Temporal Object-Oriented System (TOS) for Modeling Biological Data

Abad Shah Syed Ahsan Ali Jaffer
Department of Computer Science and Engineering
University of Engineering and Technology, Lahore, Pakistan

Contact/corresponding author: Syed Ahsan
Email: ahsancs@hotmail.com Tel: 92-322-4379469

Abstract: In the last two decades, exponential growth of biological data has posed enormous challenges for the computing and database communities. This wealth of complex and diverse data demands new modeling techniques for seamless and efficient data management. The object-oriented data modeling technique emerged as an optimum choice for modeling scientific data types due to its flexibility and ability to cope with complexity and diversity of the data. In this paper, we investigate the possible applicability of the Temporal Object-Oriented System (TOS) in Bioinformatics domain. We have extended TOS by developing a methodology for dynamic evolution of ROF (Root of Family). Temporal Object Query Language (TOQL) has also been extended by including new operators. In our opinion, our proposed model due to its inherent temporal nature, offers better support to deal with peculiarities of biological data such as data and schema evolution, data provenance and temporal instability. [The Journal of American Science. 2009;5(2):31-39]. (ISSN: 1545-1003).

Keywords: Bioinformatics, temporal object-oriented system, schema evolution, data provenance, temporal instability

1. Introduction

Current research in biology has produced a huge amount of data owning to advances in hardware and wet lab experimental techniques. Further success in the life sciences hinges critically on the availability of better computational and data management tools and techniques to analyze, interpret, compare, and manage this huge volume of data. The existing data management techniques are often challenged by the inability to handle instability, evolving nature and implicit scientific knowledge that is characterized by biological data [4]. These problems stem from various inherent peculiar characteristics of biological data [5,7].

We identify these characteristics of the biological data and information which makes it different and difficult to handle from the conventional (or non-biological) data. Consequently, these characteristics also makes different and difficult to develop a database management system for the handling and maintaining of this type of data. In biology, the data and its inter-relationships have a profound effect on the system of which they are part of therefore it is important and useful to identify the characteristics of both data and the information of their relationships in a biological system [19]. The characteristics are listed and described as follows.

(i) Biological data and its information are highly evolutionary, uncertain and incomplete. The latest research invalidates the established facts as they are completely changed or modified [20].

(ii) This is unprecedented type of data [21]. For example, a molecule, such as a bacteriocin, can be coded ‘mostly’ on plasmids and transposons, though ‘rarely’ on chromosomal DNA, plus a transposon can insert itself into a plasmid: should one classify the gene location as transposon or plasmid, or both.

(iii) Depending on the environment, the behavior of an object/entity of a biological system can vary [22].

(iv) Same data item (biological object) can have different structure in the different environments [24].

(v) Data in bioinformatics is semi-structured [22].

(vi) Even accurately studied and analyzed a biological system can become erroneous and liable to be discarded due to data curation, interpretations, tinkering and experimentation at some later stage [24].

(vii) A biological data object can increase its size in the incremental fashion with the availability of new information about the object [21].

(viii) Biological data is explorative and iterative in nature as it depends upon scientific inquiry.

In our opinion, the conventional data management approaches such as hierarchical, network, structured and relational, although used in certain application areas of biology, are unsuitable and constrained to handle biological data due to the above mentioned characteristics.

In this paper we propose Temporal Object-Oriented System (TOS) for modeling biological data. In our opinion, due to its inherent temporal nature, it offers better support to deal with peculiarities of biological data such as data and schema evolution, data provenance and temporal instability.

Section 2 presents an introduction to TOS and summary of similar significant efforts available in literature. Suitability of TOS for Bioinformatics is explored in Section 3.In section 4 we have presented a methodology for dynamic evolution of ROF as well an algorithm is provided to serve as basis for the implementation of proposed methodology. In Section 5 we provide new operators for TOQL [1] to manipulate semi-structured biological data; use of proposed operators is illustrated with the help of sample queries. Last section identifies some future directions which be perused for more research in Bioinformatics using TOS as basic platform.

2. Related Work

Object-oriented databases and data models have been favored by researchers to store and manipulate complex data in engineering and science. In the last fifteen years, Temporal data modeling for object oriented databases have been an active area of research [9,10,11] and significant work regarding schema evolution for Object-oriented databases is available in the literature [12,13]. Temporal and evolutionary Object-oriented data modeling practices have been used in the areas of clinical and medical informatics successfully [8,14,15,16].

