, for a specific patient). This partitioning mechanism is invoked through a special syntactic construct of TOQL, illustrated in the following example:

Example:

The following query retrieves the room assignments of a patient named “Some patient” split in subsets, with each subset pertaining to a calendric year.

```
select r.timeslice, r.partition
from Patients as p,
    (partition valid as interval '1'
     granularity year as calendar)(valid p.room) as r
where p.name = "Some Patient"
```

Additionally, TOQL extends the standard grouping mechanism designated in the ODMG standard (the *group by* clause), so as to allow formulation of groups based on the timestamps of the variants. Variant groups may then be used for aggregate value computation.

5.2.3.5 Advanced Features of TOQL

Besides the features presented in the paragraphs above, TOQL provides additional features facilitating the conversion between snapshot and temporal domains, the combination of temporal information contained into different temporal objects through *temporal joins*, the conversion from period timestamping to instant timestamping, and variant rearranging so as to formulate maximal timestamps either on the valid time axis or on the transaction time axis. For more information on these features, the reader is referred to the TOQL language specification [UoA96].

5.2.3.6 TOQL processor architecture

The TOQL component has been designed as a query preprocessor to the OQL, and uses the features of the temporal TODBMS structures stored in the TODM. The overall architecture of the TOQL processor is depicted in Figure 12.

TOQL queries may originate either from interactive users, submitting ad-hoc queries through a database terminal monitor, or from applications that submit queries through the embedded TOQL API (application programming interface). Both interactive and embedded queries are first processed by the lexical and syntax analysis modules, in order to determine whether the submitted query conforms to the syntax rules of TOQL. If this step completes successfully, type checking is conducted to verify that types are used properly. TOQL follows closely the guidelines governing OQL, which state that operations may be freely mixed in the context of a query, as long as they respect the language’s type system. This step uses information from the database metaschema, which is retrieved from the database when the session (interactive or non-interactive) is initiated.

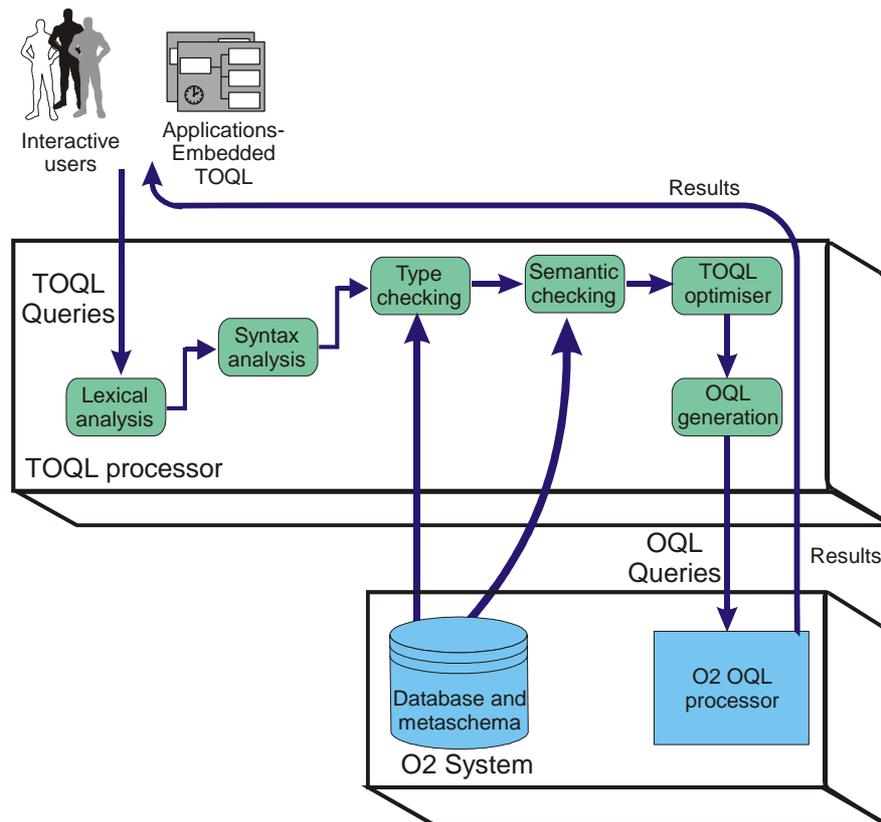


Figure 12 - TOQL Processor architecture

Having verified that the query is syntax-wise and type-wise correct, the semantic analysis module determines the actual operations that must be performed, in order to evaluate the query. This step involves the addition of extra operations which perform automatic conversions of temporal data to their snapshot values, wherever appropriate. Metaschema information is used in this phase too. An optimisation step follows, which reduces the number of implicit conversions to a minimum and limits method invocations, so as to enable the O₂ system's optimiser to formulate a more efficient query execution plan.

The final step of the TOQL processor is the formulation of an OQL query, which is submitted to the OQL processor of the O₂ system. The result returned by the O₂ system is collected and returned to the user (or application) that submitted the original TOQL query.

6. Overview of the Pilot Applications

6.1. The Delta S.A. DSS Application

Delta S.A. is the largest dairy products company in Greece. Many of dairy products manufactured by Delta S.A. have a limited shelf-life time (2-4 days, depending on the product), and if this life time expires products are returned to the company. Customers are not charged for returned products, thus product returns lead to considerable profit losses for the company, and consequently Delta S.A. is interested in minimising these returns.

In order to minimise product returns, Delta S.A. must be able to accurately predict dairy product consumption within the next few days, so as to supply its customers with the appropriate dairy product quantities. Over-estimation of the consumption will lead to the production and distribution of excessive product quantities, which will be finally returned to the company, whereas under-estimation will lead to product shortage and thus sales reduction.

Predictions of the dairy product consumption within the following days are based on historical data, taking into account the period of the year, the day of the week, special events such as national and religious holidays, weather conditions etc. To this extent, flexible means for storing and manipulating historical data are required, and the TOODBMS provides these facilities. Daily consumption data, along with additional information such as local weather conditions, are uploaded to the temporal database from the distributors' handheld computers, through an automated process. Each distributor records in his/her handheld computer all data concerning product quantities loaded and delivered to local customers, as well as quantities of expired products returned to the company. Once this information has been entered in the temporal database, sophisticated TOQL queries are employed to:

1. calculate the quantities of products that will be manufactured and loaded on the trucks for delivery.
2. produce accurate reports for the sales & distribution of dairy products per route, per distributor and per product.

By exploiting the TOODBMS temporal capabilities and the expressive power of the Object Oriented model, these queries have replaced complex and voluminous code used in the previous system for the same purposes, while the time needed to make the necessary computations has also been reduced.

Besides handling historical data on the production, loading, consumption and returns of dairy products, the DSS application developed for Delta S.A. handles all other aspects related to product manufacturing and distribution, such as:

- customer administration (gain of customers from competitive companies, loss of customers, start of partnership date, etc).
- route administration, i.e. management of the routes followed by Delta S.A.'s distribution fleet. Route administration is used to keep track of the defined routes, the customers served by each route, the products delivered by each distribution truck etc.
- product administration, including the introduction and discontinuation of products, shelf-life time and price changes, packaging details and so on.

