CONCEPTUAL UNIVERSAL DATABASE LANGUAGE (CUDL) AND ENTERPRISE MEDICAL INFORMATION SYSTEMS

Nikitas N. Karanikolas, Christos Skourlas
Technological Educational Institution of Athens
Ag. Spyridonos Str., 12210, Athens, Greece
nnk@teiath.gr, cskourlas@teiath.gr

Maria Nitsiou, Emmanuel J. Yannakoudakis
Athens University of Economics and Business
76, Patission Str., 10434, Athens, Greece
mbnit@aueb.gr, eyan@aueb.gr

Keywords: Hospital Information Systems, Electronic Patient Records, Dynamic databases' schema evolution, correlating heterogeneous data.

Abstract: Today, there is an increased trend for Electronic Patient Records (EPR) incorporating and correlating heterogeneous information imported from various sources and from different medical applications. New possibilities are also given by the rapid technological progress and the development of independent software applications and tools that handle multimedia medical data. Moreover, users (e.g. doctors, nurses) often prefer to use general purpose software (e.g. word processors) and specific applications and tools for organizing and accessing medical data and they only partially use Hospital Information Systems (HIS). Therefore, it is necessary for the HIS to provide the capability for encapsulating in their EPR externally created information. Dynamic evolution of the HIS must also be supported by flexible Database schemata. In this paper, we conclude that modern HIS should be designed and implemented using database management systems offering new enhanced database models and manipulation languages. Eventually, we describe and discuss how to use the Frame Database Model (FDB) and the new Conceptual Universal Database Language (CUDL) for supporting dynamic schema evolution and covering the needs of the users.

1 INTRODUCTION AND PROBLEM'S OUTLINE

Recent years information related to patients increases continuously and the quality of offered services in the modern hospitals' environment is susceptible to drastic improvements following the development of science and technology. This fact leads to the adoption and application of information technology based solutions in order to record and process effectively data that emanates from the medical as well as from the administrative-economic operations of the nursing organism. The improvement of provided services of care and the tracking of cost constitute two of the most important reasons for the import and exploitation of information technology in the hospital organism. The Hospital Information Systems (HIS) are constituted by several applications and interconnected sub systems. The Information that a hospital should have in its disposal, with regard to the medical incidents, is based on an enormous size of data that should be combined suitably.

Karanikolas and Skourlas (Karanikolas et al, 2000, Karanikolas et al, 2002) have studied for five years (1997-2001) the operation of Information Systems at University Hospitals in Greece and their conclusions can be summarized as follows:

Certain divisions of the Hospital have purchased and installed in different time periods distinct, operational, "legacy" systems covering their needs, e.g. a Hospital Information System (HIS) and a separate Laboratory Information System (LIS). There are various problems of interconnection between the different systems and their integration, for value added services, is a very complicated task. Clinical departments have difficulties using the complex and text only based electronic patient record of the legacy systems. Hospital units also have to operate and maintain local sources of information apart from the database(s) of the HIS. Doctors have only partially used the existing complicated HIS all these years. They prefer to use general purpose software for writing diagnosis notes and discharge letters, and they also have a rigid trend for storing, retrieving and handling medical images from various sources, e.g. MRI, CT and Ultrasound images.

Based on the above conclusions, in this paper we propose the exploitation of a new database model (the Frame Data Base model - FDB), a new database language (Conceptual Universal Database Language - CUDL) and the CAIRN architecture (Karanikolas et al, 2000, Karanikolas, 2007). We use them for the development of Enterprise Medical Information Systems (EMIS) that integrate selective data from legacy systems, hierarchically structured data and further information (e.g. external files or other resources). Such information could include medical images and notes related to the diagnosis and the therapy for specific cases, lists of experts etc. The main advantage of this approach is that the doctors continue to use their favourite software (e.g. word processor for their notes) and store various documents and statistics in their favourite system. Hence, they contribute information in the enterprise system using a document management module (with simple interface) which interferes automatically. Hence, doctors can work their way and the medical files they collect and produce are correlated and eventually form personalized electronic patients' files. Moreover, we propose the integration, into the enterprise system, of a text data-mining module (Karanikolas et al, 2005, Hearst, 1999). Such a module can offer great assistance to the doctor. For example, text mining can permit the automatic or semi-automatic selection of ICD Diagnosis codes from textual data (specifically from patient discharge letters coming from either external files or legacy systems).