A new domain of applications is identified in [23] by specifying characteristics of the domain for which Temporal Object-Oriented System (TOS) is more suitable. The applications such as Computer-Aided Construction (CAC), and the Web-based applications belong to this class of applications or domain. One typical characteristic of the objects of this class of applications is that they frequently change their structure (instance-variables and methods), state (or data values), or both. Here, we list the main characteristics of the application domain as identified by Shah in [23]. More details can be seen in [23]. We also add a few more characteristics which are endemic to Bioinformatics, thus broadening the scope of the applications conforming to these characteristics. .

i) Applications without hierarchical structure
ii) Rapid prototype development
iii) Incremental growth of objects
iv) Desirability to trace back changes to a specific object
v) Where the grouping of objects is not important
vi) Simultaneous capturing of changes to both parameters (i.e., structure and state) of objects
vii) Polymorphic behavior of objects.
viii) Evolutionary Behavior of the Objects.
ix) Vague Functional Characteristics(This results from imprecise and incomplete data and information about the application domain)
x) The requirements are emergent i.e., the activity of developing, delivering and using the software itself yields more requirements thorough better understanding of the problem.

These characteristics can form the basis of identifying the situations where Temporal Object-Oriented System (TOS) approach is more suited as compared to traditional approaches. Most of the characteristics of bioinformatics domain listed in Section 1 and those listed above are common. These common characteristics suggest that bioinformatics also belong to same class of applications for which Temporal Object-Oriented System (TOS) approach is more suitable.

2.1 The Temporal Object System (TOS)

An object is represented by its structure and state. With the passage of time an object may change its structure and/or its state. By associating time to both the structure and the state of an object, we can keep the history of changes to that object. The Temporal Object System (TOS) data model was introduced in 1992 to capture all types of changes (structural, stature or combination of both) to an object in a uniform fashion [1,2,3]. TOS [1,2,3] defines a temporal object (TO) to be an ordered set of objects which is

constructed at different time instances. A temporal object is represented as *TO = {(SR t1, STt1), (SRt2, STt2),..., (SRtn STtn)}* where *ti* ≤ *ti+1* for all *1* ≤ *i* < *n*, and where the ordered pair *(SRti, STti)* is the *i*-th object of the temporal object which is constructed at the time instance *ti*, with structure *SRti* and state *STti*. An *i*-th object of the temporal object is referred to as its *i*-th stage [1,2,3] A new stage (or *current stage*) of a temporal object shares the structure and/or state from the previous stage, which is not defined in the new stage. A temporal object may also be referred to as *an ordered set of stages*. For example, in Figure 1 the temporal object *TOa* of the family *Fi* has *n* number of stages. The first and last stages of a temporal object are significant because they hold the initial and current knowledge of the temporal object. We refer to these stages as the *birth stage* (stage *Si.a* in Figure 1) and the *current stage* (stage *Sn.a* in Figure 1) of the temporal object *TOa*. The current stage (or the *n*-th stage) is the latest stage that is appended to the temporal object. A new stage is appended to a temporal object when a change occurs to the structure and/or state of the temporal object.

In Fig1, a Temporal Object TO_a is represented by large oval, circles represent the stages, double oval represents root-of-family (ROF), down arrows show transition from one stage to another, where $S_{1,a}$ is birth stage and $S_{n,a}$ is current stage of TO_a, up arrow from TO_a to ROF shows Temporal Inheritance.