Taking these into account, the DSS application constitutes a system supporting all the information requirements concerning product production and distribution. This system runs on the Windows NT platform, and has been implemented using the C++ TOOBIS bindings and libraries for database interaction, and the MS Visual development environment for the realisation of the user interfaces. The following figures present sample interaction screens of the Delta Dairy S.A. DSS application.

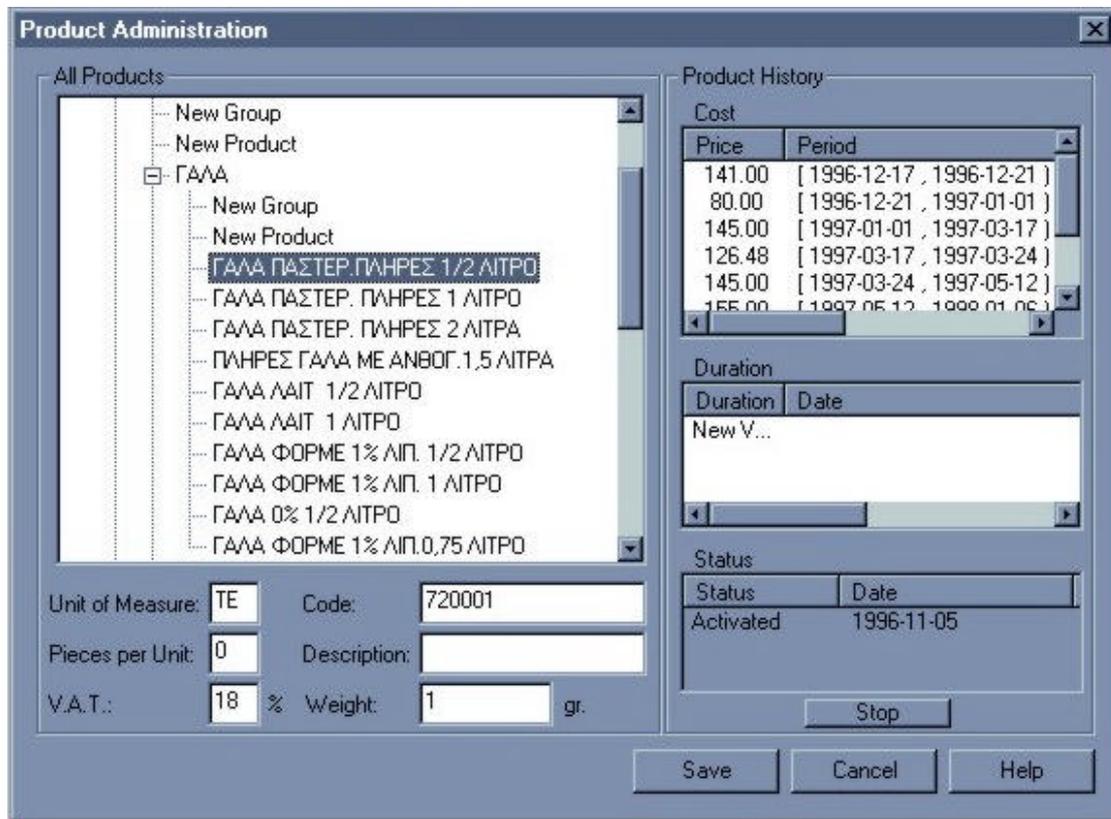


Figure 13 – Interaction screen for product administration

Load planning

Load planning for: 22 January 1998

Route: FCN001A Distributor: ΚΟΣΣΥΒΑΚΗΣ - ΔΗΜΗΤΡΟΥΛΑ

Event Occurrence

Event: Rain

Influence: -1 %

	Order	Yesterday	1 week before	2 weeks before	Aver
ΓΑΛΑ ΠΑΣΤΕΡ. ΠΛΗΡΕΣ 1/2 ΛΙΤΡΟ [720001]	29	29	29	37	
ΓΑΛΑ ΠΑΣΤΕΡ. ΠΛΗΡΕΣ 1 ΛΙΤΡΟ [720002]	67	66	67	70	
ΓΑΛΑ ΠΑΣΤΕΡ. ΠΛΗΡΕΣ 2 ΛΙΤΡΑ [720003]	28	26	30	30	
ΠΛΗΡΕΣ ΓΑΛΑ ΜΕ ΑΝΘΟΓ. 1,5 ΛΙΤΡΑ [720004]		1			
ΓΑΛΑ ΛΑΙΤ 1/2 ΛΙΤΡΟ [720005]	8	9	9	10	
ΓΑΛΑ ΛΑΙΤ 1 ΛΙΤΡΟ [720006]	14	14	15	16	
ΓΑΛΑ ΦΟΡΜΕ 1% ΛΙΠ. 1/2 ΛΙΤΡΟ [720007]					
ΓΑΛΑ ΦΟΡΜΕ 1% ΛΙΠ. 1 ΛΙΤΡΟ [720008]					
ΓΑΛΑ 0% 1/2 ΛΙΤΡΟ [720009]	4	6	5	6	
ΓΑΛΑ PLUS 0,75 ΛΙΤΡΟ [720010]	4	5	5	4	

OK Cancel

Figure 14 - Interaction screen for load planning

Delivery

Delivery Date: 07 January 1998

Route: FCN001A Distributor: ΚΟΣΣΥΒΑΚΗΣ - ΔΗΜΗΤΡΟΥΛΗΣ Ο.Ε.

Save Cancel

Today's Route Performance

	Plan Load Quantit	Load Quantity
ΓΑΛΑ ΠΑΣΤΕΡ. ΠΛΗΡΕΣ 1/2 ΛΙΤΡΟ [720001]	37	37
ΓΑΛΑ ΠΑΣΤΕΡ. ΠΛΗΡΕΣ 1 ΛΙΤΡΟ [720002]	70	70
ΓΑΛΑ ΠΑΣΤΕΡ. ΠΛΗΡΕΣ 2 ΛΙΤΡΑ [720003]	30	30
ΠΛΗΡΕΣ ΓΑΛΑ ΜΕ ΑΝΘΟΓ. 1,5 ΛΙΤΡΑ [720004]		

Route Performance

Customer Sales and/or Returns

Customer: ΑΓΓΕΛΙΔΗ ΑΦΟΙ & ΣΙΑ ΟΕ Save Delivery

	Sold Quantity	Returned Quantity
ΓΑΛΑ ΠΑΣΤΕΡ. ΠΛΗΡΕΣ 1/2 ΛΙΤΡΟ [720001]		
ΓΑΛΑ ΠΑΣΤΕΡ. ΠΛΗΡΕΣ 1 ΛΙΤΡΟ [720002]		
ΓΑΛΑ ΠΑΣΤΕΡ. ΠΛΗΡΕΣ 2 ΛΙΤΡΑ [720003]		
ΠΛΗΡΕΣ ΓΑΛΑ ΜΕ ΑΝΘΟΓ. 1,5 ΛΙΤΡΑ [720004]		
ΓΑΛΑ ΛΑΙΤ 1/2 ΛΙΤΡΟ [720005]		
ΓΑΛΑ ΛΑΙΤ 1 ΛΙΤΡΟ [720006]		
ΓΑΛΑ ΦΟΡΜΕ 1% ΛΙΠ. 1/2 ΛΙΤΡΟ [720007]		