2 MATERIALS AND METHODS

2.1 The Frame Database Model

Yannakoudakis et al, 1999, investigated dynamically evolving database environments and corresponding schemata, allowing storage and manipulation of variable length data, a variable number of fields per record, variable length records, fields that accept repetitions, composite fields (attributes having subfields) and manipulation of authority records. A new framework called FDB - Frame DataBase - was proposed and discussed for the definition of a unified schema that eliminates completely the need for reorganization at both logical and internal levels. In the FDB schema a subset of data functions as a descriptor (metadata) to determine the structure and the way of handling of all the data and this property makes easy the schema evolution.

Another approach for the support of dynamic database schema evolution, for medical data is the Entity-Attribute-Value (EAV) model (Anhøj, 2003). There are three variations of the EAV model. The latest (and the most mature) one (EAV/CR) organizes the entities into classes and consequently each class hosts the local and also the inherited attributes. The attributes of a class's instance can have more than one value (repetitions), however these repetitions are obligatory characterized by different time stamps. Moreover, there is no clear way for defining composite attributes as FDB does.

2.2 The CUDL Language

The FDB framework is complicated and the management and operation of it is difficult, laborious and time-consuming. It requires from the user a very good acquaintance of this framework and the structures and organisation of it (metadata and data). An indicative example of the FDB's complication is the need to transform (pivot) FDB data for displaying them in a more convenient layout. The CUDL language (Karanikolas et al, 2007) is tailored to undertake the management of the complicated structures of the FDB model. By the use of CUDL, access in the information of an FDB application becomes very simple with the use of particularly simple statements. End users conceive the data of the FDB system via CUDL as an extension of the relational model in the sense of organizing data into tables. Users can define and manipulate data organized in the form of tables (entities in the FDB model), rows (frames in the FDB model) and columns (tags in the FDB model). The extensions refer to the possibility that each field can have multiple values, or a cell can be a list of values. Also a cell in CUDL can have multiple subfields and a cell can also entertain a whole table. To our knowledge, there is no language defined for the EAV model, to simplify its usage.

2.3 Method of Solving

Documents stored into local databases will not simply form a flat collection of unrelated documents. As an example, a patient can have more than one episode in different periods of time. It is obvious that various documents are related to the patient's episodes, incidents, examinations, etc. Thus, there is a need to group all these related documents in a hierarchical tree structure. The system has to implement a tree structure of categories where each level has a different number of fields that characterize the category. The fields that characterize the first hierarchy level can be demographic elements of patients while the fields that characterize the second hierarchy level can be elements that describe a specific episode (e.g. incident, admission date, discharge date, final diagnosis and so on). The leaves (documents) in this hierarchy can also be characterized by a number of fields. This set of fields forms a (document) profile. Document Management and retrieval can be conducting by giving values and constraints for the various fields of the document. A feature of interest is the support of both multiple-value (multi-row) and single-value fields. Data types supported by the system can include integer and real numbers, dates, strings, lookup fields and multiple attribute (or multi-column) fields. The combination of a multiple attribute field that accepts multiple values derives the ability to use two-dimensional tables in the position of a single field. It must be possible to store structured documents (e.g. PDF and XML documents), other documents downloaded from the Internet (HTML documents), etc.

These capabilities of FDB model and CUDL language allow us to easily design and implement hierarchically structured medical data, allowing also support for future schema evolution. They also give us the opportunity to design and implement the integration structures for the correlation of heterogeneous data (for importing data either from "legacy" systems or from external files and other resources).

3 STRUCTURING HIERARCHICAL MEDICAL DATA WITH THE CUDL LANGUAGE

In this section we focus on a Hospital Information System in order to clarify the concepts described in the previous section. Information of interest could be related to episodes or incidents (admission etc.) of the patient to the hospital. For example, in each one of the incidents the patient may need the care of various doctors, and he may undergo some operations. Certain physicians can participate in the operations and certain (the same or others) attendant physicians treat the patient. During the incident, the patient may undergo Laboratorial or Radiological Examinations on the same day in different hours or in different days. In consequence the data we want the HIS to hold for the various episodes could be the following: patient's code, social security institution related information (e.g. code, name, rates for the insured patients), data related to each incident of the patient (e.g. code, admission date, discharge date), data related to the doctors involved (e.g. attendant physicians, surgeons), data for each operation that took place and the description as well as the results of the laboratorial examinations (e.g. laboratory, test, date, results) during the incident. For each patient we are also interested to keep demographic data.