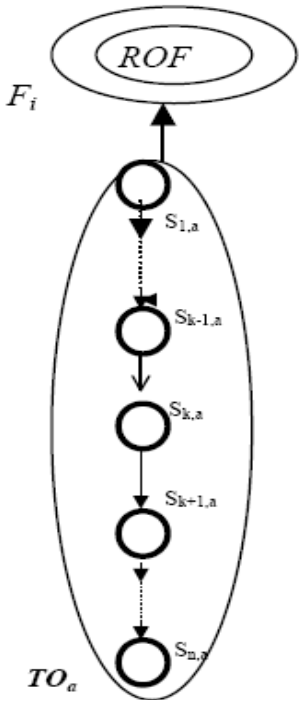

Figure 1: A Temporal Object TO_a

TOS [1,2,3] introduces notion of family to group temporal objects sharing a common context. All temporal objects within a family can be treated in similar way by responding uniformly to a set of messages. A set of similar structures and/or states defines a common context of a family. The common context of family is refereed to as the *root-of-family* (ROF) where common knowledge about all its temporal objects is maintained [1,2,3]. Formation of ROF must be preceding the creation of any of the temporal object of a family. Apparently the concept of family sounds similar to concept of Class which serves as a specification of objects in Class-based object oriented systems. But family in TOS encapsulates more features than a Class. For example, in a Class, the structure of Class is always shared by all of its instances (objects) and change in structure affects all of the created objects. Where as in a family, the structure or state of each temporal object share the ROF only at the time instance of its birth. After that each temporal object is independent and a change in a particular TO doesn't affect the ROF or any other object of the family. In other words, the ROF of family is read only, it doesn't change with the passage of time.

Frequent and rapid changes in biological data demand evolution of schemas. Concept of ROF is not exactly same but analogous to schemas in databases; current architecture of TOS does not provide support

for the evolution of ROF when temporal objects of a family change their stages during their life span. Since creation of new temporal objects in TOS is dependent on ROF, that is why we believe that ROF for TOS should have provision for evolution to make a consistent schema available for new temporal objects.

3. TOS for Bioinformatics

Historically object-oriented paradigm is favorable for the problem solving in complex domains. It is a well acknowledged theory that the object-oriented databases and data models are an appropriate approach for biological data modeling, because they allows to represent strongly interconnected data more directly for simpler usage and maintenance. Biological data is available in so many different types ranging from semi-structured text, unstructured text, images, 3D structures, tables, trees and graph structures. In object oriented models, scientific data types can be efficiently implemented through abstract data type definitions. By means of encapsulation, object-oriented database systems allow to support complex operations within the database schema. Encapsulation is important to maintain complex consistency constraints and to maintain consistency of derived data, but also to integrate complex external operations. In [8], a comprehensive discussion is presented on the use of object-oriented data model for biomolecular databases. It is also considered that object-oriented data models are a suitable basis for the integration of heterogeneous databases which is among the important challenges faced by Bioinformatics community.

In the past, research [17] has shown good results on integration of object-oriented and temporal databases.

Keeping all above facts in mind we have investigated the suitability of TOS [1,2,3] for Bioinformatics. It is unjust to claim that any single data model can serve as solution to all issues, challenges and problems of Bioinformatics; similarly we hereby don't declare that TOS is capable of addressing all the issues of biological data modeling. But important is to note that, TOS provides good support to tackle with the issues of data evolution, data provenance, and temporal instability in particular and data size, distribution in general.

TOS presents an elegant mechanism for evolution of data objects, a temporal object evolves throughout its life as change occurs to its state and\or structure, but TOS does not allow *root-of-family* (ROF)to evolve and imposes a read-only restriction on it, to make TOS model completely evolutionary we have extended this particular aspect of TOS, detail of which is presented in section 3.2. TOS is also capable for addressing the data provenance issue of biological data. Data provenance is knowledge about the origin of some piece of data and the process by which a particular data object reached at its current status and form. Data provenance replies two important queries i.e. about the originating source and derivation history of data items.

With the help of temporal parameters (life sequence, life span and time span) described by TOS, lineage or provenance of data objects can be traced easily. An ordered sequence of stages of a temporal object is referred to as its life sequence. Suppose temporal object TO_a has its life sequence given as:

$L_{TOa} = \{ S_{1,a}, S_{2,a}, \ldots, S_{j,a} \}$

Where $S_{1,a}$ is the birth stage and $S_{j,a}$ is the j-th stage of the TO_a , if we consider the j-th stage as current stage of a temporal object then L is referred to as a complete life sequence of temporal object.

Our proposed extension in TOS described in section 4 particularly deals with the issue of temporal instability of biological data.

3.1 The Cell Structure Example:

It is perceived that Bioinformatics domain may face a crisis due to hesitation of computing and database professionals towards Bioinformatics, as normally computer science professionals are not well equipped with the knowledge of microbiology, exceptions apart [4]. To avoid the crisis it is suggested to choose examples from the domains more familiar to computer science specialists and issues addressed by those scenarios can be mapped to Bioinformatics problems as well [4].

Keeping the above fact in mind, we present a simple hypothetical example to explain the TOS [1,2,3] concepts in detail. The following example will be of equal interest to computing experts due to its simplicity and to microbiologist due to its relevancy.