Product delivery

Figure 15 - Interaction screen for delivery data

6.2. The Glaxo Wellcome Clinical Trial Management Application

Glaxo Wellcome is one of the pioneering companies in medical research. When new medications are prepared, their efficiency must be assessed and potential undesirable effects must be traced, before these medications are widely used for disease treatment. The process of evaluating medication efficiency and detecting possible undesirable effects is called a *clinical trial*. A clinical trial involves various phases:

- The *trial planning phase*, which is responsible for initialising the clinical trial, and it is followed by the trial preparation and the trial-reporting phase.
- The *trial preparation phase*, which includes the preparation of the products, the budget specification, the selection of the clinical centres where patients will be monitored, the finalisation of contractual documents as for instance the observation notebooks (documents used to record all information related to patients), and so on.
- The *monitoring of the trial* is composed of patients' visits to the centres and their investigators, and the monitoring of the patients and the observation notebooks.
- The *trial data management phase* is responsible for the management of data collected from the clinical trial: data is retrieved from the observation notebooks and its coherency is verified, in order to prepare a "clean" database for the evaluation and reporting of the trial.

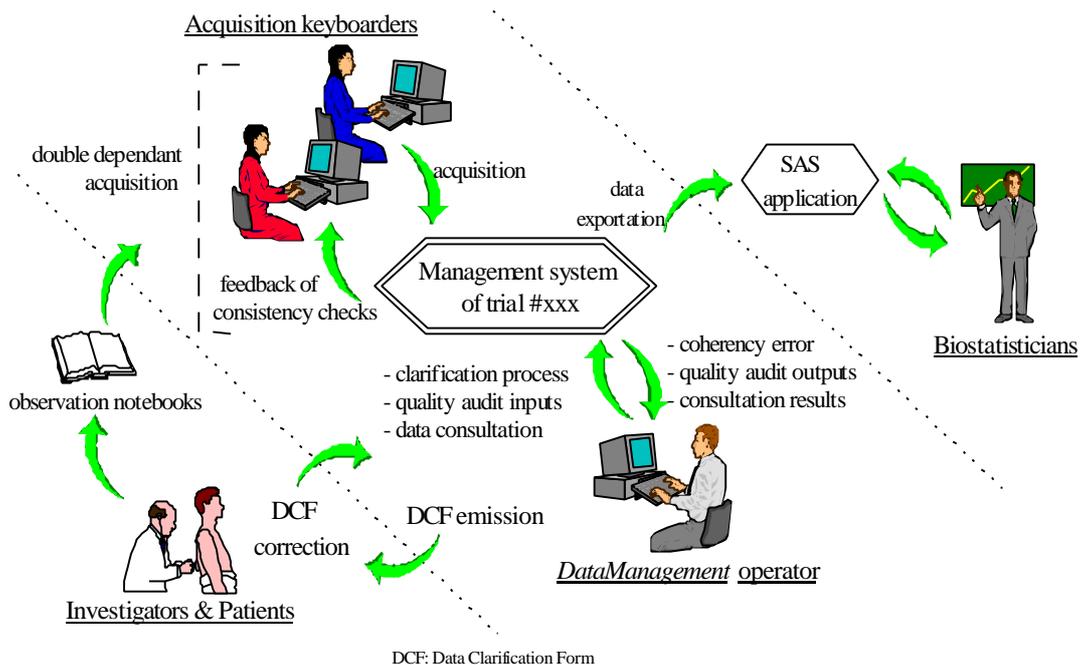


Figure 16 – Overall functionality of the clinical trial management application.

In the context of the TOOBIS project, a *clinical trial management application* has been developed, which is used during the trial data management phase. The functionality of this application, illustrated in Figure 16. The first task performed is the acquisition of the information collected in the clinical centres. This information has been written on the observation notebooks by the investigators and/or the patients. In order to store the information in the data management system, and to minimise the acquisition errors, two different keyboarders perform a double-dependent data entry.

The second keyboarder performs his task with no knowledge of the values entered by the first keyboarder, but if the system detects that the value (s)he is entering is different from the one entered by the first keyboarder, then (s)he has to choose which of the two values is correct.

When some data have been stored into the system, the data clarification process may start. Data management operators launch coherency checks. When incoherence errors are detected, *Data Clarification Forms* –DCFs– are emitted by the system. These DCFs are sent to the investigator responsible for the patient for whom the error has been detected, in order to resolve this incoherence. Subsequently, the corrected DCFs are returned to the system, and the information in the database is corrected, accordingly. When all the data have been acquired, and no more DCFs are emitted, then the trial database is declared *clean*.

Some data management operators are also responsible of performing quality audits on the system regarding the inconsistency and incoherence errors detected, in order to improve the clinical trial management system by minimising such errors in subsequent trials.

The biostatistic analysis and the trial evaluation tasks are performed using different software systems, as for instance the SAS software, which provides statistical tools; the data required for the different analyses are retrieved from the data management system.

The *clinical trial management application* presents temporal requirements on both time dimensions:

- The *transaction time dimension* is used to keep track of all database modifications of the same entity. For example, erroneous values corrected in the double acquisition procedure are stored as values with past transaction time timestamps. The transaction time dimension is mainly used for quality and audit control report generation, e.g. “*how many errors were detected in the acquisition phase*”.
- The *valid time dimension* is used to maintain the history of the changes of a patient status during a trial. The valid time dimension is timestamped either using *instants*, when the corresponding information is instantaneous (e.g. patient temperature measured at some time point), or using *periods*, when the corresponding information is continuous (for example, the hospitalisation period for a patient). The valid time dimension is used when reasoning on the evolution of patient’s state, e.g. “*How long does it take to obtain a 10% increase of the MBEV³, regarding theoretical values, since the beginning of the treatment?*”.

The *clinical trial management application* has been developed on the Windows NT platform, implemented using the C++ TOOBIS bindings and libraries for database interaction, and MS-Windows visual tools for user interface development. Sample interaction screens from the clinical trial management application are presented in Figure 17 and Figure 18.

³ MBEV: *Maximum Blown Expiratory Volume*, a measure used in clinical trials for asthma medication.