Figure 1 illustrates a part of the simplified relational database schema of the integrated HIS comprising of seven discrete objects (entities): Patients, Doctors, Social Security Institutes, Laboratorial Examinations (test), Radiological Examinations (MRI, CT, ECG, etc), Diagnoses and Operations. Diamonds (acting as relationships between entities in the case of the well-known Enhanced / Entity - Relationship model (Badia, 2004) are depicted as objects: Incidents, Lab Examinations / Incident, Rad Examinations / Incident, Doctors / Incident, Operations / Incident, Doctors / Operation).

The FDB model is more compact and simpler. As we can see in the examples of the following subsection the same seven discrete entities are involved but the model comprises by only one object describing all the relationships between the entities. Therefore, the CUDL language permits the definition of more sophisticated relationships (sophisticated diamonds or CUDL diamonds). In our example, the whole set of the six relationships (see the circle in figure 1) is replaced by a single sophisticated relationship (a single CUDL diamond). The rationale for this fact is that the underlying (FDB) model can offer the possibility of having repeatable values, sub-fields and a whole table in place of a field (tag, in our terminology). Therefore, we can easily include the attendant physicians, all the laboratorial and radiological examinations that took place in an incident, all the doctors that

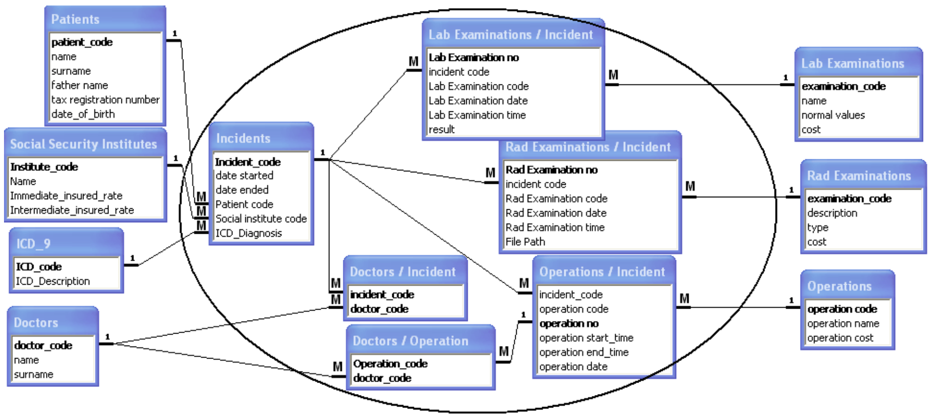

Figure 1: A part of the simplified relational database schema.

participated in each operation, etc. In order to create the schema of the FDB database, the designer provides CUDL language statements and the metadata of the schema are updated. The following is an excerpt of the CUDL data definition statements used to create an FDB schema compatible with the above relational schema:

```
# Add tag_attributes entity =
  'Incident' title = 'Laboratorial
  Examinations' repetition = 'R'
  datatype = 'compo'
# Add subfield_attributes entity =
  'Incident' tag = 'Laboratorial
  Examinations' title = 'LE code'
  datatype = 'char' length = '5'
# Add subfield_attributes entity =
  'Incident' tag = 'Laboratorial
  Examinations' title = 'LE date'
  datatype = 'date'
# Add subfield_attributes entity =
  'Incident' tag = 'Laboratorial
  Examinations' title = 'LE time'
  datatype = 'time'
# Add subfield_attributes entity =
  'Incident' tag = 'Laboratorial
  Examinations' title = 'LE result'
  datatype = 'char' length = '15'
```

3.1 Using CUDL

As we see in the Figure 2, users conceive the Incident data in the following way. Each record (frame object in the FDB terminology) of the incident diamond combines simple tags (incident code, dates, patient code, social security number, ICD diagnosis) and four lists: 1) doctors treating the patient, 2) a whole table storing data related to the operations that the patient undergoes in the incident, 3) a table of the laboratory examinations done and 4) a table of radiological examinations. It is worth

