Artificial Intelligence in Medicine 49 (2010) 11–31

Contents lists available at ScienceDirect

Artificial Intelligence in Medicine

journal homepage: www.elsevier.com/locate/aiim

Intelligent visualization and exploration of time-oriented data of multiple patients

Denis Klimov *, Yuval Shahar, Meirav Taieb-Maimon

Department of Information Systems Engineering, Faculty of Engineering Sciences, Ben-Gurion University of the Negev, P.O. Box 653, Beer-Sheva 84105, Israel

ARTICLE INFO

Article history:
Received 20 August 2008
Received in revised form 31 January 2010
Accepted 16 February 2010

Keywords:
Multiple patients
Intelligent user interfaces
Temporal databases
Temporal abstraction
Knowledge-based systems
Information visualization
Interactive data mining
Oncology

ABSTRACT

Objective: Clinicians and medical researchers alike require useful, intuitive, and intelligent tools to process large amounts of time-oriented multiple-patient data from multiple sources. For analyzing the results of clinical trials or for quality assessment purposes, an aggregated view of a group of patients is often required. To meet this need, we designed and developed the *VISualizatIon of Time-Oriented RecordS* (VISITORS) system, which combines intelligent temporal analysis and information visualization techniques. The VISITORS system includes tools for intelligent retrieval, visualization, exploration, and analysis of raw time-oriented data and derived (abstracted) concepts for multiple patient records. To derive meaningful interpretations from raw time-oriented data (known as *temporal abstractions*), we used the knowledge-based temporal-abstraction method.
Methods: The main module of the VISITORS system is an interactive, ontology-based exploration module, which enables the user to visualize raw data and abstract (derived) concepts for multiple patient records, at several levels of temporal granularity; to explore these concepts; and to display associations among raw and abstract concepts. A knowledge-based delegate function is used to convert multiple data points into one delegate value representing each temporal granule. To select the population of patients to explore, the VISITORS system includes an ontology-based temporal-aggregation specification language and a graphical expression-specification module. The expressions, applied by an external temporal mediator, retrieve a list of patients, a list of relevant time intervals, and a list of time-oriented patients' data sets, by using an expressive set of time and value constraints.
Results: Functionality and usability evaluation of the interactive exploration module was performed on a database of more than 1000 oncology patients by a group of 10 users—five clinicians and five medical informaticians. Both types of users were able in a short time (mean of 2.5 ± 0.2 min per question) to answer a set of clinical questions, including questions that require the use of specialized operators for finding associations among derived temporal abstractions, with high accuracy (mean of 98.7 ± 2.4 on a predefined scale from 0 to 100). There were no significant differences between the response times and between accuracy levels of the exploration of the data using different time lines, i.e., absolute (i.e., calendrical) versus relative (referring to some clinical key event). A system usability scale (SUS) questionnaire filled out by the users demonstrated the VISITORS system to be usable (mean score for the overall group: 69.3), but the clinicians' usability assessment was significantly lower than that of the medical informaticians.
Conclusions: We conclude that intelligent visualization and exploration of longitudinal data of multiple patients with the VISITORS system is feasible, functional, and usable.

1. Introduction: intelligent visualization of time-oriented data of multiple patients

A key task facing clinicians and medical researchers is the analysis of time-stamped, longitudinal medical records, particularly, records of multiple patients. This capability is necessary to support, for example, quality assessment tasks, analysis of clinical trials, and the discovery of new clinical knowledge. Although the task of accessing patient data has been solved mostly through the increasing use of electronic medical record (EMR) systems, there still remains the task of intelligent processing of time-oriented records of multiple patients, including the capability for interactive exploration of the results. For this task, standard means, such as tables, time-oriented statistical tools, or even more advanced temporal data-mining techniques, are often not adequate, since their use may demand

* Corresponding author at: Medical Informatics Research Center, Department of Information Systems Engineering, Ben-Gurion University of the Negev, P.O. Box 653, Beer-Sheva 84105, Israel. Tel.: +972 8 6477160; fax: +972 8 6477161.
E-mail address: klimov@bgu.ac.il (D. Klimov).

0933-3657/$ – see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.artmed.2010.02.001

specialized, advanced knowledge, or they may be applicable only in particular cases.

To derive meaningful patterns and interpretations, known as *temporal abstractions* (or *abstract concepts*), from raw time-oriented patient data, we use a *knowledge-based temporal-abstraction* (KBTA) method [1]. Through a domain-specific temporal-abstraction knowledge base acquired from a domain expert, this method derives interval-based temporal abstractions, for example, the pattern, "a period of more than two months of grade 1 or higher bone-marrow toxicity, followed within three months by a decrease in liver-function" (these concepts are defined in the context of a particular oncology therapy protocol).

The temporal abstractions computed by the KBTA method for an *individual* patient or small number of patients can be efficiently visualized through an ontology-driven interface that we developed previously, known as *KNAVE-II* [2], which has been shown to be functional and usable [3]. However, analysis of the data for large patient *populations*, such as clinical trial results or quality assessments of clinical-management processes, requires a new tool that provides aggregate views of time-oriented data and abstractions of *groups* of patient records. In addition, certain patterns can be discovered only through the analysis of multiple patient records. Therefore, as part of the current study, we designed and developed a greatly enhanced extension of the KNAVE-II system, designated the *VISualization of Time-Oriented RecordS* (VISITORS) system, which combines intelligent temporal reasoning computational mechanisms with information visualization methods for display and exploration of time-oriented records of multiple patients. Fig. 1 presents an overview of the main interface of raw and derived concepts and the semantics of VISITORS.

Furthermore, the VISITORS system also enables users to interactively specify temporal and knowledge-based constraints, through a graphical expression-specification module, which enables users to define the patient subsets selected for exploration (e.g., the lists of patients displayed in panel A of Fig. 1). Underlying the expression-specification module is the *ontology-based temporal-aggregation* (OBTAIN) specification language, which includes a set of operators and constraints that enable unsophisticated users to graphically construct (i.e., to specify) three types of expression: *Select Patients* (Who?), *Select Time Intervals* (When?) and *Get Patient Data* (What?). The goal of these expressions is to retrieve a list of patients, a list of relevant time intervals, or a list of time-oriented patient data sets, respectively; combinations of these lists may be manipulated by the user according to the patients, time periods and/or data values that are to be analyzed further. A full exposition of the OBTAIN language and the graphical expression-specification module is beyond the scope of this study and can be found elsewhere [4].

Aggregation of the longitudinal data of raw and abstract concepts of a group of patients is another unique aspect of the VISITORS system. We have defined and implemented in VISITORS the "*delegate value*" method, namely, given a single patient's time-oriented data for a specific concept (raw or abstract) over a specific time interval (including a predefined granularity level), we calculate the *delegate value* of an *individual* patient's data at each time granule (or for another specific time period) using a function specific to the concept and the temporal granularity. For example, assume that on 1 January 2007 there were three Platelet values for a particular patient: 17,700 cells/ml at 5 a.m., 38,900 cells/ml at 11 a.m., and 43,250 cells/ml at 8 p.m. Thus, if we select the *mean* as the

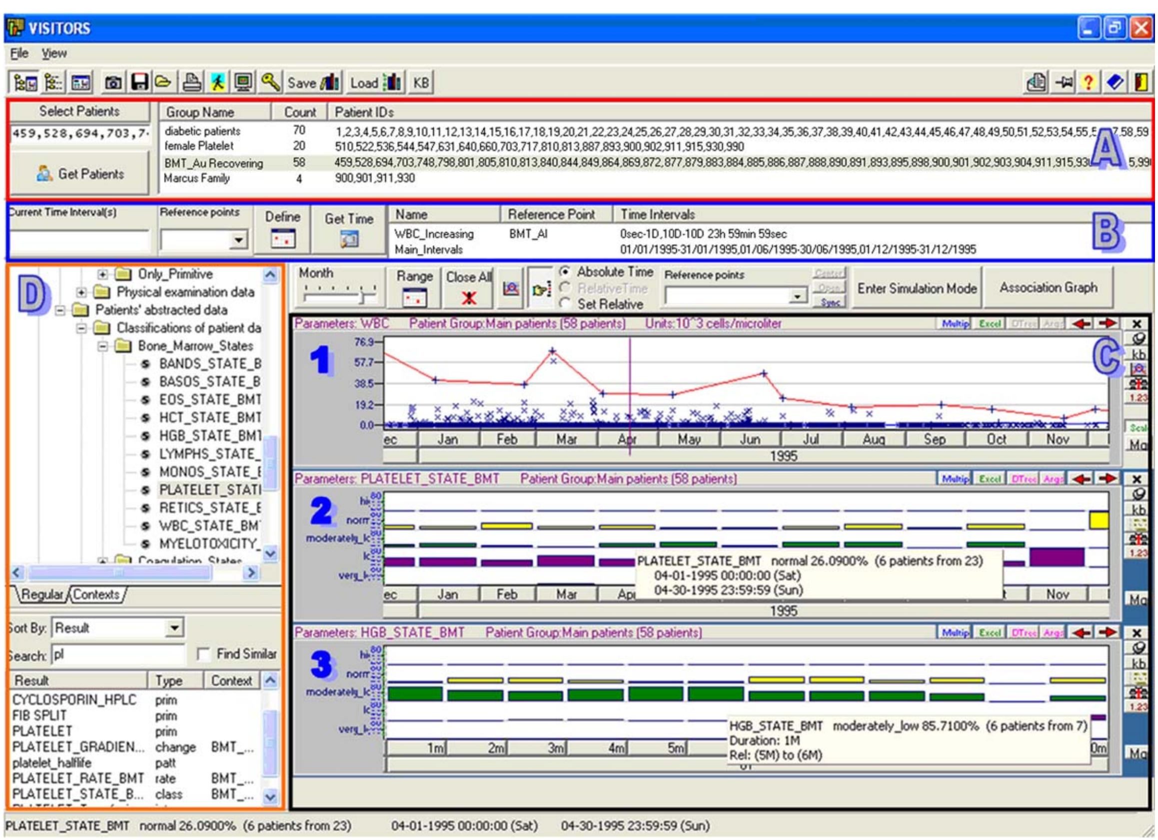

Fig. 1. The VISITORS main interface, this case in the hematological oncology domain. The two top panels display lists of patients (denoted by A) and lists of time intervals (denoted by B), retrieved by computing the previous population-expressions. The graphs (denoted by C) show the data for a group of 58 patients for the White blood cell (WBC) count raw concept (graph 1) and for the monthly distribution of the values of the Platelet-state abstract concept during 1995 (graph 2). Graph 3 shows the monthly distribution of the values of the Hemoglobin (HGB)-state concept during the first year (relative time line) following BMT (see Sections 4.4 and 4.5 further details of the visualization and exploration operators). The left panel (denoted by D) of the interface includes a knowledge-based browser showing the domain's ontology.

delegate (representative) function, then the daily average Platelet count for that patient was 33,283 cells/ml. However, during the exploration time, the user can choose any other suitable delegate function (such as *mode* or *maximum*).

The *population delegate value* represents the aggregated value of a group of patients calculated by the concept- and the temporal granularity-specific statistical function within a specific time granule (or for another specific time period), e.g., the maximal value of the raw Hemoglobin values for a group of patients over an entire month. This function can also be manipulated at will.

The following features distinguish the VISITORS system from other data exploration tools:

- The system provides a *5-step* iterative loop for intelligent investigation of multiple time-oriented patient records: (1) specification, (2) retrieval, (3) visualization, (4) interactive exploration, and (5) knowledge-based temporal analysis.
- Time-oriented data are graphically displayed and explored in an intuitively similar fashion for both *individual* and *multiple* patient records.
- Particular consideration is given to the *temporal* aspect of the conceptual and graphical representations: The data can be aggregated at and explored within various temporal granularities, such as hour, day, and month. Support is also provided for a calendrical (absolute) timeline and for a timeline relative to a special event [e.g., the months following a bone-marrow transplantation (BMT) event], or to another clinically significant time point (e.g., start of high fever).
- The computational reasoning supports not only a view of *raw* time-oriented data and its statistics but also a meaningful display of various *knowledge-based interpretations* of the raw data, based on temporal-abstraction domain ontology, the KBTA computational mechanisms, and specialized time-oriented aggregation operators. The exploration interface is also based on the same ontology, which supports a semantic exploration of the data (using semantic relations such as "derived from" or "part of") and enables navigation of semantically related raw and abstract concepts.

The first two steps of the iterative loop, i.e., specification and retrieval of the multiple time-oriented patient records, are described in detail in another study [4]. In the current study, we focus on the other three capabilities: visualization, interactive exploration, and knowledge-based temporal analysis of multiple patient records.

The main contributions of the current study lie in: (1) providing formal definitions for the operators that perform visualization, exploration, and knowledge-based temporal analysis of longitudinal data for multiple patients; (2) implementation of a complete architecture developed using formal definitions; and (3) a functionality and usability evaluation of the implemented system.

2. Related work

2.1. Combining domain knowledge, temporal abstraction and information visualization in medical informatics

The use of a domain knowledge base can both support an in-depth analysis of longitudinal patient records and simplify and facilitate the data exploration process, since the user can explore only high-level concepts based on complex temporal patterns (or, in general, on any abstract concepts) previously defined in a domain-specific knowledge base and detected in the patients' data. In addition, it has been demonstrated that visual representation can often communicate information much more rapidly and effectively than any other method [5]. Thus, the combination of these two approaches could significantly improve the exploration of patient data, as has been suggested by a clinical survey [6].