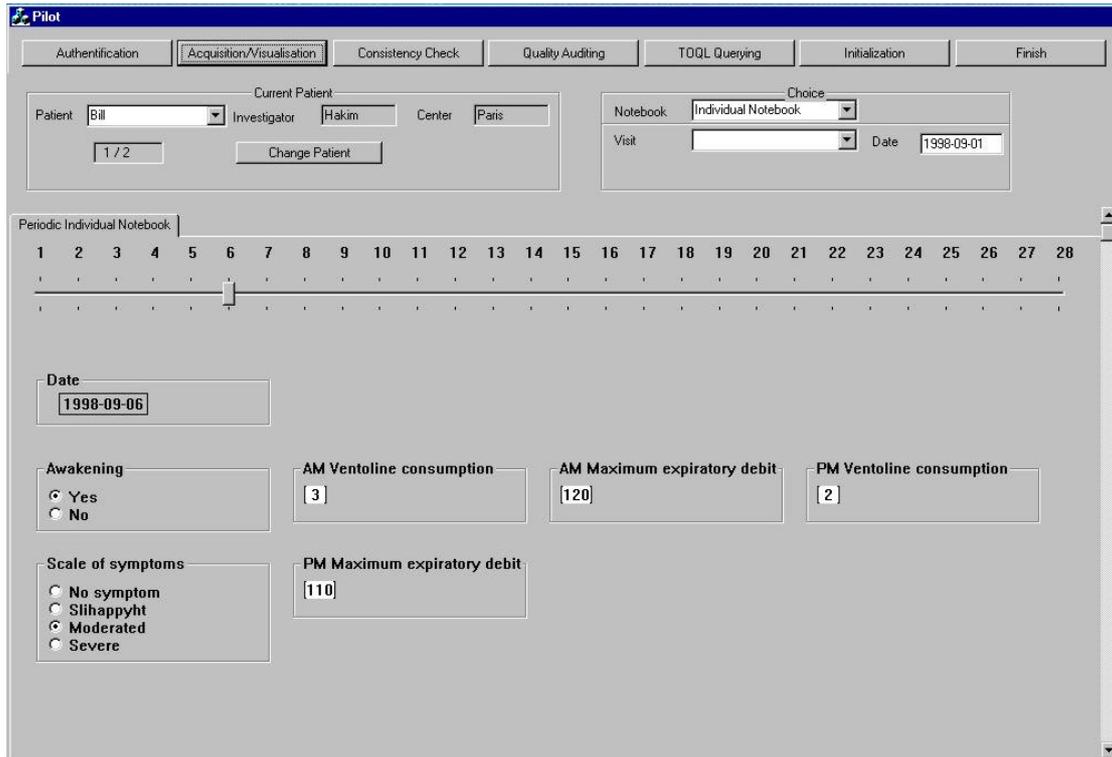


Figure 17 - Interaction screen for data acquisition

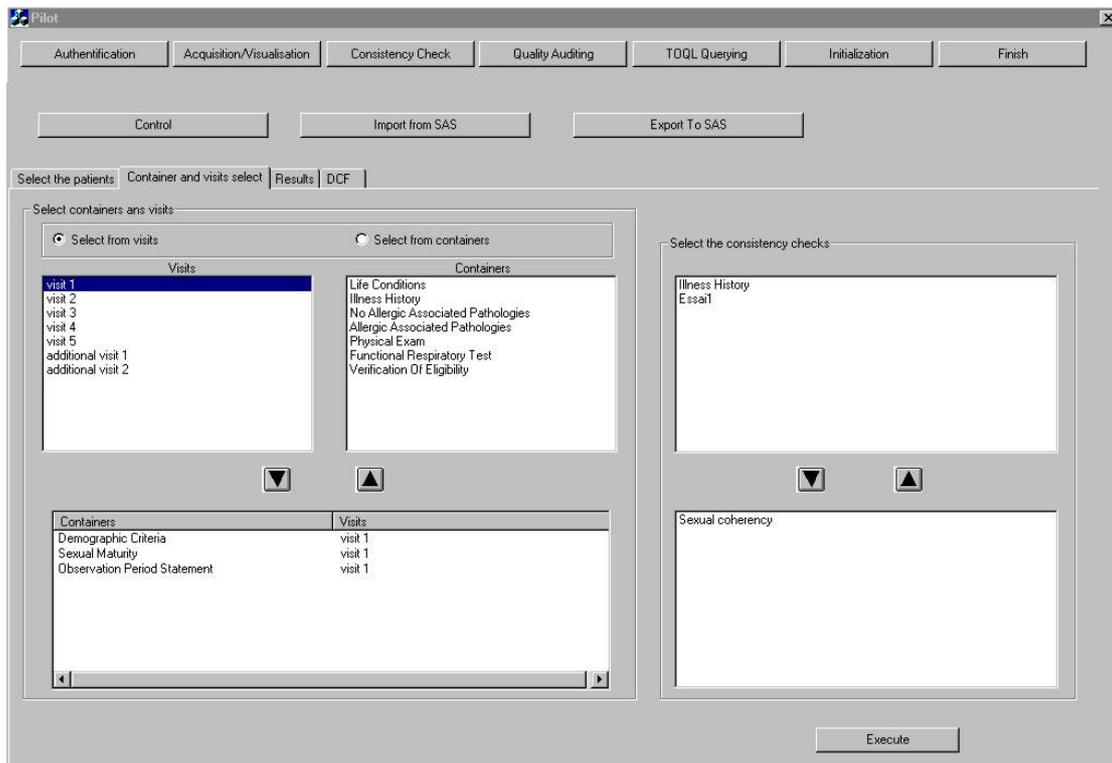


Figure 18 - Interaction screen for data quality control

Since most clinical trials follow the same pattern of work and considerable parts of the information collected in them is common, a *meta-application* has been developed in the context of the TOOBIS project, in order to further facilitate clinical trial

management. The *meta-application* enables the user to describe various parameters of the application, such as the duration of patient monitoring, the information collected at each patient examination, etc. Once the clinical trial is adequately described, the *meta-application* automatically generates definitions for database objects which will be used for the management of the data from the specific trial. Standard quality control reports and audit reports are also automatically generated by the *meta-application*. In this sense, the *meta-application* speeds up considerably the process of developing applications specific to each trial. A sample interaction screen of the *meta-application* is presented in Figure 19.