Incident code	S001
Date started	13/5/2007
Date ended	20/5/2007
Patient code	A001
Social institute code	T001
Incident doctors	I001 I002 I079

Incident operations

Op code	Op time started	Op time ended	Op date	Op doctors
E002	13:35	15:05	14/5/2007	I001 I005 I100 I065
E015	12:00	13:00	16/5/2007	I012 I100 I032

Laboratorial Examinations

LE code	LE date	LE time	LE result
UREA	15/5/2007	10:00	32,4 mg/dl
UREA	15/5/2007	14:30	32,5 mg/dl
UREA	16/5/2007	08:00	31,6 mg/dl
CREA	15/5/2007	10:00	1,17 mg/dl
CREA	16/5/2007	08:00	1,08 mg/dl
PROT	15/5/2007	10:00	7,19 g/dl
PROT	16/5/2007	08:00	6,95 g/dl

Radiological Examinations

RE code	RE date	RE time	File Path
U/S Kidney	16/5/2007	12:00	\\FS1\RIS\Uaz34.tif

Figure 2: An Incident frame object.

noticing that in the case of the operation with code E002 we have a list of the four doctors participating in the operation.

The way the users conceive the data is combined with the CUDL language capabilities for easy but powerful retrieval. An indicative CUDL retrieval statement could be:

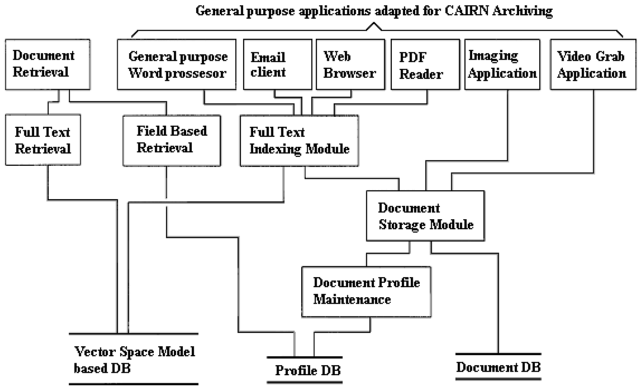

Figure 3: The CAIRN architecture.

```
# Find data when entity = ‘Incidents’
  and subfield = ‘LE date’ restr data
  = ‘16/5/2007’ hide and subfield =
  ‘LE code’ and subfield = ‘LE result’
  and tag = ‘Incident code’ restr data
  = ‘S001’
```

With this statement, we search for an incident with incident code ‘S001’ and want to see all the data concerning the laboratorial examinations that took place on May 16th, 2007. By using the word ‘hide’ we declare that we do not want to include in the results the examination date.

4 THE CUDL LANGUAGE AND THE INTEGRATION OF LEGACY MEDICAL DATA

There is a need for digital asset management, otherwise for designing and implementing documentation tools capable for the collection, organization, administration and retrieval of unstructured and semi-structured information imported from various sources (North Carolina, 2006). Several commercial applications and tools (e.g. NEXTPAGE, HYPERION and AUTONOMY) exist today that incorporate some of the above mentioned capabilities. Karanikolas, 2007, studied and captured user requirements for special purpose Information Retrieval and Documentation tools. Such Computer Assisted Information Resources Navigation (CAIRN) tools are capable for the collection, organization and administration of unstructured and semi-structured medical information imported from various sources. The architecture of CAIRN system incorporates all the required modules and the way that these modules are related.

In the CAIRN architecture (see Figure 3), a module (named “Document Profile Maintenance” module) provides a graphical user interface and users can fill the values of documents’ fields.

It is interesting that each field of a document’s profile is able to hold more than one value and that each value can be composed of one or more attributes. Therefore, a field can have a single value, a list of values or a table of values. In the latter case the rows of the table represent the multiple values and the columns represent the attributes of each value (row).