In the current study, using a domain-specific knowledge-base, we applied the KBTA method [1] for automated derivation of meaningful context-specific interpretations and conclusions (temporal abstractions) from raw time-oriented patient data. In general, the KBTA method may be described as follows: the input includes a set of time-stamped measurable concepts (e.g., Platelet count, Red blood cell count) and external events (e.g., bone-marrow transplantation). The events typically create the necessary interpretation contexts (e.g., the therapy protocol used), which could change the interpretations of the data. The output includes a set of interval-based, context-specific concepts at the same or a higher level of abstraction and their respective values (e.g., a period of two months of grade 1 bone-marrow toxicity in the context of particular chemotherapy protocol).

Other studies have investigated different techniques for temporal abstraction. Silvent et al. [7], for example, proposed combining temporal data abstraction techniques with data mining approaches, a method that would update the prior domain knowledge. The temporal-abstraction mechanisms put forward by Miksch et al. [8] do not require predefined domain knowledge and can process high-frequency temporal quantitative and qualitative data. Another method for calculation of temporal abstractions has been applied to ECG data by using a fast Fourier transform [9,10]. In that method, a curve made up of a series of data points was transformed to a set of bends and lines in between these data points. However, the above frameworks concentrate on the methodology of the temporal abstractions rather than on visualization and exploration issues. Moreover, the interfaces used do not fulfill most of the desiderata defined below (in Section 3) for exploration of longitudinal patient data.

Unlike the KBTA method, in which the domain expert defines the temporal patterns for temporal abstraction derivations, Combi and Chittaro [11] used, as the temporal ontology, an object-oriented model based on the calculus of events [12]. They validated their approach by building the CARDIOlogic Temporal Abstraction System (CARDIOTABS) for the abstraction of temporal cardiology data, which supports nonvisual construction of queries regarding patient data and very simple graphical interfaces for patient data visualization. They did not, however, evaluate the usability of the interfaces.

To date, most visual exploration systems in medicine have focused solely on the visualization and exploration of static raw patient data, as reviewed by Chittaro [13]. The LifeLines project [14,15], for example, provides a general visualization environment for personal histories, which can be applied to medical and court records, professional histories and other types of biographical data. The extended version, LifeLine2 [16], includes the option of aligning the patient data according to key medical events, i.e., a relative timeline view.

In the area of visualization of data for multiple patients, a number of systems have been developed. The InfoZoom system [17] uses a novel technique to display data sets as a highly compressed table that always fits completely onto the screen. The overall goal of the MedView project [18] is to develop models, methods and tools to support clinicians in their daily diagnostic work. As part of MedView, two information visualization tools were developed and tested as a solution to the problem of visualizing insights derived from large amounts of clinical data. The first tool, *The Cube*, enables interactive recognition of patient patterns through a 3D display of a set of 2D parallel diagrams (each using a horizontal time axis and a vertical value axis), where each diagram represents a single patient attribute (e.g., diagnoses, allergies). Thus, patients who have similar values for several attributes might have parallel lines connecting the different 2D diagrams. The second tool, SimVis, is based on a similarity assessment-based interaction model for exploring data; the tool was designed to help clinicians to classify and cluster clinical-test data. A third system, the Interactive Parallel Bar Charts (IPBC) system [19] adopts 3D bar charts as its basic visualization technique and augments them with several interactive features, thereby exploiting 3D space to significantly increase both the number of time-series that can be simultaneously analyzed in a convenient way and the number of values associated with each series.

As noted earlier, the above exploration systems focus mostly on the visualization of raw longitudinal data or only partially support high-level meaningful interpretations of these data as abstract clinical concepts. There is, however, a growing realization of the importance of integration of the visual, analytical and user-centered methods, as is shown in the review of Aigner et al. [20] of different visualizations of time-oriented data, in which the exploration task is solved by providing complex integrated interfaces. However, we view the integration somewhat differently: the visualization must be simple and must represent the results of a complex temporal reasoning computational process, namely, various types of users must be able to explore the data in an equally efficient fashion. Moreover, such integration must be modular, accessible, and independent of a particular medical domain.

To address the issues discussed above, we devoted the current study to extending our previously developed KNAVE-II system [2] for visualization and exploration of individual patient data to support the visualization and exploration of the time-oriented records of multiple patients. This effort required significant modification of the original tools, as explained in the Methods section below.

2.2. Other temporal aspects relevant to the study

2.2.1. Temporal and visual-temporal query languages

Recall that the OBTAIN specification language and its graphical expression-specification tool are used by end-users to specify the relevant patients or time periods to be retrieved and explored. Although the OBTAIN specification language is *not* intended to be, in any way, a general *temporal query language*, the OBTAIN expressions specified by users could be considered as types of *queries*, since their goal is to retrieve a set of patients, time-intervals, or time-oriented data. Thus, it is relevant to compare the OBTAIN language to existing temporal (visual) query languages.

The main difference between the OBTAIN specification language and the previous pure time-oriented query languages, such as TSQL [21], HSQL [22], TSQL2 [23], GCH-OSQL [24–26], TLSQL [27] and most recently t4sql [28,29], is that according to these languages SQL queries are applied directly to raw-data databases; thus, the expressivity of the language is limited by the pure SQL and by the temporal-extension capabilities of these languages. In contrast, a broad set of temporal [knowledge-based] constraints for specification of either patients or time intervals can be used in OBTAIN expressions, since to answer the OBTAIN expressions, additional computational modules (such as the module for calculation of delegate values that is described in the Methods section), and a runtime capability for computation, on the fly, of knowledge-based temporal abstractions, are provided. Thus, on one hand, the semantic expressivity of the OBTAIN language, according to our desiderata, is enhanced relative to standard query languages. On the other hand, recall that OBTAIN is not a general query language, thus, important features such as nested queries, join, and other query operators are not supported in the OBTAIN language, unlike most of the query languages mentioned above.

With respect to visual temporal query languages, the TVQL [30], TVQE [31], and MQuery [32] temporal visual query languages and their appropriate visual query environments use a series of sliders, checkboxes, and other widgets to specify the temporal constraints on the start or end boundary time points of temporal intervals or relationships between temporal intervals. The *TimeFinder* system [33,34] is a visual exploration and query system for exploring time point-based data sets, based on a direct manipulation metaphor. The main disadvantage of these systems is the lack of access to domain knowledge, and thus meaningful temporal abstraction cannot be specified. Moreover, these systems focus only on retrieving time-oriented *data* and do not support the retrieval of a set of *subjects* (e.g., patients). Chittaro and Combi [35] provide a framework for visual representation of temporal intervals and of their interrelations. However, the proposed techniques are focused on the visual definition of temporal queries regarding the time-oriented data, rather than on the specification of groups of patients.

Several proposals have been made to enhance query capabilities through the use of domain-specific knowledge, in a manner that somewhat resembles this aspect of our research framework [36,37], but they are specific to *only* one domain (do not have underlying generic temporal abstraction ontology) and cannot be used in other medical domains.

2.2.2. Temporal aggregation

In historical databases, *temporal aggregation* (or *temporal grouping*) is a process in which a time line is partitioned over time and the values of various attributes in the database are grouped over these partitions [38]. As such, temporal aggregation can be seen as a temporal extension of the standard SQL operator *group by*. The temporal grouping algebra and its application to the SQL language were well defined by Clifford et al. [39]. A typical example of temporal aggregation is the monthly accumulation of salary payment. Due to the large number of temporal data and their distribution over the time line, efficient algorithms to perform temporal grouping are required, as mentioned by Moon et al. [40], who proposed several methods for large-scale temporal aggregation.

The temporal aggregation process is also relevant to our study. We perform a type of temporal aggregation during the visualization of longitudinal data for multiple patients (actually, of the *delegate values* of these data: see Section 4.2) or during the retrieval of patient groups. However, our algorithms for aggregation of the time-oriented data for each individual patient in a group are applied separately to the data for each patient so as to produce delegate values for each concept type at each temporal granularity (see Section 4.2). For example, the VISITORS system and its underlying computational modules enable users to aggregate individual patient data by interval-based temporal abstractions (using the KBTA methodology: see Section 2.1) and to further aggregate these abstractions into distributions at various temporal granularities (e.g., monthly distribution of the state of bone marrow following a bone-marrow transplant). However, these operations are quite different from the semantics of standard temporal aggregation methods in temporal databases. Thus, such methods are less relevant to our study.

2.2.3. Temporal granularity

Various medical domains require representation and exploration of longitudinal patients' data within different time granularities. For example, in intensive care units, medical parameters are measured using a granularity of *minutes* (or even *seconds*); diabetic patients perform glucose tests three times a *day*, and the clinical history of most chronic patients spans several *years*. Another problem is that even for the same patient, temporal clinical information is represented at different time granularities; e.g. (1)"the patient has taken statin medications during the past three months", and (2) "his glucose level rose sharply from 8:00 a.m. to

6:00 p.m. on September 18th, 1996". Thus, the need to support various time granularities is very clear.

In the case of the VISITORS system, we approach these two problems in the following fashion: (1) we perform a summarization of each individual patient's time-oriented data (i.e., calculating delegate values as described in Section 4.2) according to the temporal granularity level, and (2) we support interactive exploration of longitudinal data for individual/multiple patients within different time granularities (see Sections 4.3 and 4.4).

Our intention here is not to propose a general approach to the treatment of temporal granularities, even though such approaches do, indeed, exist. Goralwalla et al. [41] showed an approach to the handling of granularity in temporal data. They separated temporal data into two groups: *anchored* (calendrical day or month, e.g., January 1st, 2008 or May 1978), and *unanchored* (time intervals, e.g., 2 months, 5 h 20 min, etc.) data. Thus, a temporal granule is a special kind of unanchored temporal data [42].

In our framework, a special kind of unanchored temporal data is a *relative time line* (e.g., the period before or after bone-marrow transplantation). A relative time line is a determinate time span [42], in which one boundary time point (start or end) is referenced to a special clinical event (e.g., bone-marrow transplantation) or another clinically significant time point (e.g., start of high fever). Clifford and Rao [43] have proven the importance of using integral time granularities that can be composed of lower level granules. Thus, we define a relative month as 30 days, and a relative year as 12 months, or 360 days. Note that: (1) these periods do not correspond to any anchored durations, such as calendrical months; and (2) weeks are not allowed as unanchored units in our framework, according to the reasoning of Clifford and Rao [43].

Other techniques for the treatment of the temporal granularity issue include the work of Dal Lago and Montanari [44] and other approaches that were demonstrated within clinical domains [26,45]. The GSTP system [46], for example, provides access to a set of implemented algorithms (such as to the AC-G [47], and to an algorithm for converting calendrical expressions into periodical granularities [48]). These algorithms support a solution to the multi-granularity temporal constraint satisfaction problem (TCSP). The GSTP system supports a rich set of predefined temporal granularities: common temporal granularities (e.g., days, months), and special granularities (e.g., weeks, academic semesters). Moreover, new user-defined granularities can be added.

Note, however, that in our study, we focus on *interactive exploration* of time-oriented data for multiple patients (actually of their delegate values) within a predefined set of temporal granularities (seconds, minutes, hours, days, months, and years): thus, developing theoretical algorithms for dealing with temporal granularities was not relevant for the current study.

3. Desiderata for effective exploration of time-oriented data for multiple patients

Our study of the problem of effective and usable visualization and exploration of raw clinical data and, especially, of derived meaningful abstractions from these data, revealed the following set of desiderata for the intelligent interface and exploration operators supporting the task of exploration of time-oriented data for multiple patients.

1. Evaluation of the functionality and usability of the KNAVE-II system for exploration of longitudinal data of *individual* patients [2] demonstrated the importance of the visualization and exploration of meaningful temporal interpretations that were derived from raw clinical data using context-sensitive domain-specific knowledge. Thus, the visualization and exploration of time-oriented interpretations of raw data is also an important requirement for a system whose goal is to explore longitudinal data for *multiple* patients.
2. Representation and computation of raw and abstract concepts at various temporal granularities, especially for interval-based concepts [45], is very important for the clinical domain. Recall in particular the two requirements mentioned in Section 2.2.3: (1) the need to handle different temporal granularities for different medical domains, and (2) the need to support multiple types of clinical data that are represented at different temporal granularities within the same longitudinal patient record. The system should be able to compute statistical aggregations of both *raw* and *abstract* concepts at *any* temporal granularity (e.g., day, month) or for any arbitrary time period (say, from 8 August to 27 September 2007) for either individual or multiple patients. Fulfilling such a requirement constitutes a novel aspect of this study.
3. Before exploring the time-oriented data of a group of patients, the clinician must select the patients that will comprise the patient population to be analyzed. Current selection tools support the specification of patient groups by using demographic (e.g., age, sex) or raw-data time and value constraints (e.g., Hemoglobin values) [49,50]. However, to enable clinicians to quickly and efficiently specify patient populations, there is a pressing need for the ability to specify the relevant patients by using knowledge-based and statistical aggregation constraints, e.g., "select patients whose most frequent Hemoglobin-state value (i.e., the *mode* of Hemoglobin-state values) during the first two months following bone-marrow transplantation was higher than the mode of the Hemoglobin-state of whole patient population".
4. Current information-visualization systems proposed for multiple patients focus solely on the display of patients' *raw* longitudinal data [14–18]. Moreover, such systems do not support either population aggregation capabilities for raw data or visualization of temporal *abstractions* for multiple patients. Thus, we must investigate and implement suitable visualization techniques for effective display of interval-based temporal abstractions (not only of raw data) for multiple patients.
5. The system should be able to support the following tasks for interactive exploration of the data for multiple patients:
   - Performing interactive dynamic exploration of the data of *multiple* patients at different temporal granularities (i.e., zooming in and out of the time line). This is a basic necessary task, which has been solved in several systems for interactive exploration of raw data for the *individual* patient [14–16] and for abstract concepts [2]. However, the visualization and interactive exploration of both raw and abstract concepts for multiple patients require an additional extension.
   - Changing dynamically the method used for the aggregation of patients' data at different temporal granularities (i.e., aggregating differently blood glucose values at a granularity of days versus months). There is no standard solution available for this task.
   - Supporting both an *absolute* (i.e., calendrical) time line and a *relative* time line (which refers to some clinical key event) and performing an alignment of the patients' data along significant clinical reference event (or time point), including the computation of the appropriate aggregation within the relative time line.
6. One of the more important tasks in the analysis of multiple patient data is an investigation of potentially meaningful interrelations, especially *temporal* interrelations such as associations among subsets of raw patient data and abstract concepts. For such purposes different techniques of data mining (e.g., temporal association rules), are often used [51], although

they typically focus only on temporal precedence relations. Current visual data mining systems in the medical domain focus mostly on *raw* data [18,19]. What is required is an interactive user-driven method to compute, display and explore temporal associations for the data of multiple patients among raw and *abstract* concepts.