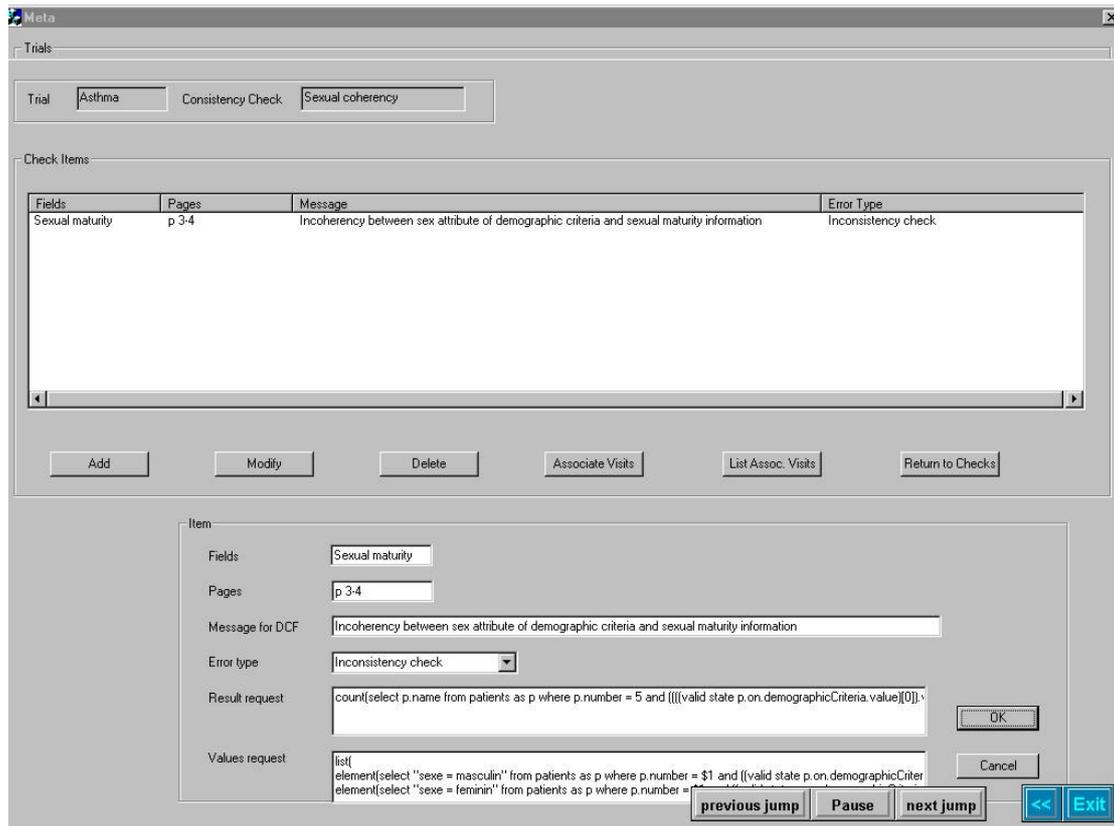


Figure 19 – Interaction screen for the meta-application

7. Future Exploitation

7.1. 01 Pliroforiki

01 PLIROFORIKI (01P) is an IT company and is very active in Greek Public and Private sector offering IT consultancy representing 25% of its activities, training representing 40% of its activities and systems integration and development representing 35% of its activities. The experience gained in this project will be transferred to all the above mentioned company activities.

The exploitation of the TOOBIS results can be projected in the following directions:

7.1.1 TOOM Methodology for Consultancy, Training and Development activities

01P has sound experience in developing integrated information systems providing customised solutions for large organisations of the Greek Public and Private Sector. Currently, the SW development activities of 01P for commercial projects are being carried upon a technological platform involving the use of a Relational DBMS, mainly ORACLE and SQLServer.

It is not intended that the technologies used will be changed. What will be changed will be the methodology and the development life cycle. The methodology being developed in TOOBIS is based in UML. 01P is using the UML-compliant SELECT-PERSPECTIVE, for medium S/W development projects for the private sector. The results of the application of TOOM methodology seem very promising. Already 01P with University of Sorbonne and University of Athens has packaged a portfolio of training services. The response to the seminars provided seems very promising, receiving positive comments from the audience.

7.1.2 Pilot application development for customised applications

The Pilot application of DELTA integrates many MIS elements. Practically, the DELTA pilot is a solution that can be considered to reside in the domain of OLAP and data mining applications. 01 Pliroforiki, as a consultant and customised software developer, sees added value in this area, since all data warehouse applications have clear requirements of time management.

The banking sector and financial applications seem to be an ideal domain for TOOBIS exploitation. In addition, the area of health is an area where time management is of high importance. 01P has started marketing poll activities in this area. The actions in this domain will be co-ordinated with MS&I and Ardent Software.

The area of OLAP and DSS is directly within the interests of 01P, as it foresees that the market shifts towards this category of applications i.e. to analytical processing applications. 01P foresees that the future On-line Analytical Processing applications based on data warehousing and data mining approaches will create great opportunities in the market. To this end, the company tries to find technical and organisational means to exploit the results of TOOBIS in the area of customised application development for analytical applications. The pilot application of DELTA may be positioned in the class of market needs concerning data warehouse information integration of large Enterprises to performing Decision Support activities. Table 4

shows the market Growth in the Data Warehouse sector. It can be seen that the services area is rapidly growing, and this is of strategic importance to 01P.

The experience gained by implementing the Pilot of DELTA will be transferred to other business sectors as well as within the same sector of distribution and product deliveries. There are numerous similar situations where an organisation needs to efficiently integrate, consolidate and extract valuable information currently dispersed in legacy information systems; namely, banks requiring to monitor the behaviour of their clientele, large distribution and sales organisations, public organisations etc.

	1994	1999	Compound Annual Growth Rate
TOTAL MARKET SIZE	\$1,568	\$6,969	34,7%
Data Extraction/Movement	\$65	\$210	26,4%
Administration	\$10	\$450	114,1%
RDBMS	\$288	\$1,100	30,7%
Hardware	\$1,075	\$3,950	29,7%
Consulting Services	\$130	\$1,250	57,3%

Table 4 - Data Warehouse Market Segment Revenue Forecast, Source: Gartner Group Inc., Prices in Million US\$

01P has a project with National Bank of Greece in the area of Data Warehouse and Decision Support applications. In this sense 01P sees the DSS applications as an important class of applications contributing to a significant amount to its software consulting and development activities. Synergy in this sense was applied between TOOBIS and Hyperbank, a project in the area of High Performance Networking and Computing in the 4th Framework that started in January 1997. The knowledge gained by the Hyperbank participation in the area of data warehouse and data mining have had impact on the development of TOOBIS. In addition, 01P applied the consolidated knowledge on building analytical applications in the area of Health in a significant project that started March 1998, in the area of health telematics. This project maintains Cardiac Patient historical data, and uses this data in analyses to support treatment and health strategies.

7.1.3 Object Technology

01P is convinced that the future is with object technologies. There is a core team of experts developed in 01P to be specialising for Internet / Intranet development with Object Technologies. To this end, and in this strategy, the following actions have been taken:

- Strategic affiliations with Case / Methodology vendors (SELECT UK). 01P has signed a team agreement with SELECT UK involving the use of the Select Enterprise methodology for systems' implementation to consultancy and training activities.

- 01P has signed an agreement with Ardent Software Inc., for distributing the O₂ Database in Greece. This collaboration will transform the experience gained by TOOBIS in this area.
- Gaining of expertise with CORBA, and CORBA-O2 binding. Corba is a developing distributed solutions standard.
- Gaining of expertise with Java and O2/Java binding. Java is spreading in Intranet development and a standard for GUI development delivery.
- Introducing the Microsoft approach to designing and delivering distributed applications with OLE/COM, with Microsoft Transaction Server offering transaction support for distributed business objects for the provision of middleware solution.
- Gaining of expertise with TOOM, to be used in training and consultancy for capturing the temporal requirements of applications.

To this end 01P transforms the potential gained within TOOBIS to the pursue of commercial consultancy activities to the Greek market in the OO area, focusing in Internet development. The distributed business objects' dimension is of high interest, and 01P investigates the possibility to consider the base developed within TOOBIS as a source to delivering distributed time objects.