Cell, in biology, the unit of structure and function of which; all plants and animals are composed. The cell is the smallest unit in the living organism that is capable of integrating the essential life processes. Cells can be separated into two major groups—prokaryotes, cells whose DNA is not segregated within a well-defined nucleus surrounded by a membranous nuclear envelope, and eukaryotes, those with a membrane-enveloped nucleus. Though the structures of prokaryotic and eukaryotic cells differ, their

molecular compositions and activities are very similar. Fig 2 shows the structure of a TOS for two basic categories of a cell, where RTOS denotes the root node of the system and there are two simple families (rectangles): Prokaryote family and Eukaryote family.

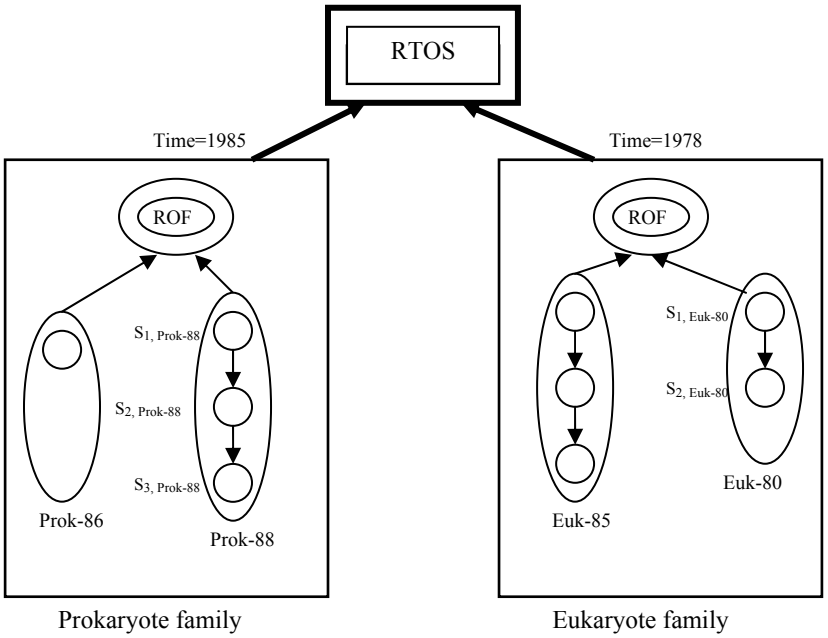

Figure 2: Simple families of two basic cell groups

TOS supports representation of time with different granularities depending upon application domain, but to keep our example simple, in this paper we use an abstract time "year" in each stage, temporal object and family. There are two temporal objects representing information of two prokaryotic cells Prok-86 and Prok-88 (say Bacteria and Archaea) in prokaryote family, Prok-86 comprised of only one stage which is the birth and current stage created at time instance, time=1986. Prok-88 cell is at its current stage S_3 due to evolution in structure or state, maybe due to metabolic reactions etc. , A new sage is appended to a TO only if the structure and/or state associated with its existing current state changes. Two eukaryotic cells Euk-80 and Euk-85 are also modeled in the similar fashion showing transition from their birth stage to current stage.

ROF (Prokaryote)
Instance-variables
{
Time: 1985
Diameter: ,
Surface-to-volume ratio: ,
Num-of-basic-chemicals:
}
Methods {growth_rate: }
	ROF (Eukaryote)
	Instance-variables
	{
	Time: 1978
	Diameter: ,
	Surface-to-volume ratio: ,
	Num-of-basic-chemicals: ,
	Metabolism type: ,
	Cell division type:
	}
	Methods {metabolic_rate: }

Table 1: ROF's of simple families: Prokaryote and Eukaryote

Fig 3(a) shows the birth stage of temporal object Prok-88 and its current stage which resulted due to change in value of one instance variable i.e. Num-of basic-chemicals and addition of one more instance variable i.e Cell wall material, where as Fig 3(b) represents birth stage of temporal object Euk-80 and its current stage which resulted due to change in value of two instance variables i.e. Num-of basic-chemicals and Cell division type and due to addition of one more instance variable i.e Genetic Code.

