Adaptive Retrieval Time-Related Data Model for Tracking Factors Affecting Diabetes

Ibrahim AlBidewi[1], Fahad Alotaibi[3]
Department of Information System
Faculty of Computing and Information Technology
King Abdulaziz University, Jeddah, Saudi Arabia

Nashwan Alromema[2]
Department of Computer Science
Faculty of Computing and Information Technology-Rabigh
King Abdulaziz University, Rabigh, Saudi Arabia

***Abstract*—In the last four decades several dozens of representing time-oriented data/knowledge bases have been presented. Some of these representations violate First Normal Form (1NF) by using Non-First Normal Form (N1NF) prototypes and temporal nested representations, while others simulated the concepts of temporal data with relational data representation without violating 1NF. In this article, a new interval-based knowledge representational data model with an optimized retrieval techniques are employed for modeling and optimality retrieve a biomedical time-varying data (factors/observations that affect the diabetes). The used time-related data model is more compact to represent time-varying data with less memory (capacity) storage with respect to the main representations in the literature, but which is as expressive as those representations (a transformation algorithms show that data represented in this model can be transferred to/from the representations in the literature with zero percent loss of information). A new data structure is defined with the optimal retrieval techniques to prove some basic properties of the used time-model and to ensure that the time-model is an extension and reduction of the main representations in the literature, namely *TQuel* and BCDM. The expressive power, reducibility, and easy implementation of the proposed model, especially for the legacy systems, are considered as advantages of the proposed model.**

Keywords—Diabetes database; time-data model; diabetes observations; valid-time data; knowledge-based data

I. Introduction

The reproduction method of Time-varying data representation in RDBMS is considered as bases of time-oriented information, and it provides temporal information models and stores data associated with the past, present and future. TDB also offers communicative and proficient ways to reproduce, stock up, and inquire about special temporal situation of the stocked information in contrast to the conventional databases which record single state of the real world phenomena. TDB is a vital area of study, with a dynamic population of a number of hundred investigators who have brought about thousands of research articles over the recent thirty years [1, 2, 4]. Conventional information record is utilized to stock and treat the information that refers to the present moment in time, without supporting the temporal features which maintain temporal database and store data referring to what went before, to current time and to upcoming time. There has been a discussion within the recent thirty years on how to represent, execute and inquiry about time-based information record in an efficient way. An increasing concern with time-varying information records in several practice fields which address matters in managing temporal information [2, 3, 4, 9, 14, 18, 26, 33]. Most of these publications touched upon the various features of time-varying information records.

The remaining part of this article is structured as it follows: Section 2 introduces the problem background and the existing techniques that deal with modeling time-varying data, Section 3 introduces the classification of time-varying information prototype expansions. Section 4 describes the representational methodology of time-data models. Section 5 compares three TDB prototypes in terms of memory storage representation. Lastly, Section 6 draws a conclusion and a close this article.

II. Background

Handling time-varying database can be accomplished by two approaches as mentioned in [5, 6, 10], namely, (1) The integrated approach is to build a complete TDB Management System from zero, which offers a primal information kind and deals with the various situations/time examples of the information being stocked. This approach involves bottom-up construction an entire TDBMS, which is a very huge and time-wasting job. It is also not easy since the fundamental values used by profitable DBMS to increase effectiveness of functions have to be improved and many of abstract works need to be performed to prove that the novel structure is entirely wholly perfect. The scope of time and workforce needed for this advance is parallel to that required by profit-making sellers to build up DBMS that all of us know nowadays. (2) The stratum method is to develop a strategy that expands time-independent information representation to time-varying information representation on top of common DBMS that behaves as a layer. The second method does not entail any alterations to the available information record technique. It can be merely elaborated by constructing an innovative method for time-varying backup in addition to the available common DBMS.

Modeling temporal database in relational framework by appending extra vertical divisions for time to the correlation(s) appears to be a straightforward way [5, 11, 12, 13, 20], yet it does not solve many delicate matters particularly for time-varying information. Many expansions of relational information representation to time-varying information

This project was funded by the Deanship of Scientific Research (DSR) under grant no. G- 27-005-1430.

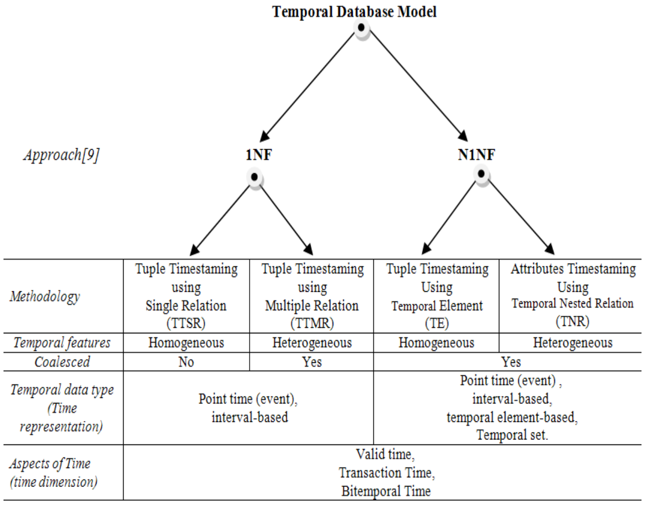
The authors, therefore, acknowledge with thanks to DSR's technical and financial support representations Methodology.

representation have been offered with their strengths and weaknesses [5]. More than 23 time-varying information prototypes have been proposed, some of these prototypes are in [13, 16-22, 27, 28,29-30].

In this paper, a taxonomy of all the possible extensions of Temporal Database Model (TDM) are developed as shown in Fig. 1 and according the naming conventions appear in the consensus glossary in [15]. Based on that, a novel information reproduction, a semantic time-varying relational data model that extends model in [33] (Snodgrass' Tuple Timestamped data representation) are introduced to deal with interval-based knowledge representation in relational data model. A new data structures, a cost model for the memory storage use for the proposed time-data model are defined. The extended model, and time-based relational algebra also provided to prove some essential characteristics and to ensure that the proposed model is an extension and reducible of BCDM.