4. Methods

The specific methods that we used are presented below in six subsections, each in the context of its relevant desideratum as presented in Section 3.

4.1. Knowledge-based abstraction of raw time-oriented data for multiple patients

The VISITORS system is an intelligent interface to a distributed architecture specific to the tasks of retrieval, knowledge-based visualization, and interactive exploration of the time-oriented records of multiple patients. Fig. 2 describes the overall architecture of the VISITORS system. The architecture capitalizes on what we denote as a *temporal abstraction services* (see details below). In particular, we are assuming that all the necessary elements for the temporal abstraction framework (shown in Fig. 2 by dotted lines), such as those designed in our previous studies or equivalent services, are available.

The OBTAIN expressions constructed by end-users (clinicians) are submitted to a SQL- and C#-based multiple-patients retrieval module, which we refer to as *Multiple-Patient Time-Oriented Query* (the *Multi-TOQ*) module. The Multi-TOQ module is responsible for the retrieval of a set of patients, a set of temporal intervals, or a set of time-oriented patient data according to the following three types of OBTAIN expressions, respectively: *Select Patients*, *Select Time Intervals*, and *Get Patients Data*. The Multi-TOQ module combines specialized computational methods, such as calculation of delegate values (see Section 4.2), with database operators, such as the selection of patients by demographic (e.g., sex, age) values. It also creates and fills the virtual databases used for retrieval purposes, for example, calculation of the population mean values for concepts mentioned in the OBTAIN expression.

The Multi-TOQ module interacts with the intelligent *temporal mediator IDAN* [52] that integrates relevant time-oriented data and knowledge, to obtain answers regarding the temporal abstractions. IDAN uses several *temporal abstraction services*, which differ in the mode in which they are used for processing time-oriented data. If the abstractions do not exist as yet, they are computed on the fly by a C#-based, goal-driven temporal abstraction module, which we refer to as *Tempura* (a variation of the *Resume* temporal abstraction system [53]), or by the prolog-based *ALMA* temporal-abstraction module [52] (in the case of complex clinical patterns). Otherwise, if the abstractions have already been previously computed, in a data-driven manner, by the *MOMENTUM* system [54], they are simply retrieved by the Multi-TOQ module. The MOMENTUM system is an active middleware (which is built on the concept of an active knowledge-based time-oriented database) specific to solving the temporal abstraction task for large groups of subjects (e.g., patients) by using an incremental version of the KBTA method [55]. MOMENTUM is a data-driven system, which generates temporal abstractions defined in its knowledge base incrementally, as new patient data arrive, and saves the abstractions in a special layer in the database.

4.2. Statistical aggregation of patient time-oriented data and their abstractions

It is often the case that for a particular patient there are several measurements of the same raw concept during the time granule of interest. For example, several blood-glucose level values during the same day or several Hemoglobin values during the same month or same year. However, the exploration analysis might require a single value at each desired temporal granularity, and the problem might refer to either raw data or abstract concepts.

To aggregate patient data at arbitrary temporal granularities or during specific time periods, we defined the concepts of a *delegate* (representative) *value* that is calculated by a *delegate* (representative) *function*: given a single patient's time-oriented data for a specific concept (raw or abstract) over a particular time interval (including a predefined temporal granularity level), we calculate the delegate value of the patient's data at each time granule (or for a particular time period) using a function specific to the concept and the temporal granularity. Such a calculation is performed for each patient in the group.

In the VISITORS framework, we make the following *four assumptions*:

Assumption 1. We need to support exactly six temporal granularity levels: *seconds*, *minutes*, *hours*, *days*, *months* and *years*, with the lowest granularity being *seconds* and the highest being *years*.

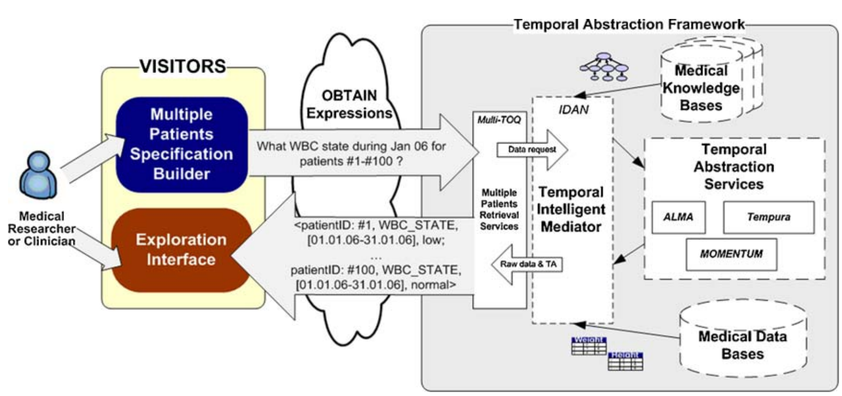

Fig. 2. Overall VISITORS architecture. End users (clinicians) use the expression-specification module of VISITORS to create and submit an OBTAIN expression to the Multiple-Patient Time-Oriented Query (Multi-TOQ) module. The Multi-TOQ module interacts with the IDAN temporal abstraction mediator regarding the requested patients' data and/or temporal abstractions (TAs). The IDAN mediator integrates the relevant data and knowledge from the appropriate sources – indicated by the user in the OBTAIN expression – to retrieve the raw data or to derive, using a temporal abstraction service (the Momentum, Tempura or ALMA temporal abstraction computational modules), a set of abstract time-oriented (interval-based) concepts from these data. The derived temporal concepts are returned to Multi-TOQ for further processing, if needed. The resultant data (patients, time intervals or time-oriented data) answering the OBTAIN expression are returned to the exploration interface.

Assumption 2. The patient has only one concept value at the lowest given temporal granularity level (e.g., for data provided at the temporal granularity level of months and days, no concept has more than one value per day).

Assumption 3. A single delegate value can be requested and computed for any time granule (e.g., second day following bone-marrow transplantation) or arbitrary time period (e.g., January 7th, 2007–February 2nd, 2007), but a series of delegate values at a desired temporal granularity level for a particular concept can be represented for a time period that contains only a whole number of time granules, e.g., a monthly summary can be requested for a period of a whole number of calendrical months.

Assumption 4. The *input data* for our approach has the following data structure:

$$input\_data \equiv < Patient_n, Concept_c, T\text{-}Start_{c,n,m}, T\text{-}End_{c,n,m}, value_{c,n,m} > *, 1 \le n \le N, 1 \le m \le M_n,$$

where N is the number of patients, M_n is the number of *values* of $Concept_c$ for $Patient_n$; $T\text{-}Start_{c,n,m}$ and $T\text{-}End_{c,n,m}$ are the start and end times of the *m*th laboratory test (or temporal abstraction) for $Patient_n$, with value $value_{c,n,m}$. The symbol * denotes 0 or more repetitions.

4.2.1. Computing a single delegate value of a raw-data concept for a specific aggregation time period

The *delegate value* for $Patient_n$ of raw concept $Concept_c$ for a specific aggregation time period [$T\text{-}Start_{aggreg}$, $T\text{-}End_{aggreg}$] is computed by the delegate function DF_c (which is possibly specific to each combination of concept and temporal granularity level) from the *input_data* as follows: $delegate\_value_{c,n,T\text{-}Start_{aggreg},T\text{-}End_{aggreg}} = DF_c[(T\text{-}Start_{n,1}, T\text{-}End_{n,1}, value_{c,n,1}), \ldots (T\text{-}Start_{n,i}, T\text{-}End_{n,i}, value_{c,n,i}), \ldots (T\text{-}Start_{n,K}, T\text{-}End_{n,K}, value_{c,n,K})], T\text{-}Start_{aggreg} \le T\text{-}Start_{n,i}, T\text{-}End_{n,i} \le T\text{-}End_{aggreg}, 1 \le i \le K = K_{c,n,T\text{-}Start_{aggreg},T\text{-}End_{aggreg}}$ where $K = K_{c,n,T\text{-}Startaggreg,T\text{-}Endaggreg}$ is the number of instances of $Concept_c$ for $Patient_n$ measured within the desired [$T\text{-}Start_{aggreg}$, $T\text{-}End_{aggreg}$] period, i.e., K varies for each concept, patient, and aggregation period.

The delegate function is defined in the knowledge base or is chosen at runtime by the user from several predefined default functions. For example, assume that after a bone-marrow transplantation procedure, a patient's Platelet counts on two separate occasions during the same day were 22,000 cells/ml at 10 a.m. and 17,000 cells/ml at 9 p.m. If we specify, in the domain knowledge base, the *mean* as the default delegate daily function for the Platelet count (raw-data) concept, then the patient had a daily delegate value of 19,500 cells/ml for the Platelet count (raw) concept. However, during the interactive exploration time, the user can also select any other suitable delegate function (such as the mode or the maximum).

Indeed, in theory, almost any function from multiple time-stamped values into one value, of the same concept that has the same domain and units as the possible values of the original concept, can serve as the delegate function. However, unlike standard statistical functions, it must be applied to *each time granule* in the relevant aggregation *temporal granularity* (e.g., day). Of course, the selected function must make clinical sense, and is thus specific to each clinical concept and aggregation time granularity level.

4.2.2. Computing a series of delegate values of a raw-data concept at a desired temporal granularity for a given overall aggregation time period

The computing of a series of delegate values is performed for each time granule at the desired temporal granularity level for an overall aggregation time period. Thus, a *series of delegate values* for $Patient_n$ of raw concept $Concept_c$ for an overall aggregation time period [$T\text{-}Start_{overall}$, $T\text{-}End_{overall}$] has the following data structure:

$$delegate\_values_{c,n,T\text{-}Start_{overall},T\text{-}End_{overall}} = < Patient_n, Concept_c, \text{-}Start_{aggreg\,n,j}, T\text{-}End_{aggreg\,n,j}, delegate\_value_{c,n,j} > *, 1 \le n \le N, 1 \le j \le J_{c,n},$$

where N is the number of patients; $J_{c,n}$ is the number of delegate values of $Concept_c$ for $Patient_n$; $T\text{-}Start_{aggreg\ n,j}$ and $T\text{-}End_{aggreg\ n,j}$ are the start and end times of the time granule that is specific to the *j*th delegate value for $Patient_n$; and $J_{c,n}$ varies with each patient, concept and the particular overall time period of a series of delegate values.

The values for boundary time granules of $Concept_c$ for $Patient_n$ of the start point $T\text{-}Start_{aggreg\ n,1}$ of the first time granule and of the end point $T\text{-}End_{aggreg\ n,Jc,n}$ of the last time granule are:

$$T\text{-}Start_{aggreg\,n,1} = BeginOfTimeGranule(T\text{-}Start_{c,n,1}) \ge T\text{-}Start_{overall},$$
$$T\text{-}End_{aggreg\,n,Jc,n} = EndOfTimeGranule(T\text{-}End_{c,n,Mn}) \le T\text{-}End_{overall}.$$

The *j*th index denotes the *j*th delegate value of the specific $Patient_n$. For example, assume that laboratory tests for $Concept_c$ were performed for the patient several times on each of the days January 3rd, January 15th, and January 20th, then the $T\text{-}Gran_{aggreg}$ is *day*, and the *DF* is *mean*. Thus, the first delegate value will be the mean value of all of the concept-measurement results on January 3rd, the second delegate value will be the mean value of all of the results on January 15th and so on. In this case J = 1.3 for the overall aggregation period of January. If $T\text{-}Gran_{aggreg}$ is *month* and the overall aggregation time period is 1 January–31 August for the same year, then the first delegate value will be the mean value of all the tests during January. Note that several of the months during the requested overall period might have no measurements, and hence no delegate values representing them.