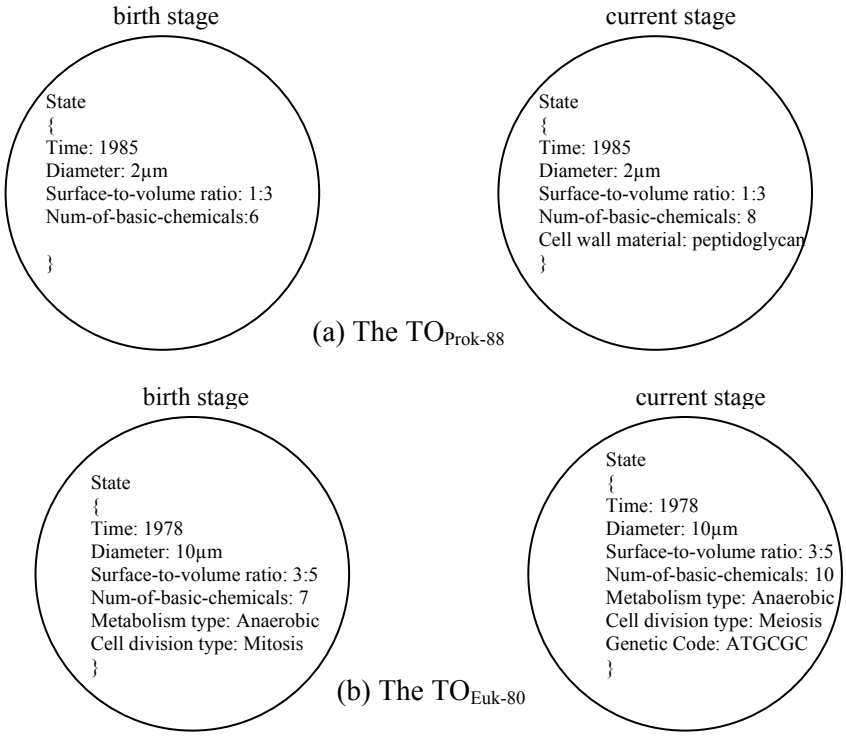

Figure 3: The birth and current stages of temporal objects Prok-88 and Euk-80

3.2 ROF Evolution Methodology

Schemas for biological (or medical) databases have a marked tendency to grow to large sizes to model the diverse sorts of data, and to evolve rapidly as laboratory protocols, instrumentation, and knowledge of molecular biology evolves. Robustness over the long term is a challenge because of evolution to the schema as our understanding of biological systems grows and as new experimental techniques are developed [18]. For this to achieve, biological databanks need a dynamic evolving schema [4].

Schema or class evolution normally takes place due to change in requirements or computational environment or due to discovery of new facts and all of these factors are very common in Bioinformatics. Requirements of biological scientists cannot be precisely determined and generalized. What at present is considered as an attribute may become of such importance over time, that it has to be upgraded to an entity type [4]. Keeping in mind the temporal instability of biological data, ROF must evolve to capture the homogeneous changes available in all the temporal objects of a family. There are there possible situations for a class to change its structure [1,2,3]. These are:

Type I: Adding new instance variables, methods, and/or rules

Type II: Deleting instance variables, methods, and/or rules

Type III: Changes to an instance variable, a method, and/or a rule

We propose a methodology for evolution of ROF when any of above type of change causes a temporal object to promote from a previous stage to its current stage, so whenever some structural change is observed in temporal object of a family, it will become a candidate for ROF update process. If with the passage of time a new attribute become member of all TOs of a family then ROF of this family must be updated by including this new attribute as a member of ROF.

We propose that, structure of a TO should be divided in two logical portions i.e. primary portion and secondary portion. Primary portion will contain information about those members which a temporal object inherits from ROF at its birth time where as secondary portion of temporal object will retain information of members which specifically belong to this temporal object and are not part of ROF. This approach will make comparison process very fast which is required to analyze the consistency and uniformity in the structure of TOs as a result of any change.

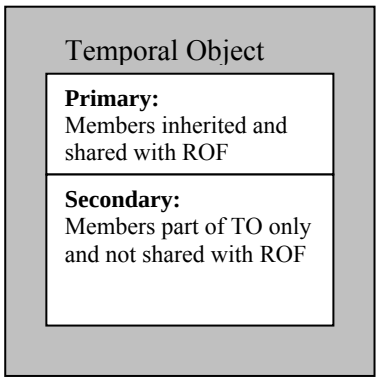

Fig4: Proposed Structure of Temporal Object

Our proposed methodology presents an efficient solution of the problem, whenever a TO moves to a new stage as a result of any change, first of all that change will be analyzed to find out the nature of the change, if new stage resulted due to change in state, then this change will be ignored simply because only change in structure demands schema (ROF) to be evolved. But if structural change is observed then further analysis will be done to find out the type of change i.e. either type I, type II or type III change in a TO. Further analysis will identify that whether the change occurred to the primary or secondary portion of a temporal object.