7.1.4 The TOOBIS platform

It is expected that the involvement of 01P in TOOBIS will result in substantial benefits. It is anticipated that both partners, 01P and MS&I, leading the TOODMBS development effort, could develop a commercial product after the end of the project. This product would be an add-on library of system software modules to O₂ OODBMS and will be distributed by Ardent Software Inc. as a separate module for customers interested in temporal support.

The approach to developing the temporal extensions has been open and based on standards (ODMG). The benefit of this approach is that the TOOBIS extensions may be ported to other OODBMS platforms such as the POET™ database. A possibility to offer distributed solutions based on standard distribution approaches such as CORBA is being investigated. All the above are seen under the scope of different packaging possibilities which would facilitate a more flexible exploitation path.

7.2. *Matra Systemes et Informations*

7.2.1 Interest of TOOBIS in MS&I activities

Matra Systemes & Information is a leading company in the realisation of Information Systems in the areas of C³I (Command, Control Communication and Intelligence) and spaceborne/airborne observation systems. Those systems are designed to assist users in their mission during operations, offering them a broad range of services: information acquisition and transmission, decision support, simulation and evaluation, operations command and control.

Defence is one major application domain of C³I systems, but operational requirements of many non-defence applications and systems are very similar to those of defence domain. Additionally, in all these systems, requirement on information retrieval and

processing related to time are of a very high importance. In TOOBIS, MS&I brings its technological expertise, gained from the development of military applications. In return, it will use the results of TOOBIS in temporal databases, as a baseline to develop applications in several civilian sectors. MS&I expects to transfer this expertise to application area such as crisis management, humanitarian mission planning and control, mobilisation of aero-transportable equipment for inspection, communication and command, systems for the regulation of traffic in intra urban areas, fleet monitoring and police information systems.

Due to the strong time requirements in C³I domain, MS&I considers the future results of TOOBIS project (generic software packages to manage both valid and transaction time in Information Systems, and a methodology to design temporal applications) as a key point for many future systems developed by MS&I.

So, at the end of the project, the results of TOOBIS will be exploited in the best way for new developments of such systems. MS&I expects that an advanced solution for time management will allow both easier application development with reduced costs, and the development of more powerful functionalities. These results would have been difficult and expensive to implement with ad-hoc solutions.

Moreover, it is expected that the TOOBIS solution will be helpful to enter new markets related to civilian C³I or similar Information Systems on one hand, and related to data warehouse in the other hand.

These considerations lead to an exploitation plan relying on two main axes:

- exploitation of TOOBIS results for future systems that will be developed in the present market of MS&I
- exploitation of TOOBIS results to enter new civilian markets related to MS&I activity.

For the first axis, the domain concerned is defence and earth observation, where target systems are well identified, due to the strong involvement of MS&I in this domain. MS&I is going to be the software house of the future Aerospatiale/Matra merged company, so this area will also include aeronautics.

Regarding the second axis, the domains of potential interest initially identified were the three following ones:

- civilian C³I, or similar systems domain, including systems such as systems to assist crisis management, systems for humanitarian interventions, information systems for logistics, fleet management, and management of production.
- health domain, including information systems for clinical trial management systems and health systems (networked systems connecting liberal doctors, hospital and health administration, and aiming to assist both health authorities and doctors in the optimisation of quality and costs in the healthcare process),
- information systems for insurance company, banking and financial sectors.

7.2.2 Market analysis

Actions including commercial staff of MS&I have been performed during the project to get a more accurate view of these domains. In this framework, a market analysis has been performed. The main points of this market study are presented in an annex.

The results of the market analysis indicate that:

- the domain of civilian C³I initially identified was too large. It have been split in more specific domains, some of which are not interesting for MS&I. In particular, humanitarian C³I and crisis management areas have been removed from the exploitation axis, because their market is too small,
- the domains of Banking and Insurance previously identified as a domain of potential interest have been removed, because MS&I is very loosely related to this domain,
- police information systems appear as a domain of interest.

Considering all factors, the following domains are emerging from the market study:

- Health
- Defence and Earth observation
- Logistics and distribution
- Institutional Systems (including police Information Systems)
- Territory management

For the exploitation of these domains, the results of TOOBIS are expected to be a valuable competitive advantage.

7.2.3 MS&I offer

The offer may be a product, a service (facilitated by a competitive background knowledge or based on an "internal" software tool), or a complete system (in which a temporal component brings a competitive technology for a less expensive development or for the satisfaction of an important requirement).

The possible technological components on which a MS&I proposal may be based result from the outputs of the project:

1. the complete TOODBMS. It is composed by TODL, TOQL and TODM based on O2 or another ODMG-compliant OODBMS. This requires the industrialisation of the TOODBMS
2. the TODM component may be used :
 - either with O2 (or with another ODMG-compliant OODBMS), which requires an easier industrialisation than a TOODBMS.
 - or with an Object-Relational DBMS (e.g. Oracle V8), requiring a complete or partial porting on another technological background. Such a porting has been proven to be feasible
3. the TOOM methodology and/or the background knowledge in time management mechanisms, for Information System development, used in conjunction with "classical" of development and software (off-the-shelf DBMS products)
4. the Clinical Trial Management application, that may be seen:
 - as a generic software package (that required an industrialisation of the pilot) (4a).

- as a technological background for the development of other applications in health domain (4b).

The axis of development of MS&I for the chosen domain are based on the following items:

- (2) or (3): systems for Defence and Space areas, Farming and Food industry,
- (3): Logistics, including Port Communication and Information Systems,
- (4b): Health domain: Clinical Trial Management Systems, and other Health Information Systems, Disease Management Systems.

7.2.4 Expected MS&I business

The expected businesses in Information System with partial contribution of the TOOBIS technology are presented in Table 5 and Table 6 in the Annex.

7.3. Ardent Software Inc.

In TOOBIS, Ardent Software is interested to market the framework resulting from TOOBIS and to be reseller of the products defined in the project in compliance with the agreement established between the industrial partners of the TOOBIS project.

Positioning - The O₂ DBMS Technology faces two different kinds of competitors: relational vendors and object oriented vendors. Against the relational vendors, our advantage is the excellence of the technology that is the best to deal with structured data, such as temporal data. Although relational vendors have a strong commercial presence, a significant share of the MIS market is captured by object-oriented vendors. The new Object/Relational Mappings enforce our presence in the relational market.

Currently, no object-oriented vendors are shipping temporal extensions. The O₂ DBMS Technology is the European market leader and ahead concerning ODMG compliance which is the key component for the perennality of applications.

Dissemination plan – it includes:

- Distributing marketing material by providing the TOOBIS CD-ROM Demonstration to the Universities.
- Providing specific license contracts for demonstration purpose for all the TOOBIS end-users as decided at the last Consortium Meeting. This contract provides an O₂ license free of charge for a period of 2 years. It includes the delivery of all O₂ releases and documentation set needed to prepare to the deployment phase of the applications.
- Participating in the International Conference OOIS'98 on the 9-11 Sept. 1998. Ardent was part of the TOOBIS Tutorial and presented the ODMG Standard.