III. Modeling Time-Varying Data

Modeling time-varying information in relational framework differs in many perspectives [9], the most recurrently declared methods are tuple timestamping with 1NF, and attribute timestamping with N1NF as shown in Fig. 1. Integrating time in relational data representation could be done by one of the TDB prototypes approaches that are shown in Fig. 1, which shows the approaches of modelling time-varying data in relational framework. These approaches can be either by applying 1NF or by violating 1NF using multi-value attributes or nested relations. The first method -1NF- has two differentiations, specifically; (1) Tuple Timestamping Single Relation (TTSR), this approach incorporates time in relational data model by adding extra timestamps attributes to the conventional association. Several time-varying information representation debated in [9] can be classified under this method. An instance of some of these time-varying information prototypes are LEGOL 2.0 by Jones [29], Temporally Oriented Data Model by Ariav [31], HSQL by Sarda [32], and TQuel by Snodgrass [33]. The time-varying information record representations can be harmonized or diverse. The time-varying harmonized feature of temporal information record relations [32] is defined as time-varying associations in which the temporal time of all information merits (that is to say the time over which they are determined) in each tuple is the same [9]. And as stated in [9, 24, 29] the representations that employ tuple timestamping are essentially temporally identical because only temporally uniform associations are probable, to conclude that the representations that are based on TTSR approach are temporally homogeneous. Temporally heterogeneous database relations are temporal database relations in which the temporal time of the information merits in each tuple can be dissimilar. Since 1NF tuple timestamped temporal relations that allow extension over or following tuples of equivalent values- are not coalesced [9] then the representations under this approach are not coalesced because tuples of equivalent values- are not unacceptable. The time representation of the stored facts can be event timestamp, or interval timestamps. Finally, this approach can model the diverse temporal time aspects like valid-time, transaction time, or Bitemporal time. TTSR method is not proficient since it presents tautology, where information merits that vary at diverse time are recurring in manifold tuples. (2) Tuple Timestamping Multiple Relations (TTMR), this method has resolved the issue of information tautology in TTSR by breaking down the temporal correlation as ensuing: Temporal values are spread over manifold associations, and non-temporal values are regrouped into a segregate association. Several time-varying information representations debated in literature [9] can be classified under this method. An instance of some of these time-varying information representations is the Temporal Relational Model by Navathe [30], and Snodgrass [8], and Tansel [16]. The temporal information representations under this method are attribute timestamping, thus, they are heterogeneous since the temporal time of the element worth in each tuple can be dissimilar, coalesced because value-equivalent tuples are disallowed. The time representation of the stored facts can be event timestamp, or interval timestamps. Finally, this approach can model the diverse time aspects like valid-time, transaction time, or Bitemporal time. The problem with the data representations under this approach is when the data for an entity is need to be to combined, a variant of bond known as temporal intersection join would be required, which is by and large costly to be accomplished. The second approach (N1NF) employs multi-value attributes, or in other words nested relations for tuple timestamping or attributes timestamping. As is shown in Fig. 1 the tuple timestamping approach uses sets of time chronons (the smallest time unit) for timestamping associated tuples. BCDM by Jensen and Snodgrass [9] is an example of this model. Prototypes under this category are homogeneous and coalesced. In attributes timestamping approach non-atomic attribute values are associated with temporal time. Historical Relational Data Model by Clifford-2 [33], HQuel by Tansel [34], Homogeneous Relational Model by Gadia-1 [32], Heterogeneous Relational Model by Gadia-2 [35], and TempSQL by Gadia-3 [35] are temporal data prototypes that can be considered under this category. N1NF may not be competent of directly utilizing a relational stocking construction or a questioning appraisal method that depends on tiny element merits, thus this approach (N1NF) will be excluded from this study. A variety of methods for managing time-varying information can be spotted in [1]. The traditional modelling technique for temporal database is not efficient since the prototypes that are based on 1NF model introduce redundancy, while the prototypes which are based on N1NF model may not be capable of directly implemented in conventional DBMS [9]. Using the integrated approach for implementing TDB prototypes in conventional DBMS is a costly task and can be accomplished only by the DBMS vendors [6]. Therefore, many temporal database prototypes have been anticipated in attempting to seize the time-varying attribute of information in ease of use and system performance [2, 34, 27]. The implementation of TDB in conventional DBMSs is not an easy task [34], this is because conventional DBMSs do not offer prototypes to back up and treat the temporal dimensions of the implemented database [5, 11]. Because of the unnecessary tautology of information, costly execution, and the complexity of implementing TDB in relational framework, an intelligent way is required to represent, execute, and question TDB in correlation outline.



Methodology	Tuple Timestaming using Single Relation (TTSR)	Tuple Timestaming using Multiple Relation (TTMR)	Tuple Timestaming Using Temporal Element (TE)	Attributes Timestaming Using Temporal Nested Relation (TNR)
Temporal features	Homogeneous	Heterogeneous	Homogeneous	Heterogeneous
Coalesced	No	Yes	Yes	
Temporal data type (Time representation)	Point time (event), interval-based		Point time (event) , interval-based, temporal element-based, Temporal set.	
Aspects of Time (time dimension)	Valid time, Transaction Time, Bitemporal Time			

Fig. 1. The Taxonomy of Temporal Data Model.

This section will begin by presenting the proposed TDB model for modelling the biomedical data. This data model is termed as Tuple Timestamp Historical Relation (TTHR). The generic means of temporal database is an information record with some sort of temporal back up. In this paper the focal point is on the valid time characteristic of time-varying information in relational information records; nevertheless, the debate might simply be expanded to information records that sustain transaction-time and bitemporal time. Then the semantics of associating time aspect to temporal database objects as well as representing these objects in TTHR will be explained in this section. Time Domain is assumed linear, bounded, and discrete constructional features of the temporal field for timestamping the proposed TDB representation as in [9]. Chronons (the smallest time unit) are utilized that have length and limited precision [14]. The temporal domains have overall orders and are similar in form and relations to the domain of ordinary figures. Associating Time Aspects to the Database Objects can be affirmed as valid time and transaction time where together are conceived as the universal features of all information record data [28, 29]. Since the data in the information schemes are built by a number of characteristics, then valid time and transaction time characteristic related to a group of aspects might be employed to record the valid time of the data, or the time when the data are present in the information record. The duration time and transaction time are universal features of the unit types or association types. The lifetime of an information record entity as in [29] is the moment in time over which it is identified, or the moment in time when the matching entity exists in the reproduced reality. The valid-time lifespan of an information record object refers to the point in time when the entity exists in the constructed reality, whereas the transaction-time lifespan of an information record object refers to the point in time when the object is present in the information record. The information record object's lifespan can be reduced to valid-time as mentioned in [31]. In this study, only the valid-time lifetime of an information record entity is considered, where the matching entity can have one or more time-varying attributes. For these time-varying attributes, a valid-time aspects is associated in such a way that the valid-time aspects are rigorously enclosed in or equal to the lifespan of the object (example valid-time interval [3, 7] for lifespan interval [1, 34], for short the term lifespan time aspect is used instead of valid-time lifespan to avoid the confusion). In general, an object type in an abstract data representation which matches with one relational schema in the rational data representation can be associated with the subsequent time features: (1) Lifetime period feature, if the lifetime of that object sort is a division of the represented moment in time, the object category will be associated with the lifetime period features. (2) Transaction time aspect indicates the presented time of the object in the information record. Association categories in an abstract representation which matches with one relational schema in the rational representation can be attributed by: (1) Lifetime period feature where the lifetime of the association kind should be a division of the lifetime of the contributing objects. (2) Transaction time, where the meaning of the transaction time features of the association kind is a division of the lifetime or transaction-time features of the contributing objects. The time-varying backup for association kinds (encompassing lifetime and transaction-time) may be as an immediate occurrence where the characteristics of these association kinds are thought of as predetermined (time-independent) features. Also, it may be a lasting occurrence where the characteristics of this association kind may be temporal features, or the non-temporal representation of these characteristics. The characteristics in the abstract representation that are matching with the vertical divisions of a chart in the rational representation might be regarded as temporal or non-temporal characteristics. In case of time-varying attributes, the time aspects that can be assigned to these attributes are as follows: (1) Valid-time feature, where these time features are supposed to be a division of the lifetime of the related object kinds or association sort. (2) Transaction-time, where these temporal features are supposed to be a division of transaction-time of the related object kind or association sort. (3) Bi-temporal time features, where these temporal features have the composite constraints of valid-time and transaction-time features. This temporal relationship and the constraint of characteristics are applicable for the characteristics that ensue from single-to-multiple association. The data in Table I depicts the features of time for which they ought to be supplied for the information record items and as shown in [12]. The temporal features that might be connected to diverse information record items are dissimilar from one information record item to another. For instance, a lifetime period may not be connected with characteristics as it shows the availability of the item or unit sort in the modeled reality. Thus, the relationship of a point in time with a diverse information record item ought to be constrained in accordance with Table I.