Type I change is possible only in secondary portion of the temporal object; ROF of a family can be updated only in case of Type I change. Type II and Type III changes are applicable to both primary and secondary portions of a TO, but ROF will not be updated in case of Type II and Type III changes, to keep the history of database consistent we do not allow deletion and modification in ROF.

If some change occurred to the primary portion of a TO, then only the primary portions of the BTOs (Brother Temporal Objects: a term used to denote the temporal objects belonging to a single family of TOS) will be searched to check that whether the change is available in all BTOs. ROF of this family will be updated only if the change is consistent with all the BTOs. By this methodology we do not have to compare the change against each member of a temporal object, simply the relevant portions of TOs will be searched to check the consistency and uniformity of the change.

In case of Type I change to a temporal object, if that change is available in all BTOs, the newly added member(s) will be included in ROF as well as moved from secondary portion to primary portion. For Type II and Type III change, ROF update process will not be initiated.

Suppose, with the passage of time, a new instance variable “Cell wall material” got inducted in each temporal object of Prokaryote family, now further creation of temporal objects of this family requires inclusion of instance variable “Cell wall material” in each newly created temporal object. Using our ROF evolution methodology, ROF of Prokaryote family will be updated against this homogeneous change.

ROF (Prokaryote)
Instance-variables
{
Time: 1985
Diameter: ,
Surface-to-volume ratio: ,
Num-of-basic-chemicals: ,
Cell wall material:
}
Methods {growth_rate: }

Table 2: ROF of simple family Prokaryote after ROF evolution

3.2.1 Algorithm

We provide a pseudo code implementation of above described Methodology to make TOS an evolutionary model by giving provision to update ROF.

```
START
Monitor TOs, when new stage is appended to a temporal object
        Analyze the nature of change
                If new stage is appended to a TO due to change in state only
                        Transfer control to END
                Else
                        Determine the type of change
                        type I change:
                                Search the secondary portion of all BTOs to verify the presence of
                                definition of newly added member
                                        If definition available in all BTOs
                                                Move the definition of newly added member(s) to
                                                primary portion of all BTOs
                                                Update the ROF by including definition of new
                                                member(s)
                                        Else
                                                Transfer control to END
                        type II change:
                                Transfer control to END
                        type III change:
                                Transfer control to END
END
```

4 Conclusion and Future Directions

Biological data poses unprecedented challenges to data management community due to its peculiar characteristics. These characteristics make biological data volatile and temporally unstable, characterized by an evolving schema. Object oriented and Temporal Object System has been used to model complex data such as in CAD, CAM and clinical data. In this paper we identified those characteristics of biological data which suggest that it belongs to the class of applications for which TOS has been suggested as a suitable model. In this paper we proposed modifications to TOS to make it suitable for modeling biological data. The modifications were primarily in Root Of Family (ROF), which was made evolutionary to accommodate and handle the three types of changes which a biological object may go through. In out opinion, TOS is inherently equipped to model biological data and the case study provided in our paper supports this. In our future work we are looking in to the possibility of using TOS to model data provenance, versioning and other related issues. We are also extending Temporal Object Query Language (TOQL) to include new operators for providing natural support to temporal biological queries.

References:

[1] F. Fotouhi, A. Shah, I. Ahmed and W. Grosky, "TOS: A Temporal Object-Oriented System," *Journal of Database Management*, *5(4),* pp. 3-14. 1994.

[2] A. Shah, "TOS: A Temporal Object System," Ph.D Dissertation, Wayne state University, Detroit, Michigan, 1992.

[3] F. Fotouhi, A. Shah and W. Grosky, "TOS: A Temporal Object System," The 4th International Conference on Computing & Information, Toronto, Canda, 353-356, 1992.

[4] S. Ahsan and A. Shah, "Identification of Biological Data Issues and Challenges through Real World Examples", *ACIT'2005 conference*, Jordan. December 2005.

[5] T. Topaloglou, "Biological Data Management: Research, Practice and Opportunities," *Proceeding of the 30th VLDB Conference*, Toronto, Canada, 2004.