Ardent and TODL – Ardent Software is interested in the TODL product resulting from TOOBIS and in compliance with the agreement previously established between the industrial partners of the TOOBIS project established a specific agreement with the University of Athens.

7.4. University of Sorbonne

The University of Paris I-Sorbonne intends to disseminate the results achieved in the project through teaching undergraduate and postgraduate students. In addition, University of Sorbonne has a doctoral curriculum entitled "Theorie et Ingenierie des Bases de Donnees" (Database Theory and Engineering) which is focusing on the various aspects of databases and DBMSs and will be an excellent vehicle for dissemination of the project results.

The Department of Mathematics and Informatics of the University of Sorbonne maintains close links with the French Society on Computer Science (AFCET). This connection will serve as a basis for transferring technological expertise to French organisations. Complementary, the Department maintains relationships with companies and administrations through consultancy, training courses and co-operative research and development projects sponsored by the French IT Ministry. The experience gained during the project and the achieved results will be transferred through this network of industrial connections.

Dissemination Plan - The dissemination actions of University of Sorbonne are seen in the following directions:

- Publications on the methodology.
- Integration of special lectures in temporal databases in DEA "Theorie et Ingenierie des Bases de donnees" and DESS "Systemes d' Information" available in University of Sorbonne have been done.

Temporal databases and temporal methodologies supporting the development of temporal database applications are not fashionable topics. They are not easily accepted now in conferences dealing with databases and/or information systems development, even if the time concept is handled everywhere. Our research works in the temporal aspects have been re-organised in two directions:

- exploration of "best practices" guidelines to introduce time specific concepts into a method,
- identification of patterns specific to temporal applications.

7.5. University of Athens

University of Athens has already collaborated with Ardent Software Inc. for the implementation of the ODL parser for the next version of the O₂ ODBMS. The first version of the parser is available for Ardent Software, generating C++ classes from ODL statements. The second version generating Java classes from ODL statements will soon be available. Ardent Software has already provided a permanent O₂ license to University of Athens.

With the expertise gained from the TOOBIS project, research on the temporal database area is going to continue. Research activities include PhDs, new proposals for European and National projects, postgraduate and undergraduate diploma theses and publications to conferences and journals.

The University of Athens (UoA) is interested in the research activities that will take place within the project, as these activities address current open research issues. The main objective of the UoA is to propose solutions to open issues and publish papers in

journals and conferences. The results of the project will also be used as teaching material in academic courses.

7.6. Delta Dairy S.A.

DELTA is the largest producer of fresh-products in Greece. It has a large distribution channel consisting of hundreds of representatives, refrigerated vehicles and sales supervisors, more than 40.000 retailers and more than 50.000 display refrigerators or freezers. DELTA is presented with the following problem to be tackled:

DELTA has invested more than 6.6 MECU in Information Technologies since 1992. It has invested in communication infrastructure and equipment as well as state-of-the-art software. The company has already begun to realise a return on this investment and it plans to continue to invest in Information Technology and innovative IT solutions.

DELTA expects to acquire extensive experience from its participation in the project and define the requirements of its important application in detail. It will, subsequently, use the expertise gained from TOOBIS to develop, together with O1-P, a complete application after the end of the project.

7.7. GlaxoWellcome

The pharmaceutical industry is one of the major components of the health care system. As healthcare actors, pharmaceutical companies have to deal with huge quantities of medical data, either research data or clinical data. Most of them are highly time dependent and cannot be interpreted without any consideration with this element.

Roughly potential fields of usage for a temporal database management system could be:

1. Management of data from clinical trials at different phases of the clinical studies.
2. Pharmaco-economic studies, where a great number of patients will be investigated during a long time, in order to assess the pharmaco-economic benefits of different drug protocols.
3. Epidemiology.
4. Tracking of adverse events.
5. Studies where long term results of various therapeutic regimes will be compared on the basis of their medical effects or on the basis of patients' quality of life.

All those domains (the list is not intended to be exhaustive) could benefit from the availability of database management systems with advanced features. GlaxoWellcome intends, following the validation of the end products of TOOBIS, to develop a fully operational application for medical diagnosis of chronic diseases. This will take place in collaboration with MS&I. It should be a product, that could not only be used internally by MS&I, but also, it could be purchased by other interested parties, such as pharmaceutical companies, insurance companies, health centres and medical institutions. Additionally, GlaxoWellcome intends to investigate the rest of the application domains described above.

8. References

- [Brinkkemper90] S. Brinkkemper, *Formalisation of Information Systems Modelling*, PhD Thesis, Thesis Publishers, Amsterdam, University of Nijmegen, 1990.
- [Brunet93] J. Brunet, *Analyse Conceptuelle orientee-objet*, PhD Thesis, University of Paris 6, 1993.
- [Catell93] Catell, R.G.G, et al. "ODMG-93". International Thomson Publishing.
- [Coad90] Y. E. Coad P., *Object Oriented Analysis*, Prentice Hall, Englewoods Cliffs, New Jersey, 1990.
- [Coad91] Y. E. Coad P., *Object Oriented Design*, Prentice Hall, Englewood Cliffs, New Jersey, 1991.
- [Griethuysen82] J. J. V. Griethuysen, *Concepts and Terminology for the conceptual Schema and the Information Base*, ISO TC97/SC5/WG3, 1982.
- [Hofstede90] T. F. V. Hofstede A.H.M., G.M. Wijers, S. Brinkkemper, *The SOCRATES project*, proc. of NGCT'90, (Noordwijkerhout, The Netherlands), pp. X1-1-X1-7, 1990 (April).
- [Jacobson93] I. Jacobson, M. Christerson, P. Jonsson, G. Overgaard, *Le genie logiciel oriente-objet, une approche fondee sur les cas d'utilisation*, A.-W. France, Eds., Bordas, 1993.
- [Jones80] Jones, S., Mason, P.J. "Handling the time dimension in a database". Proc. Int. Conf. on "Databases", The BCS, Univ. of Aberdeen, 1980.
- [Lee94] S. P. Lee, *Formalisation et aide outillee a la modelisation conceptuelle*, PhD Thesis, Paris 1, 1994.
- [Lyytinen89] S. K. Lyytinen K., Tahvainen V-P., *Modelling CASE Environments in systems Work*, CASE'89 conference papers, (Kista, Sweden), 1989.
- [Martiin94] P. Martiin, *Methodology Engineering in CASE shells: Design Issue and current Practice*, Computer science and information systems reports PhD thesis, Technical report TR-4, 1994.
- [Martin92] J. Martin, J. J. Odell, *Object-Oriented Analysis and Design*, P.-H. I. Editions, Eds., 1992.
- [ODE] Gehani N., Jagadish H.V., *Ode as an active database: constraints and triggers*, Proc. of the 17th VLDB, Barcelona, Spain, 91, pp. 327-336
- [ODMG97] The Object Database Standard: "OQL v.1.2" available from <http://www.odmg.com/>, 1997.
- [Olle91] H. J. Olle W., Macdonald I.G., Rolland C., Sol H.G. Assche F.J.M.V., Verrijn-Stuart A.A., *Information Systems Methodologies*, Addison-Wesley, 1991.