4.2.3. Computing a single delegate value for a specific aggregation time period and a series of delegate values concepts at a desired temporal granularity for a given overall aggregation time period of an abstract concept

The computing of a delegate value for a specific aggregation time period or a series of delegate values at a desired temporal granularity for a given overall aggregation time period in the case of interval-based abstract concepts [such as intervals of different (discrete) grades of bone-marrow toxicity] is similar to the aggregation of raw (time point based) patient's data, as explained in Sections 4.2.1 and 4.2.2, but requires the use of more complex delegate functions, because we must refer to both the value of the temporal abstraction interval and to its duration. Thus, standard statistical functions are not sufficient. In this case, we provide additional delegate functions, such as the value of the abstraction that has the maximal cumulative duration during the relevant aggregation time period or the value associated with the interval that has the longest duration.

Since values of abstract concepts often span time intervals (i.e., a period and not just a time point), we consider the case of an overlap between the interval during which the value holds, and the several time granules, which are at the time granularity level that had been requested for the computation of the delegate value. Assume that the abstract concept computed for the relevant patient spans four time intervals (denoted by i1–i4) within the range of three time units of the desired time granularity level (Fig. 3a). First, we extend the temporal intervals by performing an extrapolation operation that is an extension of the knowledge-based interpolation operation [56], according to the concept-sensitive relevant interpolation function (the extension of the intervals denoted by dotted lines in Fig. 3b). The extension is necessary to enable application of various statistical functions,

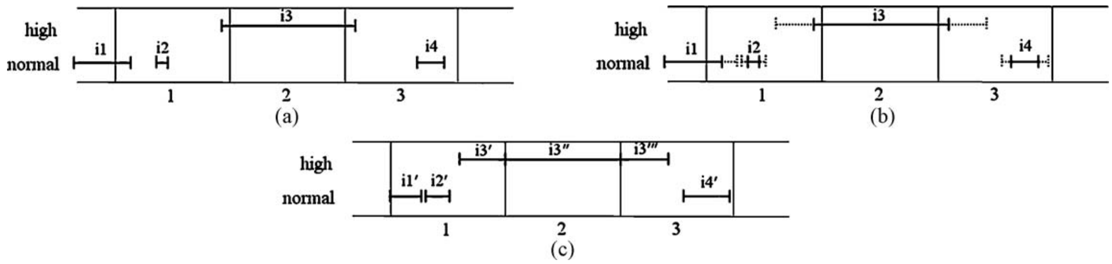

Fig. 3. Computation of a series of delegate values for an abstract concept (see text for details).

since abstractions that hold at a particular time point must be extended so as to hold over a sufficiently long time interval. Similarly, we extend longer precomputed intervals beyond their two edges, assuming that the abstract value persists beyond the crisp temporal points that define the time interval. The local extrapolation of a point or an interval's boundary point uses the $\Delta(0,0)$ interpolation function for the relevant concept [56], i.e., the maximal gap allowed between two single time points, at each of which holds the value of the abstract concept. We actually extend each boundary point to the past and the future by amount of $\Delta(0,0)/2$. In the case of adjacent intervals that have different values for the same abstract concept, the extrapolation is proportional to the duration of the intervals. Second, we segment the extended intervals according to the time granules of the desired time granularity (Fig. 3c). Third, we apply the default delegate function for that concept and time granularity. The default for all concepts is currently the value that has maximal cumulative duration within the time granule. In Fig. 3c, within the first time granule the delegate value is "normal" (note that the sum of extended durations of intervals i1′ and i2′ is more than the duration of the extended interval i3′ within the first time granule). The "high" value is the delegate value computed for the second and third time granules. Note, that interval i3‴ (part of extended i3 within the third time granule) has a higher duration in that granule than the extended interval i4′; thus, the delegate value computed for the third granule is "high"; although the original duration of interval i4 was longer than the part of i3 within the third time granule. Using other delegate functions could change the computed delegate values.

4.2.4. Computing a single population delegate value of raw-data and abstract concepts for a specific aggregation time period

The *population delegate value* represents the aggregated value of a group of patients calculated by the concept-specific and the temporal-granularity-specific statistical function within a specific time granule (or for another specific time period), e.g., the maximal value of the raw Hemoglobin values for a group of patients over a whole month. The population delegate value of a group of patients for $Concept_c$ within a specific aggregation time period [$T\text{-}Start_{aggreg}$, $T\text{-}End_{aggreg}$] is calculated by the population delegate function PDF_c from the set of original patient data, *input_data** as follows:

$$\begin{aligned}&population\_delegate\_value_{c,T-Start_{aggreg},T-End_{aggreg}}\\&\quad= PDF_c[(T\text{-}Start_{n,1}, T\text{-}End_{n,1}, value_{c,n,1}), \ldots (T\text{-}Start_{n,i}, T\text{-}End_{n,i},\\&\quad value_{c,n,i}), \ldots (T\text{-}Start_{n,K}, T\text{-}End_{n,K}, value_{c,n,K})], 1 \le n \le N,\\&\quad 1 \le i \le K, T\text{-}Start_{aggreg} \le T\text{-}Start_{n,i}, T\text{-}End_{n,i}\\&\quad = T\text{-}End_{aggreg}, 1 \le i \le K \le K_{c,n,T-Start_{aggreg},T-End_{aggreg}}\end{aligned}$$

where N is the number of patients in the group, and $K = K_{c,n,T\text{-}Startaggreg,T\text{-}Endaggreg}$ is the number of instances of $Concept_c$ for $Patient_n$ measured within the [$T\text{-}Start_{aggreg}$, $T\text{-}End_{aggreg}$] time period.

An example of a population delegate value is the maximal value of the concept across all patients within a specific month (see the line denoted by 2 in Fig. 4).

Similarly to the delegate function DF_c, the population delegate function PDF_c calculates the delegate value for $Concept_c$ over each relevant time granule. However, through the population delegate function PDF_c, we calculate a "*delegate value*" that represents a population value for $Concept_c$ for a group of patients during that

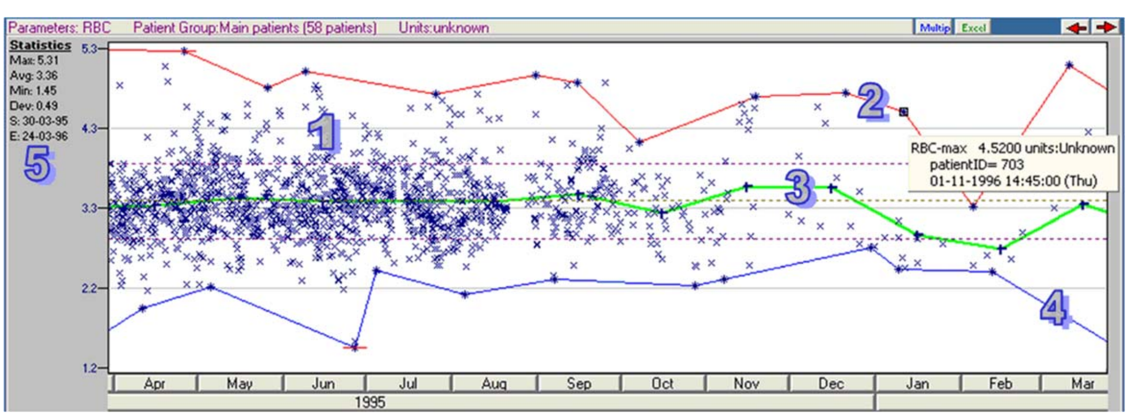

Fig. 4. Visualization of the data for the Red blood cell (RBC) count raw concept for a group of 58 patients (retrieved earlier by using a *Select Patients* expression) from April 1995 to March 1996. The individual raw data are represented at a resolution level of *seconds* with respect to time granularity, but the population statistics are aggregated at a granularity of *months*, according to the user's current request. All laboratory test results for the RBC count for patients treated during this period are displayed as (blue) X's (denoted by region 1). Through the density of the points, the user can judge the number of data instances belonging to each value or time period range. The top (red) line (denoted by 2) represents the *monthly* maximal value of the whole group. The tooltip provides detailed information about the maximal value (4520 cells/ml), the ID of the patient (703) with that value, and the observation time. The bottom (blue) line (denoted by 4) and the middle (green) line (denoted by 3) represent the *monthly* minimal and mean values, respectively. On the left-hand side (denoted by 5) are displayed statistics for the whole time period (April 1995 to March 1996). The three dotted lines drawn inside the panel indicate the mean value ± standard deviation for this period (For interpretation of the references to color in this figure legend, the reader is referred to the web version of the article).

time granule. The population delegate function can be any statistical function, such as the mean or the maximal value, which can be applied to the values of all of the patients within a specific time granule and that returns a single output value from the domain of values and units of *Concept*$_c$.

4.2.5. Computing a series of population delegate values of raw-data and abstract concepts at a desired temporal granularity for a given overall aggregation time period

The process of computing a series of population delegate values is similar to computing a series of individual delegate values: we compute a population delegate value for each time granule at a desired temporal granularity level for an overall aggregation time period. Thus, a *series of population delegate values* of *Concept*$_c$ for an overall aggregation time period [*T-Start*$_{overall}$, *T-End*$_{overall}$] has the following data structure:

$$population\_delegate\_values_{c,T\text{-}Start_{overall},T\text{-}End_{overall}} \equiv < Concept_c, \text{-}Start_{aggreg\ j}, T\text{-}End_{aggreg\ j}, population\_delegate\_value_{c,j} > *, 1 \leq n \leq N, 1 \leq j \leq J_c,$$

where J_c is the number of population delegate values of *Concept*$_c$ for the overall aggregation time period; and *T-Start*$_{aggreg\ j}$ and *T-End*$_{aggreg\ j}$ are the start and end times of the time granule that is specific to the *j*th population delegate value.

4.3. The OBTAIN specification language

The VISITORS system includes a graphical module, which is used for specification of the relevant patients, time intervals and values. Underlying the VISITORS graphical expression-specification module, is an *ontology-based temporal-aggregation* (OBTAIN) specification language. The OBTAIN specification language enables the specification of three types of expression: *Select Patients*, *Select Time Intervals* and *Get Patients Data*. A full exposition of the OBTAIN language and the graphical expression-specification module is beyond the scope of this study and can be found elsewhere [4]. We will briefly explain, however, these three expression types. The Appendix A includes the Backus Normal Form (BNF) syntax of all three expressions.

4.3.1. Select Patients expression

The *Select Patients* expression retrieves from a selected database a list of patients who satisfy a set of either *demographic* (i.e., non-temporal) constraints, for example, "select male patients [currently] insured by a certain health maintenance organization", or *time and value* [knowledge-based] *constraints*, for example: "find patients who had, during the first month following bone-marrow transplantation, at least one episode of bone-marrow toxicity (an abstraction defined by the relevant clinical protocol) of "grade1" or higher that lasted at least two days"; or [time and value] *proportional and statistical constraints*, for example, "find patients taking a Statin-type medication whose mean Low density lipid (LDL) cholesterol value exceeded that of the mean value for all patients taking Statin-type medications".

4.3.2. Select Time Intervals expression

Given a set of time-oriented patient data, this expression returns a list of time intervals that satisfy the constraints defined by the user for some portion of the patients. The goal of this expression is to find *when* a certain portion of the patients had a specific value of some raw or abstract concept or value within a predefined value range. For example, a typical *Select Time Intervals* expression is "Find periods [relative to an allogenic bone-marrow transplantation event] during which the White blood cell state value was less than "normal", and the Platelet value was between 2000 and 10,000 cells/ml for at least 50% of the patients."

4.3.3. Get Patients Data expression

Given a concept, a list of patient IDs and, optionally, a list of time intervals, the expression retrieves the values of the concept within the selected time intervals for the selected patients. The default patient list is all the patients in the database, and by default there are no time-interval constraints, i.e., values are returned for the entire timeline. For example, a typical *Get Patients Data* expression is "Get the state values of Hemoglobin for patients #1–#10 during the first two weeks following bone-marrow transplantation."

The *Get Patients Data* expression is not one of the expression types specified within the expression-specification module. Rather, it is generated by an interactive exploration process. Once a patient list has been defined by a *Select Patients* expression (or by the explicit setting of the ID list in the exploration interface), selecting a concept from the ontology browser (see Fig. 1) opens a new panel with the values of that concept for all of the patients in the list.

4.3.4. Evaluation of the graphical expression-specification module

We have previously evaluated both the functionality and the usability of the expression-specification module by a group of 10 users—five clinicians and five medical informaticians. Results have shown that both types of users were able, in a short time and with high accuracy, to graphically construct complex expressions, although the accuracy of the specification of time-range constraints regarding the start or end of time intervals was significantly lower than that of the rest of the constraints. The details of this evaluation appear in a previous study [4].

4.4. Visualization of the data for both raw and abstract concepts

The data set retrieved by the *Get Patients Data* expression can be visualized and explored. In general, we provide a two-dimensional visualization, in which the horizontal axis is the time dimension, and the vertical axis is the value dimension.