TABLE I. INFORMATION RECORD ITEMS AND THEIR PROPPED FEATURES OF TIME

Database object	Lifespan (LS)	Valid Time (VT)	Transaction Time (TT)	Bi-temporal (Bi)
Entity (E)	Yes	No	Yes	No
Attribute (A)	No	Yes	Yes	Yes
Relationship (R)	Yes	Yes	Yes	Yes

Coding that, a lifetime moment in time can be coupled with a correlation kind that has been being by itself; however, in case the association is carried out as an aspect of one of the constituting objects, it will be processed as an aspect and cannot be coupled with a lifetime scope.

A. Representing and RetrievingTime-Vaying Data (Diabetes Data) using TTHR Approch

A temporal object in *TTHR* can be associated with and restricted to time aspects stated in Table I. The valid time features for attribute objects, and lifetime aspects for entity and relationship objects are considered as a timestamping method in this article (it could trivially extend this model by replacing lifespan time aspects by transaction-time aspects, with such an expansion, the proposed model is insignificantly a constant expansion of BCDM [9, 29]). The field of applicable times can be represented as $D_{VT} = \{t_1, t_2, \cdots, t_{VT}\} \cup \{\infty\}$ and the field of lifetime intervals can be represented as $D_{LS} = \{t_1, t_2, \cdots, t_{LS}\} \cup \{\infty\}$, where ∞ is a *distinguished value* that represents a very large value for time (say 3000 ac). The time domain of both *VT* and *LS* is an order of natural numbers. In what follows, R_T is employed to indicate the schema of temporal relation. The group of attributes, which constitute R_T may be classified into ***k***ey attributes ***(K)***, non-temporal attributes (***U***nchangeable ***(U)***), temporal attributes (***C***hangeable**(*C*)**), and ***T***imestamps attributes (***T***). $R_T = \{A_K, A_U, A_C, A_T\}$ .Where $A_K = \{A_{K1}, A_{K2}, \cdots, A_{Ki}\}$, $A_U = \{A_{U1}, A_{U2}, \cdots, A_{Un}\}$, $A_C = \{A_{C1}, A_{C2}, \cdots, A_{Ci}\}$, and $A_T = \{T_{ls}, T_{le}\}$, the domain of T_{ls} and T_{le} is D_{LS} , where T_{ls} represents the Lifespan Start Time (*LSST*) and T_{le} represents the Lifespan End Time (*LSET*). Temporal relational schema R_T in *TTHR* can be represented as $R_T = \{A_K, A_U, A_C, A_T\}$, and a new secondary association representation VT_R_T is formed as $VT\_R_T = \{A_K, index, \alpha, A_T\}$ where *index* is a factor to spot the time-varying aspect A_{Cm} begin updated (where $1 \le m \le i$), α is the updated value of A_{Cm} and $A_T = \{T_{vs}, T_{ve}\}$, the domain of T_{vs} and T_{ve} is D_{VT} , where T_{vs} constitutes the Valid Start Time (*VST*) and T_{ve} constitutes the Valid End Time (*VET*). The aim of this modeling is to maintain the most recent (existing) updated information in one relation R_T , and the chronological alterations of the logical soundness of the temporal information in a secondary association VT_R_T . An example of correlation is indicated by r_t , and vt_r_t , where $r_t(R_T)$ implies r_t is an example of R_T , and $vt\_r_t(VT\_R_T)$ implies vt_r_t is an example of VT_R_T . For tuples x, y and z are employed**,** accordingly a tuple, $x = (a_K, a_U, a_C, a_T)$ in the association example $r(R_T)$ is composed of a number of attribute values associated with the T_{ls} and T_{le} , whilst the tuple(s) $vt\_x = (a_K, index, \alpha, a_T)$ in the relation instance $vt\_r_t(VT\_R_T)$ is/are strongly correlated and in reference to tuple x **.** The tuple(s) is composed of the principal key of x , the characteristics (*index*) of the temporal aspect in x , the updated temporal features value α in x , and the validity of the updated characteristic T_{vs} and T_{ve} . A division of the field of natural life time is connected to each tuple in R_T demonstrates that the availability of the item registered by the tuple is factual in the reproduced veracity all through every lifetime chronon in that division. A division of field of valid times is coupled with every tuple in VT_R_T , denotes the reality that the tuple vt_x registers the alteration of the validity of a_{cm} in x **.** This reality is regarded as factual in the reproduced veracity throughout each valid time chronon in that division, noting that the time of validity is stringently comprised in the temporal lifetime of x . Therefore, the coupled time with a tuple in *TTHR* is interval-based temporal timestamp. The semantics of the proposed data model can be informally explained via a simple example as show in Fig. 2, where a temporal relation schema *Patients* corresponding to R_T in *TTHR* is used to record *Patients* information. The auxiliary relation *VT_Patients* is employed to register the alterations of the validity of the time-varying characteristics in *Patients* and the alterations of the lifetime of the items in *Patients*. The diverse kinds of characteristics of *Patients* and *VT_Patients* are:

$$A_K = \{p\_id\}$$

$$A_U = \{p\_name, Birth\_date, sex, family\_\inf ected\}$$

$$A_C = \{weight, hight, BMI, GF, Cho, Tri, Sua, BP, Hyp,$$

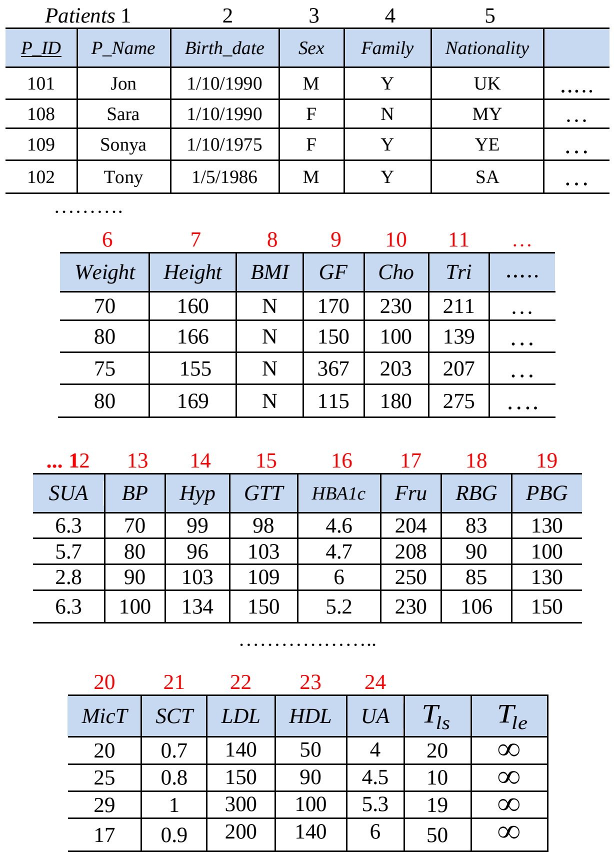
$$GTT, HBLIC, Fru, RBG, PBG, GTT, HBLIC$$
$$Fru, RBG, PBG\}$$