[6] F. Achard, G. Vaysseix, and E. Barillot, "XML, bioinformatics and data integration". Bioinformatics Review, vol. 17 no. 2, 2001 pages 115-125, Oxford University Press. 2001.

[7] H.V. Jagadish, and F. Olken, "Database Management for Life Science Research," *OMICS: A Journal of Integrative Biology* 7(1):131-137, 2003.

[8] K. Aberer, "The Use of Object–Oriented Data Models for Biomolecular Databases," Proceedings of the Conference on Object–Oriented Computing in the Natural Sciences (OOCNS) '94 , Heidelberg, 1995.

[9] E. Bertino, E. Ferrari, and G. Guerrini. T Chimera - A Temporal Object-Oriented Data Model.*Theory and Practice of Object Systems*, 3(2):103–125, 1997.

[10] E. Rose and A. Segev. TOODM - A Temporal Object-Oriented Data Model with Temporal Constraints. In *Proc. 10th Int'l Conf. on the Entity Relationship Approach*, pages 205–229, October 1991.

[11] S.Y.W. Su and H.M. Chen. A Temporal Knowledge Representation Model OSAM*/T and its Query Language OQL/T. In *Proc. 17th Int'l Conf. on Very Large Data bases*, pages 431–442, 1991.

[12] I.A. Goralwalla, D. Szafron, M.T. O¨ zsu, and R.J. Peters. Managing Schema Evolution using a Temporal Object Model. In *Proc. 16th International Conference on Conceptual Modeling (ER'97)*, pages 71–84, November 1997. Proceedings published as Lecture Notes in Computer Science, David Embley and Robert Goldstein (eds.), Springer-Verlag, 1997.

[13] F. Ferrandina, T. Meyer, R. Zicari, G. Ferran, and J. Madec. Schema and Database Evolution in the O2 Object Database System. In *Proc. 21st Int'l Conf. on Very Large Data Bases*, pages 170–181, September 1995.

[14] W.W. Chu, I.T. Ieong, R.K. Taira, and C.M. Breant. A Temporal Evolutionary Object-Oriented Data Model and Its Query Language for Medical Image Management. In *Proc. 18th Int'l Conf. on Very Large Data Bases*, pages 53–64, August 1992.

[15] C. Combi, F. Pinciroli, and G. Pozzi. Managing Different Time Granularities of Clinical Information by an Interval-Based Temporal Data Model. *Methods of Information in Medicine*, 34(5):458–474, 1995.

[16] I.A. Goralwalla, M.T. O¨ zsu, and D. Szafron. Modeling Medical Trials in Pharmacoeconomics using a Temporal Object Model. *Computers in Biology and Medicine - Special Issue on Time-Oriented Systems in Medicine*, 27(5):369 – 387, 1997.

[17] http://www.dagstuhl.de/Reports/98471.pdf Seminar Report for Dagstuhl Seminar on Integrating Spatial and Temporal Databases, Schloss Dagstuhl-Wadern, Germany, 1998. [Accessed: May 7, 2007].

[18] http://hpcrd.lbl.gov/staff/olken/wdmbio/wsproposal1.htm Web page of Workshop on Data Management for Molecular and Cell Biology, 2003. [Accessed: April 25, 2007].

[19] Bornberg-Bauer, E. and Paton, N.W.,,'Conceptual Data Modeling for Bioinformatics, *Briefings in Bioinformatic s*(2002

[20] Ostell, JM., S.J. Wheelan, J.A. Kans "The NCBI Data Model in Bioinformatics: A Practical Guide to the Analysis of Genes and Proteins 2ne Edition. John Wiley & Sons Publishing. . 2001. ISBN: 0471383910 pp. 19-44.

[21] Marijke Keet. (2003). Biological Data and Conceptual Modeling Methods. *the Journal of Conceptual Modeling, Issue: 29*

[22] Alberts, B..Molecular Biology of the Cell. Garland Science Publishers , New York (2002)

[23] Shah, A., (2001) .A Framework for Life-Cycle of the Prototype-Based Software Development Methodologies. *the Journal of King Saud University*, *Vol. 13*, Pp. 105-125,.

[24] P. Baldi and S. Brunak, (2004). Bioinformatics: The Machine Learning Approach " 2nd edn, MIT Press**,** Volume 19 , Issue 1.

10/27/2008