4.4.1. Visualization of the data of raw concepts

For the basic visualization of the data of raw concepts for multiple patients, we use the line plot visualization technique, which plots the data points on the screen according to the X and Y coordinates (i.e., according to the time and values coordinates in our case) and either connects or does not connect the plotted points by lines (Fig. 4). In this visualization, three data types are represented:

(1) *The individual patient's raw concept data for all patients in a group* (denoted as 1 in Fig. 4). In fact, the delegate values of the data for the current time granularity are represented.
(2) *The time-oriented population statistics of the whole group of patients*. In fact, the population delegate values of the group of patients for the current time granularity are represented. We represent the following three time-oriented statistics, each aggregated (calculated) within a potentially different time granularity: maximum value at each time granule (top red line denoted as 2 in Fig. 4), minimum value at each time granule (bottom blue line denoted as 4 in Fig. 4), and mean value for each time granule (the middle green line denoted as 3 in Fig. 4) population values. Each population delegate value is a function of all of the population values in the relevant time granule (e.g., month in this example).
(3) *The value statistics, which are sensitive to the particular time window of the displayed data* (denoted by the number 5 in Fig. 4). The dynamically changing content of the displayed data (e.g., by panning) causes recalculation of these statistics. Default statistics for a raw data concept include descriptive

statistics, such as the mean, maximum, and minimum values, and the standard deviation.

To formally define the configuration of the display for *raw individual patient data* and to understand the semantics of the exploration operators, we introduce the *Graph Computational Manager* (GCM), which stores both the concepts to be displayed and several important computational parameters relevant to these concepts. The GCM has the following data structure:

$$< Concept_c, input\_data*, T\text{-}Gran_{aggreg}, DF_c >, 1 \leq c \leq C,$$

where *C* is the number of concepts in the knowledge base; *input_data** is the original source data accessed by the temporal mediator and retrieved by the *Get Patient Data* expression; $T\text{-}Gran_{aggreg}$ defines the temporal granularity level of the individual patient's aggregated data (e.g., one data point per day); and DF_c is the delegate function, appropriate to $Concept_c$, used for calculating the delegate values for each patient at the $T\text{-}Gran_{aggreg}$ granularity (e.g., the daily mean).

To intelligently and more efficiently explore the M_n data points of $Concept_c$ for $Patient_n$, we use an aggregation of the values through a delegate-value function specific to each concept and time granularity level (as described in Section 4.2). Thus, for example, several daily values out of the M_n values might be represented by a single delegate value, e.g., the maximum.

The *Graph Display Manager* (GDM) controls the actual data to be displayed and has the following data structure: $< Concept_c, delegate\_values*, T\text{-}Start_{explor}, T\text{-}End_{explor}, T\text{-}Gran_{explor}, [RefP] >, 1 \leq c \leq C,$ where *C* is the number of concepts in the knowledge base; *delegate_values** were calculated by the delegate function appropriate to the temporal aggregation level $T\text{-}Gran_{aggreg}$, as stored by the GCM; $T\text{-}Start_{explor}$ and $T\text{-}End_{explor}$ define the temporal range of the current display time window, which includes 0 or more delegate values of $Concept_c$ for each of the relevant patients; $T\text{-}Gran_{explor}$ denotes the current temporal granularity level of the exploration; and *RefP* is an optional parameter that defines the reference symbolic point (e.g., bone-marrow transplantation) when a relative time line is being used.

The GCM of the *time-oriented population statistics* is similar to the GCM of the raw data. However, it uses a population delegate function, instead of an individual delegate function:

$$< Concept_c, input\_data*, T\text{-}Gran_{aggreg}, PDF_c >, 1 \leq c \leq C,$$

where *C* is the number of concepts in the knowledge base; *input_data** is the original source data accessed by the temporal mediator and retrieved by the *Get Patient Data* expression; $T\text{-}Gran_{aggreg}$ defines the temporal granularity level of the patient's statistical data; and PDF_c is the population delegate function, appropriate to $Concept_c$, used for calculating a population delegate values at the $T\text{-}Gran_{aggreg}$ granularity (as described in Section 4.2).

The GDM of the time-oriented population statistics has the following data structure:

$$< Concept_c, population\_delegate\_values*, T\text{-}Start_{explor}, T\text{-}End_{explor}, T\text{-}Gran_{explor}, [RefP] >, 1 \leq c \leq C,$$

where *C* is the number of concepts in the knowledge base; the *population_delegate_values** structure stores the values of the population statistics calculated for all time granules by the appropriate PDF_c function; $T\text{-}Start_{explor}$ and $T\text{-}End_{explor}$ define the temporal range of the current display time window, which includes 0 or more population delegate values; $T\text{-}Gran_{explor}$ denotes the current temporal granularity level of the exploration; and *RefP* is an optional parameter that defines the reference point (e.g., bone-marrow transplantation) when a relative time line is being used.

$T\text{-}Gran_{aggreg}$ and $T\text{-}Gran_{explor}$ can be different or they can be the same. For example (*see* Fig. 4), the individual aggregation temporal granularity level might be *seconds*, while the exploration temporal granularity might be *months*. If both temporal granularities are the same, each patient will be represented by only one delegate-value point for the displayed concept for each time granule.

$T\text{-}Gran_{explor}$ defines the exploration granularity of the graphical panel for all patient data displayed in the panel. It might be different for each panel, even in a case of the exploration of the data of an equal concept twice. The default is to use the same exploration granularity for all panels, which enables the synchronization of the explored data among different concepts.

If both the individual patient data and population statistics are displayed in the panel, then $T\text{-}Start_{explor}$, $T\text{-}End_{explor}$, $T\text{-}Gran_{explor}$ parameters are same in each of the GDMs.

4.4.2. Visualization of the data of abstract concepts

Temporal abstractions for multiple patients are displayed as a distribution of the delegate values of the abstract concept of the patients at the desired (interactively modified) time granularity. For the visualization, we use a modified version of the bar chart visualization technique. The modification includes providing separate [0.100%] scales for each of the possible values of the temporal abstractions (e.g., for each toxicity grade), which is useful for discovering trends in the distribution. We assume a finite number of [symbolic] values for each abstract concept. The

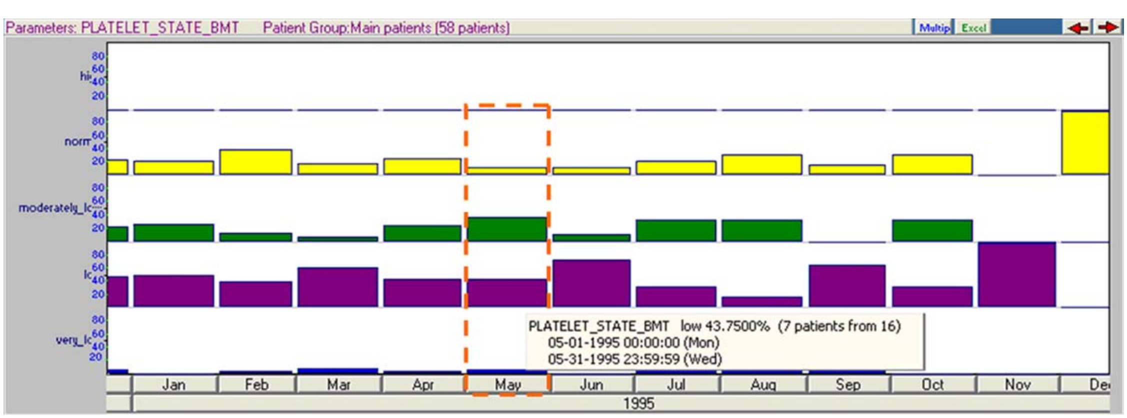

Fig. 5. Visualization of the values of the Platelet-state abstract concept for a group of 58 patients (retrieved earlier by using a *Select Patients* expression) during 1995. The user sees the monthly distribution of the patients' values during this period (e.g., the dotted rectangle denotes the distribution of the concept's values during May). Note that the vertical axis has two scales: an external ordinal scale of possible values and an internal percentage scale, from 0 to 100%, within each concept value. The tooltip provides detailed information about the contents of the displayed area: the value (low) and its proportion (43.75%), the start and end time points and the actual number of patients with the selected value (7) out of all patients (16 patients). Note that only 16 patients in the group of 58 patients had the necessary raw data (in this example, the Platelet count) during May 1995.

visualization of the values of an abstract concept for multiple patients is shown in Fig. 5.

The GCM of a temporal abstraction graph has the following data structure:

$$< Concept_c, input\_data*, T\text{-}Gran_{aggreg}, DF_c >, 1 \le c \le C,$$

where C is the number of concepts in the knowledge base; *input_data** is the original source data accessed by the temporal mediator and retrieved by the *Get Patient Data* expression; $T\text{-}Gran_{aggreg}$ defines the temporal granularity level of the patient's aggregated data; and DF_c is the delegate function, appropriate to $Concept_c$, used for calculating the delegate values for each patient in the $T\text{-}Gran_{aggreg}$ granularity.

Since the data of an abstract concept for multiple patients are displayed as a set of distributions of the abstract-concept values, one for each of the temporal granules of the $T\text{-}Gran_{aggreg}$ level, the GDM of the temporal abstraction differs from the GDM of the raw data:

$$< Concept_c, distribution*, T\text{-}Start_{explor}, T\text{-}End_{explor}, T\text{-}Gran_{explor}, [\mathrm{Ref}P] >, 1 \le c \le C,$$

where C is the number of concepts in the knowledge base; *distribution** is the data structure $\{[val_c^l, \mathrm{Prev}_c^l]_1 \ldots [val_c^l, \mathrm{Prev}_c^l]_J\}$, where val_c^l is the 1st value of $Concept_c$ (usually measured on ordinal symbolic scale) and Prev_c^l is the prevalence of patients having that value; and J is the number of temporal granules at the aggregation granularity level $T\text{-}Gran_{aggreg}$, during the exploration time interval [$T\text{-}Start_{explor}$, $T\text{-}End_{explor}$]. The delegate values are calculated by the delegate function appropriate to the $T\text{-}Gran_{aggreg}$ as stored by the GCM. $T\text{-}Start_{explor}$ and $T\text{-}End_{explor}$ define the temporal range of the current display time window, which includes 0 or more delegate values of the $Concept_c$ for each relevant patient. $T\text{-}Gran_{explor}$ denotes the current temporal granularity level of the exploration. *RefP* is an optional parameter that defines the reference point (e.g., bone-marrow transplantation) when a relative time line is being used. In the case of temporal abstractions, $T\text{-}Gran_{aggreg}$ and $T\text{-}Gran_{explor}$ are always the same; for example (see Fig. 5), the aggregation granularity and exploration granularity could both be months.

4.5. Exploration operators of time-oriented data of multiple patients

In this section, we describe the formal definition and semantics of the general exploration operators for the exploration of time-oriented data for multiple patients, i.e., the input, output, and necessary calculations performed on the patients' data. Each of the operators can be applied on three different possible types of data displayed in the graph:

- Individual raw data, such as Hemoglobin values, White blood cell count, and other laboratory tests, interventions and medications.
- Time-oriented population statistical values (e.g., maximal monthly value of Hemoglobin), calculated from the data of raw concepts for multiple patients.
- Temporal abstractions, such as the values of the Hemoglobin-state abstract concept, derived from the data of raw concepts by using the domain-specific context-sensitive knowledge base.

The application of the exploration operators to different data types differs with the performed calculation and the output. Several of the operators act in a "syntactic" fashion, i.e., perform simple magnification or minification of the displayed data without additional calculation; thus the input and output data are same. Other operators return output data that are different from the input data (e.g., by computing the mean values), but that are still in the input concept's domain of values and units.

For conciseness, we introduce the following annotation, which we use to formally define the semantics of all three operators:

- *current_values**: denotes the current data that are displayed by the GDM before the application of the operator (i.e., *delegate_-values** in the case of individual raw data, *population_delegate_-values** in the case of time-oriented population statistics, and *distribution** in the case of temporal abstractions).
- *D-function*: denotes the delegate function DF_c in the case of individual raw data and temporal abstractions or the population delegate function PDF_c in the case of time-oriented population statistics.
- *new_values**: denotes the new actual data to be displayed by the GDM after the application of the operator. The *new_values* are calculated by the *function* accordingly to the aggregation time granularity from the original data accessed by the temporal mediator.

4.5.1. The Temporal Exploration Operator

The VISITORS system enables the user to manipulate the data interactively, e.g., to pan the patient data within various time intervals or to zoom-in and zoom-out of the patient data within different time granularities ranging from seconds to decades.

For individual values of raw concepts for multiple patients the Time Exploration Operator operates as a syntactic zoom. It performs a magnification or minification of the displayed data during the specific predefined period of time (i.e., a particular time granule or a specific time period). This zoom function facilitates the display of large amounts of patient data (i.e., an overview mode) or a reduction of the time granularity to support in-depth exploration of the patients' data for a particular time granule of an arbitrary time range. The individual delegate and population delegate values of a raw data concept are not recalculated during the application of the Time Exploration Operator; thus, this operator is indeed purely

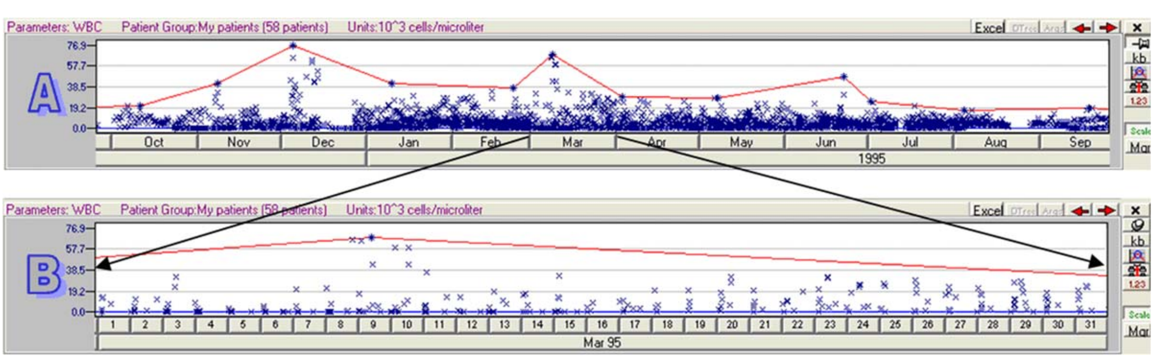

Fig. 6. Application of the Time Exploration Operator (in this case a zoom-in) to the data for the White blood cell (WBC) raw concept for a group of 58 patients selected earlier by the user. Clicking on the "Mar" (March) month widget in panel A results in a display of the data throughout the month of March, as shown in panel B. Note, clicking on the "Mar 95" granule in the panel B would again provide panel A.