$$A_T = \{T_{ls}, T_{le}\}$$

Semantically, the attributes of the Patients relation have the following meaning: P_Id– Patient’s identified number, Name- Patient’s name, Birth_date- Patient’s birth date, sex- Patient’s gender, Family_infected- a Boolean attributes to answer either any of Patient’s relative had sugar or not, nationality- Patient’s ethnicity, Weight- Patient’s weight, Height- Patient’s height, BMI- Patient’s Body mass index. The rest of the attributes are sugar related test observations, the glossary of these terms and the meaning of each can be found at “Joslin Diabetes Center”. Table II outlines the full observations’ names, descriptions, the test range, and the abbreviation of each one. The temporal database relation *Patients_VT* (Fig. 3) is an auxiliary relation of patient’s relation that recodes the tracking of different diabetes observations changes of the patients. It consists of four attributes, *P_id, Att_index, Updated_V,* and temporal attributes T_{vs} and T_{ve} , such that T_{vs} represents the Valid Start Time (*VST*) and T_{ve} represents the Valid End Time (*VET).* The semantics of *patients_VT'* attributes are, *P_id*: is a foreign key referencing to *P_id* in patients relation. *Att_index*: contains data about the locations of time-varying attributes in patient’s relation. *Updated_V*: contains the old data (observations’ result) of the updated time-varying attributes in patients relation that is indexed by *Att_index*. T_{vs} : is a temporal attribute that represents (*VST*), andT_{ve} : is a temporal attribute that represents (*VET*).

TABLE II. LIST OF OBSERVATIONS FOR DIABETES PATIENTS' TESTS [36, 37]

Observation	Test	
	Descriptions	Normal Range
Glucose Fasting (GF)	a blood test in which a sample of blood is drawn after an overnight fast to measure the amount of glucose in blood.	65-110 mg/dL
Cholesterol (Cho)	Cholesterol is tested at more frequent intervals	up to 200 mg/dL
Triglycerides (Tri)	This tests is used to help identify an individual's risk of developing heart disease.	up to 150 mg/dL
Serum Uric Acid (SUA)	This test is used to diagnose gout.	2.50-6.50 mg/dL
Blood pressure(BP)	the pressure of circulating blood on the walls of blood vessels.	120/80- 140/90
Hypertension (Hyp)	is defined as blood pressure higher than 140 over 90 mmHg (millimeters of mercury). .	90-140 and above
Glucose Tolerance Test(GTT)	A glucose tolerance test measures how well your body's cells are able to absorb glucose, or sugar, after you ingest a given amount of sugar.	95-180 mg/dL
Glycosylated Haemoglobin (HBA1c)	A form of hemoglobin that is measured primarily to identify the three-month average plasma glucose concentration.	4.5-7
Fructosamine (Fru)	Fructosamines are compounds that result from glycation reactions between a sugar.	200-285 µmol/L
Random Blood Glucose (RBG)	Random blood sugar (RBS) measures blood glucose.	80-140 mg/dL
Postprandial Blood Glucose (PBG)	A postprandial glucose test is a blood glucose test that determines the amount of a type of sugar, called glucose, in the blood after a meal.	110-160 mg/dL
Microalbumin Test (MicT)	A test to detect very small levels of a blood protein (albumin) in your urine.	< 30 mg of protein
Serum Creatinine Test (SCT)	A creatinine blood test measures the level of creatinine in the blood.	0.5 to 1.1 mg/dL (female) 0.6 to 1.2 mg/dL (male).
Low Density Lipoproteins(LDL)	LDL is called low-density lipoprotein because LDL particles tend to be less dense than other kinds of cholesterol particles.	Up to 130 mg/dL
High Density Lipoproteins(HDL)	HDL is known as the good cholesterol because it carries LDL, triglycerides, and harmful fats and returns them to your liver for processing.	> 35mg/dL
Uric Acid (UA)	Uric Acid is excreted (removed from your body) in your **urine**.	2.4-6.0 mg/dL (female) 3.4-7.0 mg/dL (male).



Patients

1	2	3	4	5	
P_ID	*P_Name*	*Birth_date*	*Sex*	*Family*	*Nationality*
101	Jon	1/10/1990	M	Y	UK	.....
108	Sara	1/10/1990	F	N	MY	…
109	Sonya	1/10/1975	F	Y	YE	…
102	Tony	1/5/1986	M	Y	SA	…

……….

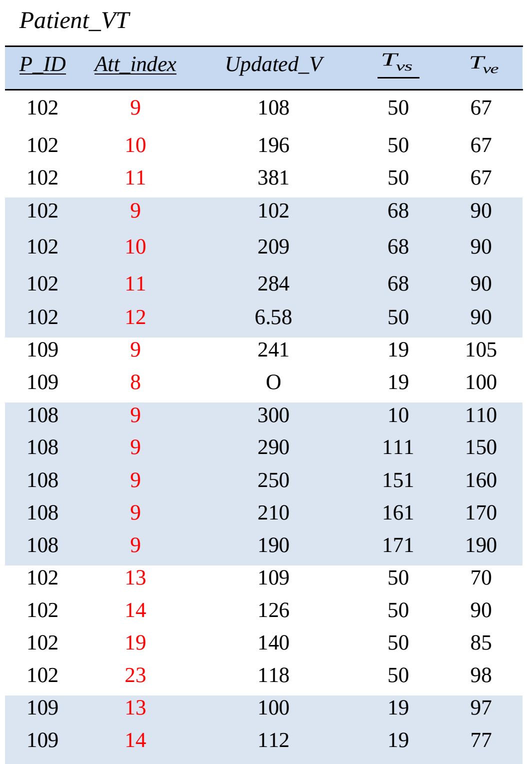
6	7	8	9	10	11	…
Weight	*Height*	*BMI*	*GF*	*Cho*	*Tri*	…..
70	160	N	170	230	211	…
80	166	N	150	100	139	…
75	155	N	367	203	207	…
80	169	N	115	180	275	….

... 12	13	14	15	16	17	18	19
SUA	*BP*	*Hyp*	*GTT*	*HBA1c*	*Fru*	*RBG*	*PBG*
6.3	70	99	98	4.6	204	83	130
5.7	80	96	103	4.7	208	90	100
2.8	90	103	109	6	250	85	130
6.3	100	134	150	5.2	230	106	150

………………..

20	21	22	23	24		
MicT	*SCT*	*LDL*	*HDL*	*UA*	T_{ls}	T_{le}
20	0.7	140	50	4	20	∞
25	0.8	150	90	4.5	10	∞
29	1	300	100	5.3	19	∞
17	0.9	200	140	6	50	∞

Fig. 2. The Representation of Patients Database relation using TTHR Approach.



Patient_VT

P_ID	*Att_index*	*Updated_V*	T_{vs}	T_{ve}
102	9	108	50	67
102	10	196	50	67
102	11	381	50	67
102	9	102	68	90
102	10	209	68	90
102	11	284	68	90
102	12	6.58	50	90
109	9	241	19	105
109	8	O	19	100
108	9	300	10	110
108	9	290	111	150
108	9	250	151	160
108	9	210	161	170
108	9	190	171	190
102	13	109	50	70
102	14	126	50	90
102	19	140	50	85
102	23	118	50	98
109	13	100	19	97
109	14	112	19	77

Fig. 3. Auxiliary Temporal Database relation of Diabetes Patients.