- [Rolland82] Rolland C., Richard C.: *The REMORA methodology for Information systems Design and management*, Information Systems Design Methodologies: a comparative Review, Olle T.W, Sol H.G., Verrijn-Stuart A.A. (eds), North-Holland (pub), 1982.
- [Rolland96] Rolland C., *Using Generic Method Chunks to Generate Process Models Fragments*, 3rd Int. Conf. on Requirements Engineering (ICRE'96), 1996.
- [Rumbaugh91] J. Rumbaugh, M. Blaha, W. Premerlani, F.Eddy, W. Lorensen, *Object-Oriented Modelling and Design*, P.-H. I. Editions, Eds., 1991.
- [Shlaer88] M. S. J. Shlaer S., Eds., *Object Oriented Systems Analysis*, Yourdon Press, 1988.
- [Shlaer92] S. Shlaer, S. J. Mellor, *Object lifecycles. Modelling the world in states*, Y. Press, Eds., 1992.
- [Snodgrass92] Snodgrass, R.T., Soo, M.D., “Multiple Calendar Support for Conventional Database Management Systems”. Dept. of Computer Science, University of Arizona, TR92-07, 1992.
- [Snodgrass96a] Snodgrass, R.T. “A Road Map of Additions to SQL/Temporal”. ISO ANSI X3H2-96-013, 1996.
- [Snodgrass96b] Snodgrass, R.T., et al. “Adding Valid Time to SQL/Temporal”. ISO/IEC JTC1/SC21/WG3 DBL MCI-142, ANSI X3H2-96-151r1, 1996.
- [Snodgrass96c] Snodgrass, R.T., et al. “Adding Transaction Time to SQL/Temporal”. ISO/IEC JTC1/SC21/WG3 DBL MCI-143, ANSI X3H2-96-152r, 1996 (b).
- [Tansel93] Tansel, A.U., et al. “Temporal Databases: Theory, Design and Implementation”. The Benjamin/Cummings, 1993.
- [Toobis96a] Deliverable «User Requirements», TOOBIS project, June 1996.
- [TSQL2/95] TSQL2 Language Design Committee. “TSQL2” edited by R.T.Snodgrass. Kluwer Publishers. Sept. 1995.
- [UoA96] University of Athens et al., “TOQL Specification”, TOOBIS Project Deliverable T33TR.1, 1996. Available through <http://www.di.uoa.gr/~toobis>.
- [Wijers91] G. M. Wijers, *Modelling Support in Information Systems Development*, PhD Thesis, PhD thesis, 1991.
- [Wynekoop93] R. N. Wynekoop J., *System Development Methodologies: Unanswered Questions and Research-Practice Gap*, 14th Int. Conf. Information Systems, (Orlando), pp. 181-190, 1993.

Annex: Market Analysis

This section presents some points of the market analysis that have been performed by MS&I in the framework of the TOOBIS Exploitation.

Defining the domains in the market of time management

The first step was to determine the economic domains in which there may exist a market for MS&I, related to time management facilities. This notion of «market» is here very general and may be based on offers relative to products, services or systems.

The following table shows a synthesis of the considered domains with their main characteristics.

Domain	Time management need	MS&I interest / position	Comments
Health	*****	****	<i>Interesting domain because (existence of a volume, existence of users needs, financially solvent)</i>
Defence, Earth observation, Aeronautics	***	***	
Logistics and distribution	****	***	
Administration, public services	***	***	<i>Includes police Information Systems</i>
Territory management & planning	****	***	
Fertilisers & food industry	*****	**	
Press and Media	****	**	<i>Includes as example opinion poll institutes, marketing, advertising agencies, geomarketing</i>
Civilian security	***	**	<i>Needs but no established market</i>
Humanitarian systems	**	*	<i>Strong needs for time management, but no actual market</i>
Banking, Insurance	*****	**	<i>The importance is ** only because MS&I is not in this business</i>
Transport	****	**	
Petroleum, mining, exploration	****	**	<i>This domain has not many actors</i>
Engineering	***	*	
Building	**	*	

Table 5 - Importance of Time Management by economic domain.

The table indicates:

- the economic domains in which time management may have an interest from our point of view,
- the absolute importance of the need for time management (between 1 and 5 stars),
- the relative importance related to MS&I (also between 1 and 5 stars).

The absolute importance results from the aggregation of three factors:

- V: the volume of the potential market (that results from the number of economic actors, or of the number of potential users, or potential systems.),
- N: the users needs, that determines the strength of time management requirements in the domain,
- S: the global financial solvability of the potential clients in the domain, that reflects the capacity of the domain to generate a real market

The relative importance related to MS&I results from the absolute importance corrected with quantitative factors as:

- "barriers", i.e. factors that make difficult the access to the identified market. The accessibility to a given domain results from many factors such as knowledge of this domain, existence of connections with its main actors, past industrial references (public image of the company), channel of information, etc.
- competitive advantages, that contributes to facilitate the entrance into the market.

The following table indicates the sub-domains of more specific interest.

Domains	Sub-domains
Health	Clinical trial management systems, disease management systems, hospital information systems.
Defence, Earth observation, Aeronautics	C ³ I, Intelligence, spatial and aerial observation.
Logistics and distribution	Mail, transports.
Administration, public services	Police information systems.
Territory management and planning	Territorial collectivities, cities, regions, water agencies, equipment services, biodiversity management.
Fertiliser and Food industry	Drinks, milk products, vegetables & fruits, meat.
Press and Media	Opinion polling institutes, marketing, advertising agencies, geomarketing, audience metrics.
Civilian security	Systems for the management of crisis (nuclear, Seveso, etc).
Humanitarian systems	
Banking, Insurance	
Transport	
Petroleum, mining, exploration	
Engineering	
Building	

Table 6 - Sub-domains by domain identified for Time Management.

Project Roster

Zissis Palaskas (Project Manager)	01P
Dimos Kontoravdis	01P
Nikos Chrysochoidis	01P
Thanassis Kapenis	01P
Michalis Soukakos	01P
Thanassis Loutsos	01P
Nicky Dimitropoulou (Admin. Support)	01P
Robert Havas	MS&I
Yann Pollet	MS&I
Michael Souillard	MS&I
Nadine Cornec	MS&I
Didier Plateau	Ardent/O ₂
Carmelo Malta	Ardent/O ₂
Catherine Lanquette	Ardent/O ₂
Vineeta Darnis	Ardent/O ₂
Chrisanthos Pavlou	DELTA
Hakim Zedghib	GlaxoWellcome
Collette Rolland	SORB
Carine Souveyet	SORB
Rebecca Deneckere	SORB
Christophe Gnaho	SORB
Adolphe Benjamin	SORB
Farida Semmack	SORB
Panagiotis Georgiadis	UoA
Costas Vassilakis	UoA
Anya Sotiropoulou	UoA
Costas Boukouvalas	UoA
Giannis Akritidis	UoA
Timos Sellis	NTUA (subcontractor to UoA)
Panagiotis Vassiliadis	NTUA (subcontractor to UoA)