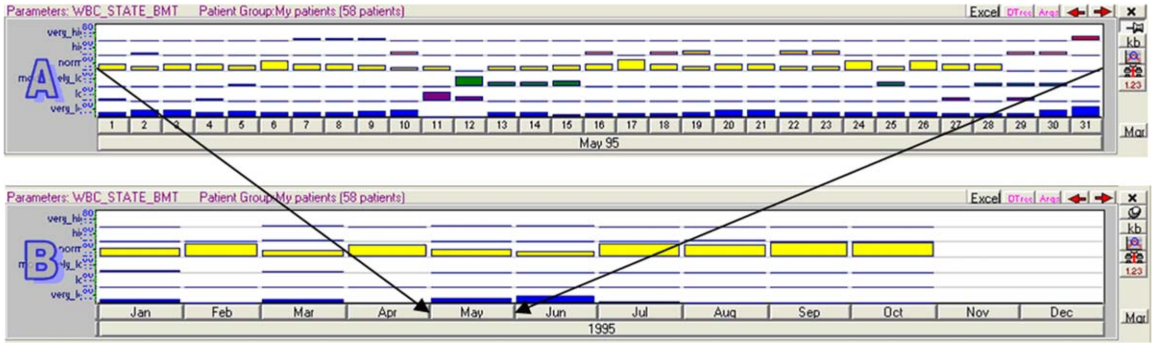

Fig. 7. Application of the Temporal Exploration Operator (in this case a zoom-out) to the data for the White blood cell (WBC) state abstract concept for a group of 58 patients selected earlier by the user. Clicking on the "May 95" month widget in panel A produces a display of the distributions of data throughout the year of 1995 according to the *months* time granularity, as shown in panel B. Note that the required raw data (in this example, the WBC count) might not necessarily have been available for all 58 patients during May 1995. Clicking on the "May" granule in the panel B would again provide panel A.

syntactic, as opposed to the semantic operators, which significantly modify the displayed data.

For an individual patient's data and for population statistics of raw concepts the Temporal Exploration Operator is applied through the GDM and may be formally defined as follows:

$$\begin{aligned}&TEO(GDM: < Concept_c, current\_values*, T\text{-}Start_{\mathrm{explor}}, T\text{-}End_{\mathrm{explor}},\\&\quad T\text{-}Gran_{\mathrm{explor}}, [\mathrm{Ref}P] >, T\text{-}Start'_{\mathrm{explor}}, T\text{-}End'_{\mathrm{explor}}, T\text{-}Gran'_{\mathrm{explor}})\\&\quad \Rightarrow GDM': < Concept_c, new\_values*, T\text{-}Start'_{\mathrm{explor}}, T\text{-}End'_{\mathrm{explor}},\\&\quad T\text{-}Gran'_{\mathrm{explor}}, [\mathrm{Ref}P] >,\end{aligned}$$

where $T\text{-}Start'_{explor}$, $T\text{-}End'_{explor}$, and $T\text{-}Gran'_{explor}$ are user-determined parameters of the new earliest and latest time points for display, and the new exploration temporal granularity, respectively. Fig. 6 shows the application of the Temporal Exploration Operator to the data of a raw concept for a group of patients.

Note that the default delegate function is the *identity* function. Thus, the output from the mediator will be recalculated according to the input temporal granularity of the data and represented using an identical temporal granularity.

Application of the Temporal Exploration Operator to the values of abstract concepts calculated for *multiple* patients is more complex and quite different. To apply the Temporal Exploration Operator in this case, we must recalculate the distributions according to the new time granularity level; thus, the application of the operator is, in fact, semi-semantic and is applied through both the GCM and the GDM:

$$\begin{aligned}&TEO(GCM: < Concept_c, input\_data*, T\text{-}Gran_{aggreg}, DF_c >,\\&\quad GDM: < Concept_c, distribution*, T\text{-}Start_{\mathrm{explor}}, T\text{-}End_{\mathrm{explor}},\\&\quad T\text{-}Gran_{\mathrm{explor}}, [\mathrm{Ref}P] > T\text{-}Gran'_{aggreg}, T\text{-}Start'_{\mathrm{explor}}, T\text{-}End'_{\mathrm{explor}},\\&\quad T\text{-}Gran'_{\mathrm{explor}}) \Rightarrow (GCM': < Concept_c, input\_data*,\\&\quad T\text{-}Gran'_{aggreg}, DF_c > GDM': < Concept_c, distribution'*,\\&\quad T\text{-}Start'_{\mathrm{explor}}, T\text{-}End'_{\mathrm{explor}}, T\text{-}Gran'_{\mathrm{explor}}, [\mathrm{Ref}P] >),\end{aligned}$$

where $T\text{-}Start'_{explor}$, $T\text{-}End'_{explor}$, and $T\text{-}Gran'_{explor}$ are user-determined parameters of the new earliest and latest time points for display, and the new exploration temporal granularity, respectively; and *distribution*′* is the new distributions of delegate values that were calculated by the delegate function appropriate to the $T\text{-}Gran_{aggreg}$ as stored by the GCM. Fig. 7 shows the application of the Temporal Exploration Operator to the data of abstract concept for a group of patients.

The user-determined parameters $T\text{-}Start'_{explor}$, $T\text{-}End'_{explor}$, and $T\text{-}Gran'_{explor}$ can be derived in several ways: panning (i.e., shifting the displayed data to the right or left) changes the $T\text{-}Start'_{explor}$ and $T\text{-}End'_{explor}$ parameters; using a calendaric-range zoom (i.e., standard calendar function) enables the user to specify the $T\text{-}Start'_{explor}$ and the $T\text{-}End'_{explor}$ time points to zoom-in to a specific time range; and clicking the specific temporal granule at the bottom part of the graphs (see Figs. 6 and 7) performs a zoom-in or zoom-out operation according to the selected time granule; thus, all three parameters may be changed. A detailed exposition of all of the ways for manipulating the temporal granularity can be found in the description of the KNAVE-II system, which focuses on exploration of individual patient data [2].

4.5.2. The Change Delegate Value Operator

As explained earlier, the VISITORS system supports aggregation of patient data according to a given time granularity. Such aggregate values are designated the "delegate values" for that granule (for example, a representative blood-glucose value for the whole day or even the whole month). To apply another delegate function or to change the granularity level of the aggregation, the user can use the Change Delegate Value Operator (CDVO). Note that the data values are changed, but the data still maintain the same type and domain of values.

For all three types of displayed data, the operator is applied through the GCM. Thus, the GDM will display the new values within the same display configuration, i.e., the earliest and latest times for display and exploration granularity are the same before and after application of the Change Delegate Value Operator. The formal definition of the Change Delegate Value Operator is as follows:

$$\begin{aligned}&CDVO(GCM: < Concept_c, input\_data*, T\text{-}Gran_{aggreg}, D\text{-}function >,\\&\quad GDM: < Concept_c, current\_values*, T\text{-}Start_{\mathrm{explor}}, T\text{-}End_{\mathrm{explor}},\\&\quad T\text{-}Gran_{\mathrm{explor}}, [\mathrm{Ref}P] >, T\text{-}Gran'_{aggreg}, D\text{-}function') \Rightarrow GCM':\\&\quad < Concept_c, input\_data*, T\text{-}Gran'_{aggreg}, D\text{-}function' >,\\&\quad GDM': < Concept_c, new\_values*, T\text{-}Start_{\mathrm{explor}}, T\text{-}End_{\mathrm{explor}},\\&\quad T\text{-}Gran_{\mathrm{explor}}, [\mathrm{Ref}P] >,\end{aligned}$$

where $T\text{-}Gran'_{aggreg}$, and *D-function*′ are the user-determined parameters of the new temporal aggregation granularity and *D-function*, respectively. The *new_values** are derived from the input data as explained in Section 4.2.

Figs. 8 and 9 show the application of the Change Delegate Value Operator to the values of raw and abstract concepts for a group of patients.

For both cases of the visualizations, i.e., for the raw and abstract concepts, we provide a Graph Manager interface (not shown here), in which the user can change the delegate function and the aggregation temporal granularity for each graph in the panel; e.g., the maximal and mean population statistics might be represented

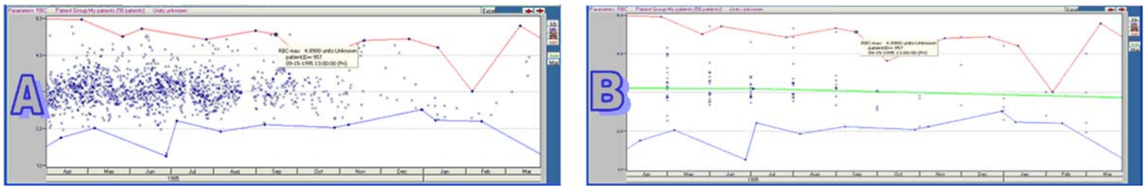

Fig. 8. Visualization of the data for the Red blood cell (RBC) count raw concept for the 58 patients (selected earlier by the user) before (panel A) and after (panel B) application of the Change Delegate Value Operator. Panel A displays the RBC data at a granularity of *seconds* and minimal- and maximal-value population delegate values (population statistics) at a granularity of *months* (the mean is not displayed). The resultant visualization (panel B) displays the *monthly mean* RBC values for each patient. The green graph represents the *yearly mean* value (one value during year 1995) for all patients for whom data had been collected during the year 1995.

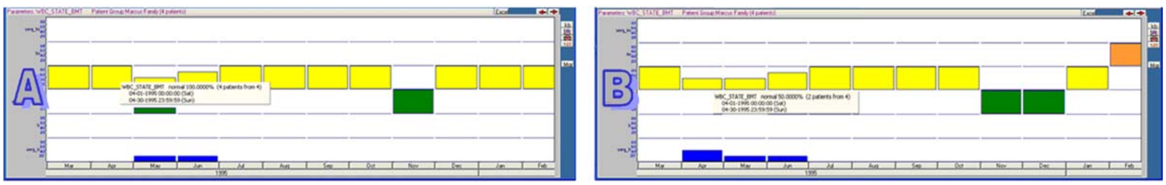

Fig. 9. Visualization of the data for the White blood cell (WBC) state abstract concept for group of patients before (panel A) and after (panel B) application the Change Delegate Value Operator. Panel A displays the distribution of the values by representing the value of the abstraction that has the maximal cumulative duration. By using such a function, the state of the WBC count for all patients was abstracted as "normal" during April 1995 (see tooltip in the panel A). The resultant visualization (panel B) displays the new distribution of the values, this time using a delegate function that selects the value associated with the longest-duration interval. Thus, during April 1995, the state of the WBC count for half of the patients was abstracted as "normal", and for half of the patients was abstracted as "very-low".

at a *monthly* granularity level, and the individual raw data might be represented at a resolution level of *seconds*, as is shown in Fig. 4.

4.5.3. The Set Relative Time Operator

Changing dynamically the point of view from an absolute (calendar-based) time line to a relative time line is a key operator in the VISITORS system. The relative time line is set by identifying clinically significant events, or another clinically significant time point in the domain's temporal-abstraction ontology (e.g., start of therapy, birth of the child, start of high fever), which serve as a date of reference (time zero) for all patient data.

Several patients in the group might not have experienced the *reference event* (e.g., a particular type of intervention). Thus, only the data points of patients who have experienced the reference event are displayed. Since each patient may have experienced the intervention at a different time from the other patients, we must align the data of all of the patients according to the reference time point, which is set as the new (relative) zero-time point. Moreover, in the case of a temporal abstraction, we must calculate the distribution of abstract concept values by including only patients who had experienced the selected event or intervention.

For uniformity, each "month" in the relative time line includes by definition 30 days and each "year" comprises 360 days (or 12 months) (see Section 2.2.3). These are, of course, only labels for durations, and no longer bear any relation to absolute calendrical months.

Note that the Set Relative Time Operator is not purely a syntactic operator, because, although the data type and domain of values remain the same, the output data values are typically quite different, all values and statistics being recalculated according to the new zero point.

The Set Relative Time Operator (SRTO) is formally defined as follows:

$$SRTO(GCM: <Concept_c, input\_data*, T\text{-}Gran_{aggreg},\ function>, GDM: <Concept_c, current\_values*, T\text{-}Start_{explor}, T\text{-}End_{explor}, T\text{-}Gran_{explor}, [\mathrm{Ref}P]>, \mathrm{Ref}P') \Rightarrow GDM': <Concept_c, new\_values'*, T\text{-}Start_{explor}, T\text{-}End_{explor}, T\text{-}Gran_{explor}, \mathrm{Ref}P'>,$$

where *RefP* is the user-determined parameter of the zero-point of the displayed data. Note that the GCM supports the computation of the new values; the aggregation ($T\text{-}Gran_{aggreg}$) and display parameters ($T\text{-}Start_{explor}$, $T\text{-}End_{explor}$, $T\text{-}Gran_{explor}$) are not changed. Only the visualized data are changed.