The methodology of the retrieval techniques of the proposed time-varying data model is different from conventional databases because temporal database holds a sequence of snapshot relations. Therefore, there are three different types of queries, namely, current query, sequenced query and non-sequenced query [8, 20]. The most prevalent queries (current) are of the form "what is valid now?", the second type of queries (non-sequenced) is of the form "what was valid at any time?", and the third type of queries (sequenced) are of the form "what is valid at/during a certain point/interval (period) of time?". The valid time data represented by the proposed data model will be shown how it can be queried using the different types of temporal query. For querying the biomedical data, an interested queries according to the diabetes medical doctor is asking about the latest reading of any of the 19s observations as in Fig. 2, the reading values of any observations in any point of time, and the reading values of any observations when the values of other observations has specific reading. Queries B1 to B2 with Fig. 4, Fig. 5, and Fig. 6 depict and show the results of such queries.

Query B$_1$: What is the latest reading of *Glucose Fasting* of Tony with *P_ID =102*?

$$Q_{B_1} \leftarrow \pi_{\text{P\_Id,P\_}Name\text{,Birth\_date,GF}}(\sigma_{P\_Id=102}(Patients))$$

P_ID	*P_Name*	*Birth_date*	*GF*
102	Tony	1/5/1986	115

Fig. 4. The Result of Glucose Fasting Query.

Query B$_2$: What are the readings of *Glucose Fasting* of Tony with *P_ID =102* during the period of [60, 100]?

$$Q_{B_2} \leftarrow \sigma_{\substack{T_{vs}\leq 100 \\ and \\ T_{ve}\geq 60}}(P\_\text{GF})$$

P_ID	*P_Name*	*GF*	T_{vs}	T_{ve}
102	Tony	108	60	67
102	Tony	102	68	90
102	Tony	115	91	100

Fig. 5. The Result of Glucose Fasting Query in the Period [60, 100].

Query B$_2$: What are the readings of *Glucose Fasting* of Tony with *P_ID =102* when the reading of *Serum Uric Acid* was not in the normal range?

For this query, the period of validity of *Serum Uric Acid* when it was in up normal reading (according to Table II this observation value ranges from 2.5 to 6.5) will be retrieved according to the query Q_{SAU} below. The overlap period of validity of *Glucose Fasting* with the period of validity of *Serum Uric Acid* when it was in up normal will give the result as shown in Fig. 4 to 6.

$$Q_{SUA} \leftarrow \sigma_{P\_Id\leq 102..and..SUA>6.5}(P\_SUA)$$

P_ID	*P_Name*	*GF*	T_{vs}	T_{ve}
102	Tony	108	60	67
102	Tony	102	68	90
102	Tony	115	91	100

SAU_Upnormal

$$Q_{B_3} \leftarrow ((\text{P\_GF} \underset{\substack{GF\_P\_id=SAU\_P\_id \\ and \\ \text{GF\_}T_{vs}\leq SAU\_T_{ve} \\ and \\ \text{GF\_}T_{ve}\geq SAU\_T_{vs}}}{\rhd\!\lhd} SAU\_Upnormal))$$

P_ID	*P_Name*	*GF*	*SAU*	T_{vs}	T_{ve}
102	Tony	108	6.58	60	67
102	Tony	102	6.58	68	90

Fig. 6. The Result of Glucose Fasting Query when Serum Uric Acid was up Normal.

B. Representing Time-Vaying Data using TTSR Approchs

More than 23 time-varying data models have been introduced in the literature [9]. These data prototypes are categorized and compared with respects to fundamental design decisions that are represented by asking how valid time is modeled, how transaction time is modeled, how attribute values are modeled, whether the model is all the same, and whether the model stick together as a whole. These five questions represent the criteria of evaluating temporal data model according to TSQL2 [9]. To generalize the approaches of modeling time-varying database, the taxonomy shown in Fig. 1 depicts that. *TTHR* is a combination of tuple timestamping and attribute timestamping data prototypes. Whereby the facts are timestamped by the lifespan of the associated entities, and the time varying attributes of these entities are timestamped by the valid time as explained in the previous section. *TTHR* is 1NF, heterogeneous, and interval-based temporal data model which is dedicated for modeling temporal database in relational framework. In this part, the modeling of *TTHR* in *TTSR* is scrutinized. For each approach/model, we stipulate the items identified in the model, present the associations to/from *TTHR* model to show that the same data are being stocked without losing any information. The study is progresses from the different approaches that are based on 1NF prototypes, and exclude N1NF prototypes which are not in the scope of this work. *TTSR* and *TTMR* approaches shown admit different representations, this work chose the representations explained in [18, 2] which are more close to the used time-model and easy to be implemented in conventional DBMS for the purpose of comparative analysis study. In *TTSR* the associations are reproduced by snapshot relations, which are 1NF associations. In below an illustration on how to embody *TTHR* associations in the *TTSR* representation is shown. Let R_{TTSR} stand for the association in the *TTSR* model that has the diagram representation $R_{TTSR} = (A_K, A_U, A_C, T_{ls}, T_{le}, T_{vs}, T_{ve})$ the corresponding relations in the *TTHR* representation are $R_T = (A_K, A_U, A_C, T_{ls}, T_{le})$ for current valid data and $VT\_R_T = (A_K, index, \alpha, T_{vs}, T_{ve})$ for historical changes of time-varying data. Since the information prototypes characterize associations in a different way and, for clearness and regularity of details, a correlation examples in tabular form is provide.

C. Representing Time-Vaying Data using TTMR Approaches

In *TTMR* the associations are accounted for by snap relation $R_{TTMR} = (A_K, A_U, T_{ls}, T_{le})$, for each temporal characteristic there are distinct associations $R_{A_{c1}} = (A_K, A_{C1}, T_{vs}, T_{ve})$ … $R_{A_{ci}} = (A_K, A_{Ci}, T_{vs}, T_{ve})$, and for the lifespan time $R_{LS} = (A_K, T_{ls}, T_{le})$, which are all in 1NF relations. Below describes how to represent *TTHR* relations in *TTMR* representation. The corresponding relations in *TTHR* representation are $R_T = (A_K, A_U, A_C, T_{ls}, T_{le})$ for current valid data and $VT\_R_T = (A_K, index, \alpha, T_{vs}, T_{ve})$ for historical changes of time-varying data. Since the information prototypes represent associations in a different way and, for both transparency and constancy of details the relation examples is shown in tabular form. A time-varying representation *patients* in Fig. 2, parallel to R_{TTMR} in *TTSR* is divided into $(i+2)$ associations, where *i* is equal to 19 (number of time-varying attributes) and the 2 other associations are the lifetime association, and the association that entails the non-temporal characteristics. The 19 associations parallel to each temporal value will be employed to register the historical changes of the validity of the temporal characteristics in *patients*. The lifetime correlation will be employed to trace the alterations of the lifetime of the items in *patients* relation and ultimately the non-time varying association is employed to register the non-temporal characteristics.

IV. Comparison of Temporal Data Models

This part will match up the three methods/prototypes in terms of information illustration and memory storage evaluation, a similar work done in [23] using different parameters. The start will be with the evaluation in terms of memory storage Point of View, Let *R* be a time-varying associational representation with a set of attributes $\{A_1, A_2, \cdots, A_n, T\}$, where these attributes can be categorized into 4 categories: ***k***ey, non-temporal (***U***nchangeable), temporal (***C***hangeable), and ***T***imestamps. They are represented by ***K***, ***U***, ***C***, and ***T*** correspondingly. Consequently, the representation of time-varying association can be re-represented as $\{A_K, A_U, A_C, A_T\}$, where $A_K = \{A_{K1}, A_{K2}, \cdots, A_{Kj}\}$, $A_U = \{A_{u1}, A_{u2}, \cdots, A_{un}\}$, $A_C = \{A_{c1}, A_{c2}, \cdots, A_{ci}\}$ and $A_T = \{A_{T1}, A_{T2}\}$ The timestamp attributes are defined as $A_{T1} = T^{c1}$, $A_{T2} = T^{c2}$. The subscripts variables *j*, *n*, and *i* represent the total number of key attributes (A_K), total number of Time-invariant attributes (A_U) and total number of Time-varying attributes (A_C), respectively.