FDB and CUDL provide us the capability to define all needed database structures for the Document Profile Maintenance module in a single CUDL diamond (a single object implementing all the required relationships). As we can see in figure 4, this object can include the following: file type, document type, relevant persons, incident code, clinical department, date, etc. The file type refers to the file type or the external resource and it can be tiff, jpeg, doc, rtf, xls, avi, mpeg, pdf, blob, dicom, etc. The document type refers to the content type of a document and it can be Kidney U/S, Thyroid U/S, …, Brain CT, Kidney CT, …, ECG, patient discharge letter, operation minutes, laboratorial examination results, video recording of operation, etc. The relevant persons can have multiple values and each one of these values can have two or more attributes (subfields). For example, in case that document type is a patient discharge letter the “relevant persons” field hosts the names of the treating physicians and their roles (head doctor, assistant, etc). In the case that document type is a Brain CT the “relevant persons” field hosts the names of the involved persons and their roles (CT operator, the diagnosis provider (doctor), etc). There are also other tags (incident code, clinical department, date).

In the Figure 4 we can see how users conceive the "legacy" data and the data (documents) imported from external resources or applications.

file type	doc	
document type	patient discharge letter	
relevant persons	Person id	Role
	I001	Head of 2nd Surgical dept.
	I002	Responsible physician
	I079	Assistant (Trainee)
	I082	Assistant (Trainee)
incident code	S001	
clinical department	2nd Surgical dept.	
Date	20/5/2007	

Figure 4: A frame object imported from a "legacy" system.

5 CONCLUSIONS

Eventually, we conclude that the proposed FDB model offers innovative and useful design capabilities to Information System developers, giving them the ability to design and implement medical databases with reduced need for new entities and interconnections among them. It also eliminates the need for reorganisation at both logical and internal levels. Therefore, designers can build databases without having to worry in the case that the users' requirements for the system and the databases will change. The use of CUDL easily solves the problem of creating enterprise systems capable for handling hierarchical structured medical data and also supporting dynamic evolution. CUDL also offers the possibility to integrate the enterprise information system and heterogeneous data imported from "legacy" and independent, external systems. We must mention that the solution of such integration problems is very difficult when traditional Database Management Systems are used.

REFERENCES

Anhøj, J., 2003. Generic Design of Web-Based Clinical Databases, Journal of Medical Internet Research, Vol. 5, No 4 http://www.jmir.org/2003/4/e27/

Badia, A., 2004. Articles, Entity-Relationship modeling revisited, ACM SIGMOD Record, Volume 33, Issue 1

Hearst, M.A., 1999. Untangling Text Data Mining, In ACL'99, 37th Annual Meeting of the Association for Computational Linguistics

Karanikolas, N.N., and Skourlas, C., 2000. Computer Assisted Information Resources Navigation, Medical Informatics and the Internet in Medicine, volume 25, No 2

Karanikolas, N.N., Skourlas, C., 2002. Automatic Diagnosis Classification of Patient Discharge Letters. *In MIE'2002, 23rd International Congress of the European Federation for Medical Informatics*, Budapest, Hungary

Karanikolas, N.N., Skourlas, C., 2005. Naive Rule Induction for Text Classification based on Key-Phrases. *Proceedings of the 6^TH International Conference on Data Mining, Text Mining and their Business Applications*, Skiathos, Greece. WIT Transactions on Information and Communication Technologies, volume 35, Data Mining VI: Data Mining, Text Mining and their Business Applications, WIT Press.

Karanikolas, N.N., 2007. Low cost, cross-language and cross-platform Information Retrieval and Documentation tools. Computing and Information Technology (CIT) Journal, vol. 15, No 1, http://cit.zesoi.fer.hr/downloadPaper.php?paper=869

Karanikolas, N.N., Nitsiou, M., Yannakoudakis, E.J., Skourlas, C., 2007. CUDL language semantics, liven up the FDB data model, *In ADBIS 2007, Eleventh East-European Conference on Advances in Databases and Information Systems,* Varna, Bulgaria, http://www.adbis.org/docs/lp/1.pdf

North Carolina State University, 2006. Digital Asset Management Task Force, Report and Recommendations, http://litre.ncsu.edu/dfiles/DAMTF.pdf

Yannakoudakis, E.J., Tsionos, C.X., Kapetis, C.A., 1999. A new framework for dynamically evolving database environments. Journal of Documentation, Vol. 55, No. 2, 144-158.