Before application of the Set Relative Time Operator, the $T\text{-}Start_{explor}$ and $T\text{-}End_{explor}$ time points have an absolute time value (e.g., 01.07.1995), but after application they have a relative time format (e.g., +1 month following the *RefP*, for example, a bone-marrow transplantation intervention). Figs. 10 and 11 show the application of the Set Relative Time Operator to the values of a raw and abstract concepts for a group of patients.

4.5.4. Other general exploration and documentation operators

Other visualization and exploration operators that are well described in our previous work regarding exploration of individual patient data [2] are also implemented in the VISITORS system. These operators include: (1) export of data to a spreadsheet, (2) performance of dynamic sensitivity analysis of possible changes to the values ('*What-if*' dynamic simulation) of the raw data, and their effect on the derived temporal abstraction for the individual patient (i.e., the user is able to simulate the effect of changing the derived temporal abstractions by modifying, deleting or adding

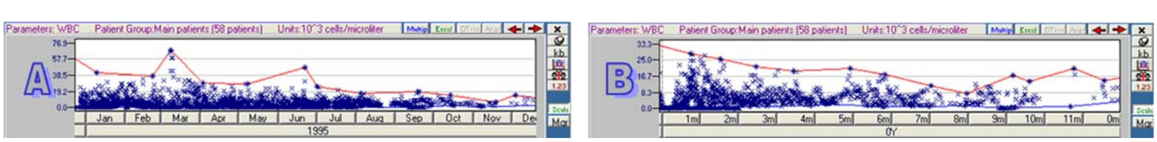

Fig. 10. Application of the Set Relative Time Operator to the data for the White blood cell (WBC) count raw concept for a group of 58 patients. Panel A displays the data for the year 1995 (the absolute time-line navigation bottom bars), and the panel B displays the data after application of the Set Relative Time Operator, showing the first year following the allogenic bone-marrow transplantation (selected by user), as shown in the new relative time-line navigation bottom bars. Note that absolute calendrical labels such as May 1995 no longer appear; instead, only relative temporal labels such as 5m (+5 months) can be seen.

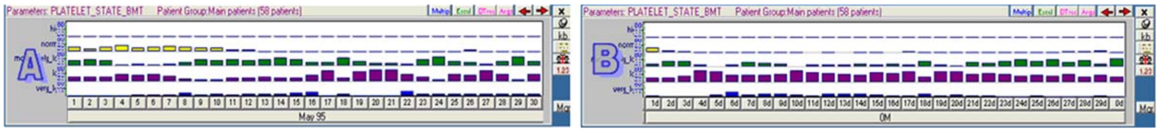

Fig. 11. Application of the Set Relative Time Operator to the data for the Platelet-state abstract concept for a group of 58 patients. Panel A displays the data for May 1995 (the absolute time-line navigation bottom bars), and panel B displays the data after application of the Set Relative Time Operator, showing the first month following the allogenic bone-marrow transplantation (selected by user), as shown in the new relative time-line navigation bottom bars. Note that absolute calendrical labels such as March 15th, 1995 no longer appear; instead, only relative temporal labels such as 8d (+8 days) can be seen.

raw data), (3) the exploration of the knowledge-based ontology (i.e., exploration of the properties of both the raw concepts and their abstractions, using meaningful domain-specific semantic relations such as *derived-from*, *part-of*), (4) support for the use of clinical profiles (e.g., a diabetic profile includes a set of diabetes-related raw and abstract concepts such as Blood-glucose value, Glucose-state level, HbA1C value, etc.), (5) support of intra-group collaboration capabilities, and (6) documentation operators.

4.6. Interactive visual exploration of temporal associations among time-oriented data of multiple patients

A *temporal association chart* (*TAC*) [57] is a new, user-driven, interactive knowledge-based visualization technique that supports the investigation of temporal and statistical associations within multiple patient records among both raw and abstract temporal concepts. A complete exposition of the computational semantics underlying TACS, as well as a detailed evaluation of their functionality and usability, appears elsewhere [57]. Here, we provide a summary of the core functionality that is relevant to the current study.

The core of the TAC is an ordered list of raw and/or abstract domain concepts (e.g., Platelet-state, Hemoglobin value, White blood cell count), designated a *temporal association template* (TAT), in a particular order determined by the user. Each concept is measured (or computed, in the case of an abstract concept) for a particular patient group during a particular concept-specific time period. The period can be different for each concept. Given the data for a group of patients, between each consecutive pair of concepts in the list, a relationship will be computed based on the delegate values of the concepts for each patient. If one of the concepts is raw, the result will be a set of relations, each relation being between a value of the first concept and a value of the second concept for a particular patient. If both concepts are abstract, the result will be aggregated into a set of extended relations—*temporal association rules*, one rule per each combination of values from both concepts, each rule representing the set of patients who have this particular combination of values for the two abstract concepts.

TACs are created by the user in two steps. First, the user creates the TAT, by the selection of two or more concepts, using an appropriate interface (not shown here), possibly changing the order as necessary; and second, the user selects the group of patients (e.g., from a list of groups retrieved earlier by *Select Patients* expressions). Although full exposition of TACs is beyond the scope of the current paper, we will briefly explain their core semantics.

4.6.1. Temporal association templates

A TAT is an ordered list of *time-oriented concepts* (TOCs) ($|\text{TOCs}| \geq 2$), in which each TOC denotes a combination of a raw or abstract domain concept (such as a Hemoglobin value or a bone-marrow toxicity grade) and a time interval <*T-Start*, *T-End*>. A specific concept can appear more than once in the TAT, but only within different time intervals. An example of a TAT listing the Hemoglobin-state and White blood cell-state abstract concepts and the Platelet count raw-data concepts, and their respective time periods, would be <(Hemoglobin-state, 1/1/95, 31/1/95), (WBC-state, 1/1/95, 31/1/95), (Platelet count, 1/1/95, 31/1/95), (WBC-state, 1/2/95, 28/2/95)>. Note that once a TAT is defined, it can be applied to different patient groups.

4.6.2. Application of a TAT to a set of patient records

When applying a TAT to a set *P* of patient records that includes *N* patients, we get a TAC. A TAC is a list of instantiated TOCs and of *association relations* (ARs), in which each instantiated TOC is composed of the original TOC of the TAT upon which it is based and the patient-specific delegate values for that TOC within its respective time interval, based on the actual values of the records in *P*. To be included in a TAC, a patient P_n ($1 \leq n \leq N$) must have at least one value from each TOC of the TAT defining the TAC. The group of such patients is the *relevant group* (or *relevant patients*). In the resulting TAC, each instantiated TOC_i includes the original TAT TOC_i and the set of delegate values (one delegate value for each patient) of the concept C_i, computed using the delegate function appropriate to C_i from the set of patient data included within the respective time interval [$T\text{-}Start_{\ i}$, $T\text{-}End_i$], as defined in the TAT.

4.6.3. Association relations

The relationship between the values of consecutive instantiated TOCs $<TOC_i, TOC_{i+1}>$, $1 \leq i < I$ (*I* is the number of concepts in the TAT) are denoted by ARs.

When at least one of the consecutive concepts is raw, the number of ARs between each pair of TOCs is equal to the number of relevant patients. Each AR connects the delegate values val_n^i and val_n^{i+1} of the pair of concepts C_i and C_{i+1}, during the relevant period of each concept, for one specific patient P_n.

In the case of an abstract-abstract concept pair, we aggregate the ARs between two consecutive TOCs into groups, where each group includes a set of identical pairs of delegate values (one pair for each concept). Each such group denotes a *temporal association rule* (TAR) and includes:

- *Support*: the proportion of relevant patients who have the combination of delegate values $<val_n^{i,j}, val_n^{i+1,k}>$, $1 \leq j \leq J$, $1 \leq k \leq K$, where $val_n^{i,j}$, $val_n^{i+1,k}$ are the *j*th and *k*th allowed values of C_i and C_{i+1}, respectively; and *J* and *K* are the numbers of different values of the concepts C_i and C_{i+1}, respectively. (We assume a finite number of [symbolic] values for each abstract concept.)
- *Confidence*: the fraction of the relevant patients who, given a delegate value $val_n^{i,j}$ of concept C_i for patient P_n, have a delegate value of concept C_{i+1} that is $val_n^{i+1,k}$; i.e., the probability $P[val_n^{i+1,k}|val_n^{i,j}]$.
- *Actual number of patients*: the number of patients who have this combination of values.

The number of possible TARs between two consecutive TOCs is thus *J***K*.

Note, that the definitions of the support and confidence properties are equal to support and confidence in the context of association rules, where performing general data mining.

4.7. Display of TACs and interactive data mining using TACs

Fig. 12 presents an example of a TAC computed by applying a TAT [*user-defined* on the fly, using another interface (not shown

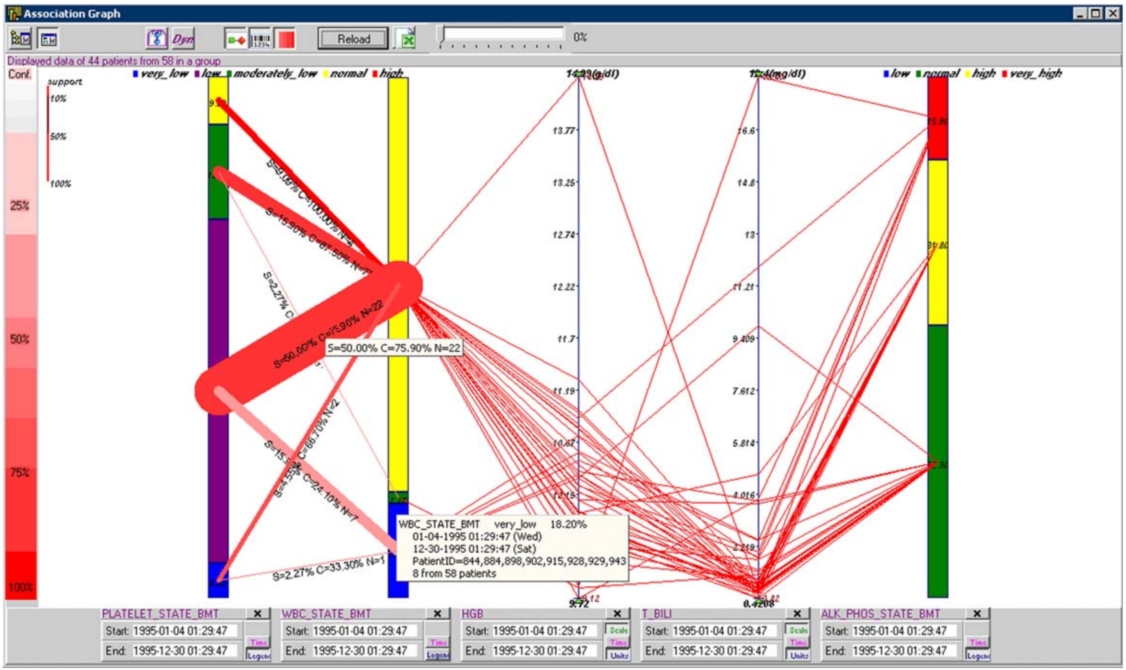

Fig. 12. Visualization of associations among three hematological and two hepatic concepts for 44 patients during the year 1995. Association rules are displayed between the Platelet-state and White blood cells (WBC)-state abstract concepts. The confidence and support scales are represented on the left.

here) that enables the user to select TAT concepts]. The TAT includes three hematological concepts (Platelet-state, White blood cells (WBC)-state abstract concepts, and the Hemoglobin (HGB) raw concept) and two hepatic concepts (Total bilirubin (T-Bili) and Alkaline-phosphatase (Alk-Phos)-state abstract concepts) applied to a group of 58 patients selected earlier by the user. For the abstract concepts, WBC-, Platelet-, and Alkaline-phosphatase states, the visualization in Fig. 12 shows the relative proportion (i.e., distribution) of all the values of the specific abstract concept for the relevant patients, within the specific time interval. It also shows each patient's mean values for Hemoglobin and Total bilirubin during the year 1995. Delegate values of all adjacent concept pairs for each patient are connected by lines, denoting the ARs. Only 44 patients in this particular group had data for all concepts during 1995.

As described above, ARs among values of abstract concepts provide additional statistical information. For example, the AR's width indicates to the user the support for each combination of values, while the color saturation represents the level of confidence: a deep shade of red signifies high confidence, while pink denotes lower confidence. The support, confidence, and the number of patients in each association are displayed numerically on the edge. For example, the widest edge among two distributions on the left hand side in Fig. 12 represents the relation between the "low" value of the Platelet-state concept and the "normal" value of the WBC-state concept during 1995. The edge shows that 50.0% of all of the patients in the relevant patient group had this, particular, combination of values during 1995 (i.e., support = 0.500). 75.9% of the patients who had a "low" Platelet-state value have also had a "normal" WBC-state value during 1995 (i.e., confidence = 0.759). This association was valid for 22 patients. Note that in this particular case, the two periods were identical (the year 1995). However, the user could have asked to visualize ARs that link different time periods (e.g., different years or different months within the same or another year).