Definition 1 (Non-temporal characteristic), A characteristic the value of which does not alter with time, a Non-temporal characteristic can be reorganized as in the case of an inaccuracy, but temporal information record does not maintain a record of it.

Definition 2 (Temporal characteristic), A temporal characteristic is a feature the merits of which are connected with timestamps.

Definition 3 (Timestamp), A timestamp is a temporal worth correlated with a Timestamped item (i.e., a feature worth or tuple).

Definition 4 (Lifespan), the lifetime of record items is the moment in the course of which the item is outlined.

Definition 5 *(Frequency of Time-varying attribute* $f(A_{cm})$, the amount of occurrence rate this characteristic is to be altered (changed). Where $A_C = \{A_{c1}, A_{c2}, \cdots, A_{cm}, \cdots, A_{ci}\}$ and $m \in \{1, 2, \ldots, i\}$. Thus, a function f defined on A_C in such a way that $f(A_{cm})$ which returns the frequency of times A_{cm} will be changing.

Definition 6 ($S(A_{\chi\gamma})$), A role to be determined on all the characteristics of R , where $S(A_{\chi\gamma})$ reverts the magnitude of a characteristic $A_{\chi\gamma}$ in bytes, and $\chi \in \{K, U, C, T\}$ and $\gamma \in \{1,2,\ldots,j\}$ (key attributes), or $\gamma \in \{1,2,\ldots,n\}$ (non-temporal features) or $\gamma \in \{1,2,\ldots,i\}$ (temporal characteristics) or $\gamma \in \{1,2\}$ (timestamping traits) for all attributes sets that construct R .

Definition 7 $Cost(A_\chi)$, A role to be determined on the subclass characteristics χ , where $\chi \in \{K, U, C, T\}$. $Cost(A_\chi)$ reverts the total of all characteristics amount in A_χ in bytes.

Definition 8 $Cost(z)$, the cost of a tuple (line) z in relation example r_t is the total of the cost of all sub-group characteristics equivalent to $\cos t(A_K) + \cos t(A_U) + \cos t(A_C) + \cos t(A_T)$.

Definitions 9 The cost of a different attribute type is defined as:

$$\cos t(A_k) = \sum\nolimits_{i=1}^{j} \cos t(A_{ki}) = K\ byte \tag{1}$$

$$\cos t(A_u) = \sum\nolimits_{i=1}^{n} \cos t(A_{ui}) = U\ byte \tag{2}$$

$$\cos t(A_c) = \sum\nolimits_{m=1}^{i} \cos t(A_{cm}) = C\ byte \tag{3}$$

$$\cos t(A_T) = \sum\nolimits_{i=1}^{2} \cos t(A_{Ti}) = T\ byte \tag{5}$$

Definitions 10 The Frequency time of all time varying attributes $A_{cm} \in \{A_{c1}, A_{c2}, \cdots, A_{ci}\}$ in a time span λ can be computed as:

$$f(A_c) = \sum\nolimits_{m=1}^{i} f(A_{cm}) = \delta\ times \tag{5}$$

The evaluation of the *TTHR* in terms of memory storage used for physical implementation in conventional DBMS should encounter the same issues for the other time-varying information prototype. Since representing time-varying information record in a co-relational structure varies in numerous proportions [9]. Since the second approach (N1NF) may not be able of straightforwardly employing a correlation stocking structure or a uncertainty appraisal technique which depends on minute characteristic merits [9], then the temporal database prototypes that use this approach will not be counted in this comparison. The storage point of view in the *TTHR* will be compared with temporal database prototypes that use tuple timestamping with 1NF. In contrast with the employed storage in diverse prototypes, a predetermined time-span not across records for the information record file construction model was applied in this study. Assume $R=\{A_K, A_U, A_C, A_T\}$ as defined previously, we then use the different approaches and compare them with the *TTHR* model, a similar study has been carried out in [7].

For TTSR model, the time-varying association in *TTSR* can be modeled as:

R_{TTSR} ($A_{k1}, \cdots, A_{ki}$, $A_{u1}, \cdots, A_{un}$, $A_{c1}, \cdots, A_{cm}$, A_T), The cost of modeling one tuple x in the association example $r(R_{TTSR})$ can be computed as:

$$\text{Cost}(x)=\text{cost}(A_k)+\text{cost}(A_u)+\text{cost}(A_c)+\text{cost}(A_T)$$
$$=K+U+C+T\ byte \quad ,$$

as indicated in (1), (2), (3) and (4).

The cost of stocking the record of the alterations of A_c with $f(A_c)=\delta$ frequency in a time span λ can be computed as:

$$=\delta(K+U+C+T) \qquad (6)$$

A change in any A_c needs the inclusion of a new line with the entire values.

In the *TTHR* prototype, the time-varying association scheme is accounted for by R_{TTHR} and R_{VT} as indicated underneath:

R_{TTHR} ($A_{k1}, \cdots, A_{ki}$, $A_{u1}, \cdots, A_{un}$, $A_{c1}, \cdots, A_{cm}$, A_T), R_{VT} ($A_{k1}, \cdots, A_{ki}$, $Index, \alpha$, A_T). The cost of modeling one tuple x in relation example $r(R_{TTHR})$ can be computed as:

$$\text{Cost}(x)=\text{cost}(A_k)+\text{cost}(A_u)+\text{cost}(A_c)+\text{cost}(A_T)$$
$$=K+U+C+T\ byte \qquad (7)$$

As indicated in (1), (2), (3) and (4). The cost of stocking up the record of the alterations of A_c with $f(A_c)=\delta$ occurrences in a time span λ can be computed as:

$$=\delta(K+index+\alpha+T) \Rightarrow$$
$$=\delta(K+1+\alpha+T) \qquad (8)$$

Index: is a new feature to list the temporal values with one byte range:

α : is a novel extra value of an alternative information category to seize information from a diverse type. Its range is supposed to be equal to the range of the biggest domain range in A_c . The range of α in byte is $S(\alpha)=Max(S(A_{c1}), S(A_{c2}), \cdots, S(A_{ci}))$. Since every change in any A_c needs introducing a novel line in R_{VT} for the previous worth of the brought about (altered) temporal values, the sum is stored in a memory bank (room) for *TTHR* over *TTSR* for a definite time span λ can be computed as is shown below:

From (6), $\text{Cost}(TTSR)=(K+U+C+T)+\delta(K+U+C+T)$

From (7) and (8),

$$\text{Cost}(TTHR)=(K+U+C+T)+\delta(K+1+\alpha+T)$$

Cost (improvement)

$$=\frac{\text{Cost}(TTSR)-\text{Cost}(TTHR)}{\text{Cost}(TTSR)}$$
$$=\frac{K+U+C+T+\delta(K+U+C+T)-(K+U+C+T+\delta(K+1+\alpha+T))}{K+U+C+T+\delta(K+U+C+T)}$$
$$=\frac{\delta(K+U+C+T)-\delta(K+1+\alpha+T)}{(K+U+C+T)(1+\delta)}$$
$$=\frac{\delta(U+C-1-\alpha)}{(K+U+C+T)(1+\delta)}$$

Since $\alpha >> 1$ then cost (improvement)

$$\approx\frac{U+C-\alpha}{K+U+C+T} \qquad (9)$$

Assume $Q=U+C$ then (9) will be:

$$\approx\frac{Q-\alpha}{K+Q+T}$$

For TTMR approach, the spared space in the storeroom for the *TTHR* over the *TTSR* would be directly proportional to $Q=U+C$ for a very large value of Q comparing to $K, T, and \alpha$ since this is due to the repeated data in *TTSR*.

The time-varying association in *TTMR* is decomposed and reproduced as:

R_{TTMR} ($A_{k1}, \cdots, A_{kj}$, $A_{u1}, \cdots, A_{un}$)

R_{Ac1} ($A_{k1}, \cdots, A_{kj}$, A_{c1} , A_T)

R_{Ac2} ($A_{k1}, \cdots, A_{kj}$, A_{c2} , A_T)

R_{Ac3} ($A_{k1}, \cdots, A_{kj}$, A_{c3} , A_T)

R_{Ac4} ($A_{k1}, \cdots, A_{kj}$, A_{c4} , A_T)

R_{Aci} $(A_{k1}, \cdots, A_{kj} , A_{ci} , A_T)$

The rate of stocking one tuple x in an association example $r(R_{TTMR})$ can be computed as follows:

From (1), (2), (3) and (4),

$$\begin{aligned}\text{Cost}(x) &= \text{Cost}(A_k) + \text{Cost}(A_u) + \text{Cost}(A_T) \\ &+ \sum_{m=1}^{i} \left[S(A_{cm}) + \text{Cost}(A_k) + \text{Cost}(A_T) \right] \\ &= K + U + T + \sum_{m=1}^{i} S(A_{cm}) + \sum_{m=1}^{i} K + \sum_{m=1}^{i} T \\ &= (K + U + T) + i(K + T) + \sum_{m=1}^{i} S(A_{cm})\end{aligned}$$

Since

$$\begin{aligned}\text{Cost}(A_c) &= i(K+T) + \sum_{m=1}^{i} S(A_{cm}) \\ &= i(K+T) + C\end{aligned}$$

Then the $\text{Cost}(x)$ can be reproduced as:

$$\text{Cost}(x) = K + U + T + C + i(K+T)$$

$$= K(i+1) + U + C + T(i+1) \quad byte \tag{10}$$

Where i is the total number of A_c .The rate of stocking the record of the updates of each A_{cm} with $f(A_{cm}) = \delta_m$ frequencies in a time span λ can be computed as:

$$= \sum_{m=1}^{i} \delta_m \left(S(A_{cm}) + K + T \right)$$

$$= \sum_{m=1}^{i} \delta_m S(A_{cm}) + (K+T) \sum_{m=1}^{i} \delta_m$$

Since form (5)

$\sum_{m=1}^{i} \delta_m = \delta$, then

$$= \sum_{m=1}^{i} \delta_m S(A_{cm}) + \delta(K+T) \tag{11}$$

The improvement of the *TTHR* over the *TTMR* in terms of storage point of view can be calculated as follows:

The cost of storing one tuple x with its history of the changes in a period of time λ using the *TTHR* and the *TTMR* can be calculated as:

From (7) and (8),

$$\text{Cost}(TTHR) = (K + U + C + T) + \delta(K + 1 + \alpha + T)$$

From (10) and (11),

$$\text{Cost}(TTMR) = K(i+1) + U + C + T(i+1) + \sum_{m=1}^{i} \delta_m S(A_{cm}) + \delta(K+T)$$

Cost (improvement)

$$= \frac{\text{Cost}(TTMR) - \text{Cost}(TTHR)}{\text{Cost}(TTMR)}$$

$$= \frac{i(K+T) + \sum_{m=1}^{i} \delta_m S(A_{cm}) - \delta(1+\alpha)}{iK + k + U + C + iT + T + \sum_{m=1}^{i} \delta_m S(A_{cm}) + \delta K + \delta T}$$

$$= \frac{iK + K + U + C + iT + T + \sum_{m=1}^{i} \delta_m S(A_{cm}) + \delta K + \delta T - K - U - C - T - \delta K - \delta - \delta\alpha - \delta T}{K(i+1) + U + C + T(i+1) + \sum_{m=1}^{i} \delta_m S(A_{cm}) + \delta(K+T)}$$

$$= \frac{iK + iT + \sum_{m=1}^{i} \delta_m S(A_{cm}) - \delta - \delta\alpha}{K(i+1) + U + C + T(i+1) + \sum_{m=1}^{i} \delta_m S(A_{cm}) + \delta(K+T)}$$

$$= \frac{i(K+T) + \sum_{m=1}^{i} \delta_m S(A_{cm}) - \delta(1+\alpha)}{K(i+1+\delta) + U + C + T(i+1+\delta) + \sum_{m=1}^{i} \delta_m S(A_{cm})}$$

Since $\alpha >> 1$ and $\delta >> 1$ then the equations will be

$$= \frac{i(K+T) + \sum_{m=1}^{i} \delta_m S(A_{cm}) - \delta\alpha}{K(i+\delta) + U + C + T(i+\delta) + \sum_{m=1}^{i} \delta_m S(A_{cm})} \tag{12}$$

Let $\omega = \sum_{m=1}^{i} \delta_m (S(A_{cm}))$ then $\omega \approx \delta\alpha$ when the size $S(A_{cm})$ of all $A_{cm} \in \{A_{c1}, A_{c2}, \cdots, A_{ci}\}$ is almost the same. Hence, (12) can be written as:

$$= \frac{i(K+T)}{K(i+\delta) + U + C + T(i+\delta) + \omega} \tag{13}$$

The save in storage space for the TTHR over the TTMR would be dependable on the values of K , T , and i , if the term $i(K+T)$ is very large compared to ω, then there will be an improvement in the TTHR over the TTMR, otherwise the TTMR has more saving in memory storage over the TTSR specially for a very large value of δ . In this paper, however, The TTHR model is proposed to overcome the complexity of the TTMR model in terms of implementation and query processing. For example, if the number of time-varying attributes i is very large which is normal in temporal database application specially the clinical database applications where very large number of attributes that can vary within a time. Suppose $i = 50$, then a need to create 50 tables, 50 index for the primary key, 50 integrity constraints, and etc. This for one object, if the database schema has more than one temporal object, then the same thing will be repeated, which would result in difficulty to manage the database schema. In addition to that, combing the temporal data of a specific object requires a distinction of relationship recognized as time-varying junction relationship, which is usually costly to put into practice. A parallel research for computing the save in a memory stocking room for the TTMR over the TTSR has been done in [4]. Based on the mathematical equations, several experiments have been carried out; the results explained in more details in our previous work in [18, 25-28]. Several experiments have been carried out at different cost of A_c and freezing $f(A_c)$ at different values, yielded the results that show as saving in memory space range from 80%-90%.

V. Conclusion

TTHR data model has accomplished a saving in memory storage as well as the easy implementation and retrieval of time-varying data in RDBMS. These issues are important in stocking memory space and that is approximately equivalent to or larger than the TTMR. The introduced time-varying information representation is proposed for its straightforwardness, because less information record items will be required to grasp the time-varying features of the temporal information comparing to the TTMR. Furthermore, implementing the TTHR to an available information record application does not necessitate numerous updates in comparison to the TTMR. Other than that, simply have to construct the supplementary association to size the chronological updates of temporal characteristics, but without changing the scheme itself. This contrasts to the TTMR as it needs to decompose the relations and redefine the integrity constraints. A case study for modeling biomedical data (diabetes) has been shown and can be implemented in any RDBMS with a saving in memory usage range from 68-81% over other temporal representations.

Acknowledgment

This project was funded by the Deanship of Scientific Research (DSR) under grant no. G- 27-005-1430. The authors, therefore, acknowledge with thanks to DSR’s technical and financial support.

References

[1] Findler, N. V., & Chen, D. (1973). On the problems of time retrieval of temporal relations causality, and coexistence. International Journal of Computer & Information Sciences, 2, 3, 161-185.

[2] Snodgrass R (1999) Temporal Data Management, IEEE Transactions on Knowledge and Data Engineering, Vol. 11, No. 1, January/February 1999, pp. 36–44).

[3] Date, C. D., Darwen, H., & Lorentzos, N. A. (2003). Temporal data and the relational data model. San Francisco: Morgan Kaufmann.

[4] Novikov, B. A., & Gorshkova, E. A. (2008). Temporal databases: From theory to applications. Programming and Computer Software, 34, 1, 1-6. Pleiades Publishing, Ltd., 2008. Original Russian Text

[5] Tansel, A. U. (2004). On handling time-varying data in the relational data model. Information and Software Technology, 46, 2, 119-126.

[6] Elmasri, R., and Navathe (2000). Fundamentals of Database Systems. 3rd edition. Addison Wesley.

[7] Jensen, C. S., Clifford, J., Gadia, S. K., Segev, A., & Snodgrass, R. T. (1992). A glossary of temporal database concepts. ACM Sigmod Record, 21, 3, 35-43.

[8] Snodgrass, R. T., (2000). Developing Time-Oriented Database Applications in SQL, 1st edition, Morgan Kaufmann Publishers, Inc., San Francisco.

[9] Jensen, C. S., Snodgrass, R. T., & Soo, M. D. (1995). The tsql2 data model (pp. 157-240). Springer US. http://people.cs.aau.dk/~csj/ Thesis/pdf/chapter12.pdf.

[10] Patel, J. (2003). Temporal Database System Individual Project. Department of Computing, Imperial College, University of London, Individual Project, 18-June-2003, http://www.doc.ic.ac.uk/~pjm/ teaching/student_projects/ jaymin_patel.pdf.

[11] Zimányi, E. (2006). Temporal aggregates and temporal universal quantification in standard SQL. ACM SIGMOD Record, 35, 2, 16-21.

[12] Wang, F., Zhou, X., & Zaniolo, C. (2006, April). Using XML to build efficient transaction-time temporal database systems on relational databases. In Proceedings of the 22nd International Conference on Data Engineering, 2006. ICDE'06 (pp. 131-131). IEEE.

[13] A-Qustaishat, M. (2001). A visual temporal object-oriented model embodied as an expert C++ Library. ADVANCES IN MODELLING AND ANALYSIS-D-,6, 3/4, 3-43.

[14] Bohlen, M. H., Busatto, R., & Jensen, C. S. (1998, February). Point-versus interval-based temporal data models. In Proceedings of 14th International Conference on Data Engineering, (pp. 192-200). IEEE.

[15] Dyreson, C., Grandi, F., Käfer, W., Kline, N., Lorentzos, N., Mitsopoulos, Y., ... & Wiederhold, G. (1994). A consensus glossary of temporal database concepts.ACM Sigmod Record, 23, 1, 52-64.

[16] Tansel, A. U. (2006). Modeling and Querying Temporal Data. Idea Group Inc.

[17] Tansel, A. U. (2004). Temporal data modeling and integrity constraints in relational databases. In Computer and Information Sciences-ISCIS 2004 (pp. 459-469). Springer Berlin Heidelberg.

[18] Halawani, S. M., & Romema, N. A. (2010). Memory storage issues of temporal database applications on relational database management systems. Journal of Computer Science, 6, 3, 296.

[19] Atay, C. (2016). An attribute or tuple timestamping in bitemporal relational databases. Turkish Journal of Electrical Engineering & Computer Sciences. (2016) 24: (pp. 4305 – 4321). doi:10.3906/elk-1403-39.

[20] Noh, S.Y., Gadia, S.K. and Jang, H., (2013). Comparisons of three data storage models in parametric temporal databases. Journal of Central South University, 20(7), pp.1919-1927.

[21] Kvet, M., Matiako, K. and Kvet, M., (2014). Transaction management in fully temporal system. In Computer Modelling and Simulation (UKSim), 2014 UKSim-AMSS 16th International Conference on (pp. 148-153). IEEE.

[22] Snodgrass R, Ahn I. Performance evaluation of a temporal database management system. Commun ACM 1986; 15:96-107.

[23] Arora, S. (2015). A comparative study on temporal database models: A survey. In Advanced Computing and Communication (ISACC), 2015 International Symposium on (pp. 161-167). IEEE.

[24] Anselma, L., Stantic, B., Terenziani, P., and Sattar, A. (2013). Querying now-relative data. Journal of Intelligent Information Systems, 41(2), 285-311.

[25] Halawani, S.M., AlBidewi, I., Ahmad, A.R. and Al-Romema, N.A., 2012. Retrieval optimization technique for tuple timestamp historical relation temporal data model. Journal of Computer Science, 8(2), p.243.

[26] Nashwan Alromema, Mohd Shafry Mohd Rahim and Ibrahim Albidewi, “A Mathematical Model for Comparing Memory Storage of Three Interval-Based Parametric Temporal Database Models” International Journal of Advanced Computer Science and Applications(ijacsa), 8(7), 2017. http://dx.doi.org/10.14569/IJACSA.2017.080741.

[27] Alromema, N.A., Rahim, M.S.M. and Albidewi, I., 2016. Temporal Database Models Validation and Verification using Mapping Methodology. VFAST Transactions on Software Engineering, 11(2), pp.15-26.

[28] Ab Rahman Ahmad, N.A., Rahim, M.S.M. and Albidewi, I., 2015. Temporal Database: An Approach for Modeling and Implementation in Relational Data Model. Life Science Journal, 12(3).

[29] Jones S, Mason P, Stamper R (1979) A Relational Specification Language for Complex Rules. Information Systems, 4(4):293{305, November 1979}.

[30] Navathe S, Ahmed R (1989) A Temporal Relational Model and a Query Language. Information Sciences, 49:147{175, 1989}.

[31] Ariav G (1986) A Temporally Oriented Data Model. ACM Transactions on Database Systems, 11(4):499{527, December 1986}.

[32] Sarda N (1990) Extensions to SQL for Historical Databases. IEEE Transactions on Knowledge and Data Engineering, 2(2):220{230, June 1990}.

[33] Snodgrass R (1987) The Temporal Query Language TQUEL. ACM Transactions on Database Systems, 12(2):247{298, June 1987}.

[34] Noh S, Gadia S (2008) Benchmarking temporal database prototypes with interval-based and temporal element-based timestamping. Journal of Systems and Software, 81(11):1931–1943.

[35] Gadia S, (1992) A Seamless Generic Extension of SQL for Querying Temporal Data. Technical Report TR-92-02, Computer Science Department, Iowa State University, May 1992.

[36] https://www.sophe.org/wp-content/uploads/2017/01/Diabetes_toolkitrevised.pdf.

[37] https://www.joslin.org/research/diabetes-research-center.