Using this visualization interface, the user can dynamically apply a *value* and *time lens* to interactively analyze the time and value associations among multiple patients' data:

- Dynamic application of a value lens enables the user to answer the question "how does constraining the value of one concept during a particular time period affect the association between multiple concepts during that and/or during additional time periods". The user can either select another range of values for the data of the raw concepts, using trackbars, or select a subset of the relevant values in the case of an abstract concept. In the future versions of VISITORS, we are planning to allow the user to vary also the delegate function to enable additional analyses.
- The system also supports the application of a *time lens* by changing the range of the time interval for each instantiated TOC, including ranges on the relative time line. The time lens can be especially useful for clinical research involving longitudinal monitoring (e.g., clinical trials). For example, the researcher can investigate the relation between the values among several concepts before and after treatment.

In addition, the user can change the order of the displayed concepts, export all the visualized data and associations to an electronic spreadsheet, and add or remove displayed concepts.

The main limitation of TACs in the current version of the VISITORS system is that the system does not recommend *which* concepts to select, nor the time periods in which to examine the edges (ARs) among them. However, we intend to combine the VISITORS system with temporal data mining tools (that we have been developing [58]) for automated detection of sufficiently frequent temporal associations.

5. Example of a clinical scenario

In this section, we present an example of an exploration clinical scenario for application of the VISITORS system and an analysis using a TAC. The example, which relates to a retrospective database of bone-marrow transplantation (BMT) patients (see evaluation Section 6), comprises an *investigation of the bone-marrow recovery characteristics of patients, who are either young* [<20 *years*] *or old* [>70 *years*], *following an autologous BMT procedure*.

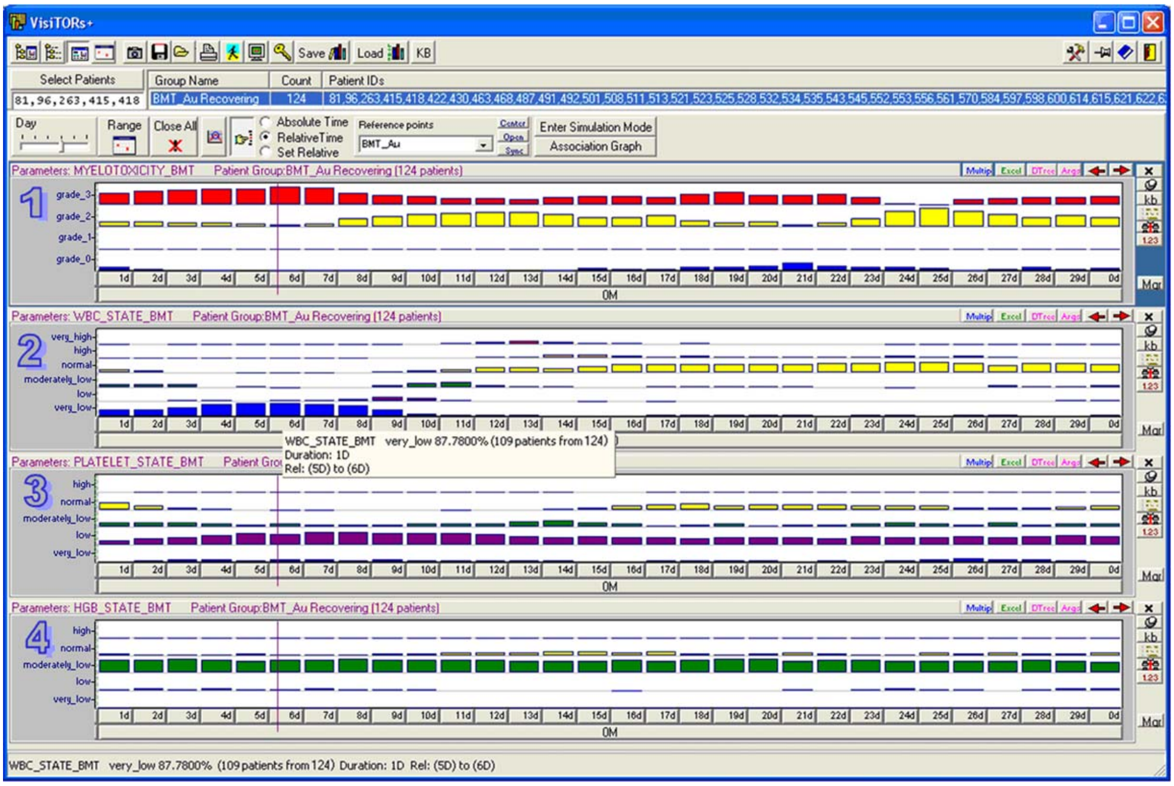

Fig. 13. Exploration of the data of 124 patients for the Myelotoxicity-, WBC-, Platelet-, and HGB-state abstract concepts (denoted by 1–4 from the top to the bottom panel). See the text for details.

(1) In a previous study [4] we introduced the graphical interfaces for construction of the *Select Patients* expression "Select all male patients, either younger than 20 years or older than 70 years, whose value of the Hemoglobin (HGB) state abstract concept was at least "moderately-low" or higher, for at least ten days, during the first month following an autologous BMT and whose White blood cell (WBC) counts were abstracted as "increasing" during those ten days". As a result of applying this *Select Patients* expression, a group of patients who recovered from the autologous BMT (designated as *BMT_Au Recovering* by the clinician) was returned and saved. Our current scenario continues from that point.

(2) In the second stage, the clinician explores the data of 124 patients in the "BMT_Au Recovering" group of Myelotoxicity-, WBC-, Platelet-, and HGB-state abstract concepts (Fig. 13) during the first month following the autologous BMT.

It is clear from the first (top) panel, i.e., Myelotoxicity-state concept panel, that until the sixth day after the BMT, the portion of patients with a high value of the myelotoxicity-state concept (grade_3) was relatively high and increased from day 1 to day 6. During these first six days, the portion of patients with the "very-low" value of the WBC-state concept also increased each day (up to 87.78% of patients in the group by the sixth day), as shown in the tooltip of the second panel from the top.

Starting from the sixth day following the transplantation, the portion of patients in the group with values of the WBC-state concept higher than "very-low" began to increase, and by the twelfth day, most patients had "normal" or higher values of the WBC-state, as shown in panel 2.

From the third panel, i.e., the Platelet-state concept panel, we can visually conclude that during the first month following the transplantation at least 50% of the patients had a daily "low" or "moderately-low" value of the Platelet-state concept. From the tenth day onwards, the portion of patients with values of Platelet-state higher than "low" appeared to increase.

Exploration of the fourth panel, i.e., the HGB-state concept panel, shows that during the first month following the transplantation most of the patients had a "moderately-low" HGB-state concept.

Thus, given the data for the selected 124 patients, by following a simple exploration process, we can conclude that: (1) the post-BMT recovery process for most patients seemed to have started on the tenth day following the BMT; and (2) the HGB-state concept did not seem to be affected by the BMT procedure or by the recovery of the WBC and Platelet counts.

(3) Finally, the clinician creates a TAC to examine the relationships among the delegate values of the 124 patients for the above four abstract concepts during the first two weeks following the BMT (Fig. 14). Note that for only 91 patients in the group were there values for all four concepts during the desired time period.

During the period of the first two weeks following the BMT, most patients had a high-grade Myelotoxicity level, a very-low WBC-state, and a low Platelet-state. Moreover, for a patient with a "grade_3" value of the Myelotoxicity-state concept, there was a 92.3% probability of a "very-low" value of the WBC-state abstract concept. Similarly, given a "very-low" value of the WBC-state concept, 86.8% of the patients had a "low" value of the Platelet-state concept.

From the TAC in Fig. 14, we can conclude that the characteristic "profile" of most of the 124 patients was a combination of a "grade_3" value of the Myelotoxicity-state, a "very-low" value of the WBC-state, a "low" value of the Platelet-state, and a "moderately-low" value of the HGB-state. If each concept were

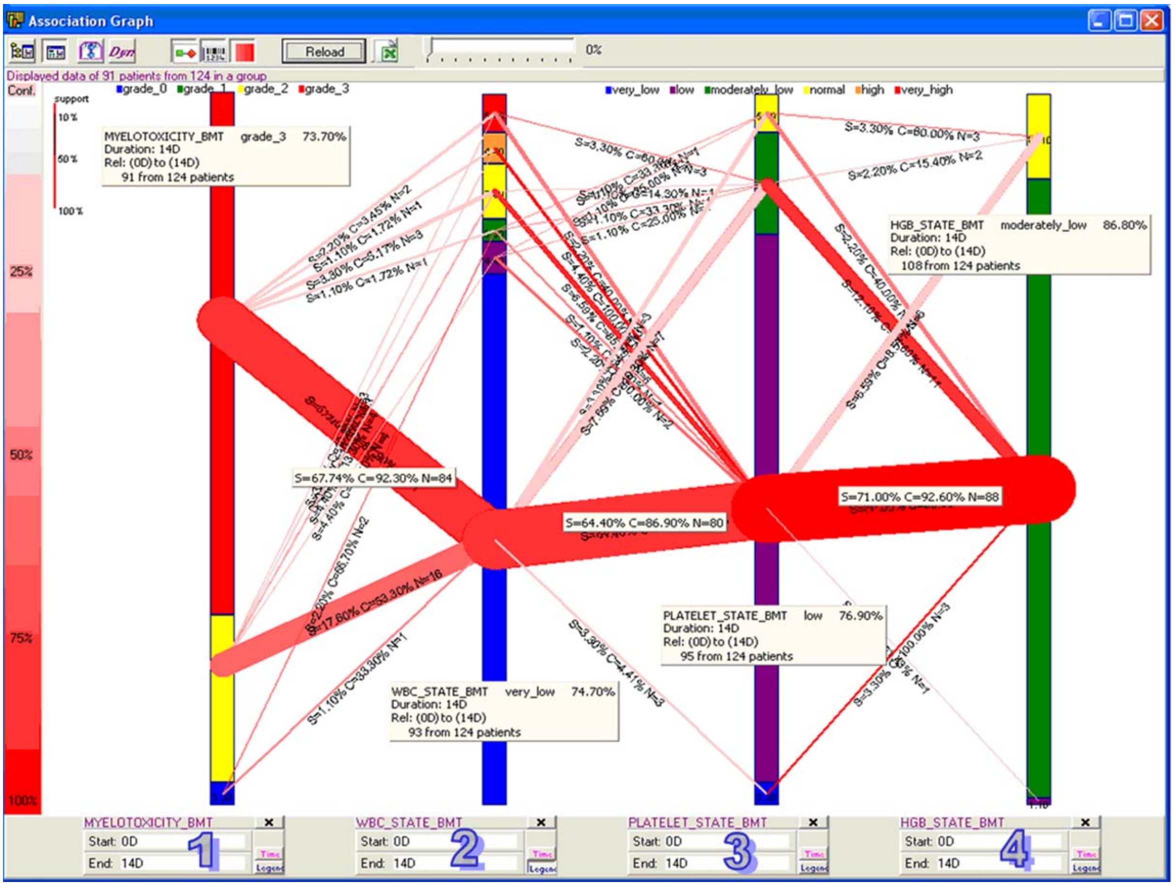

Fig. 14. Visualization of associations among the myelotoxicity-, WBC-, Platelet-, and HGB-state abstract concepts (denoted by 1–4 from the left to the right) for 124 patients during the first two weeks following BMT. See the text for details.

measured at a different period, we would get a *characteristic temporal path*.

6. Evaluation of the functionality and usability of the VISITORS system

6.1. Research questions

We designed and developed the VISITORS system according to the desiderata listed in Section 3, and envision it as potentially useful for two types of users: clinicians and medical informaticians. We also envision that the system can be used to answer different clinically motivated questions. We conducted an evaluation of the system with the aim of answering the following four research questions:

1. Overall functionality and usability: are the interactive exploration operators of the VISITORS system feasible, functional, and usable?
2. Effect of the interaction mode: it there a significant difference in the accuracy or in the time to answer between the answers obtained using the operators of the general exploration mode and those obtained using the operators of the TACs?
3. Effect of the time line: is there a significant difference in the accuracy of the clinical scenarios or in the time to answer using the two types of time line (an absolute, i.e., calendar time line, versus a relative time line, i.e., a time line referenced to an external significant clinical time point, such as a therapeutic intervention), and if so, which time line leads to a lower or a higher level of accuracy and time to answer?
4. Effect of the user group: can both medical informaticians (e.g., information system engineers who work in the medical informatics domain) and clinicians use the system effectively?

6.2. Measurement methods and data collection

To the best of our knowledge, there is no known method that is functionally equivalent to the VISITORS system. Furthermore, with current methods users can answer the complex questions for which VISITORS was designed only through the performing the laborious computations. Thus, we have chosen an *objective-based approach* [59] for evaluation of the VISITORS system. In such an approach, certain reasonable objectives are defined for a new system, and the evaluation strives to demonstrate that these objectives have been achieved. In this case, we set out to prove certain functionality and usability objectives of the VISITORS system.

The evaluation of the VISITORS system was performed in the oncology domain. In all the tests, we used a retrospective database of more than 1000 oncology patients who had received bone-marrow transplants and who had been followed-up for 2–4 years. The knowledge source used for the evaluation was an oncology knowledge base specific to the bone-marrow transplantation domain; the source was acquired with the help of a domain expert, as reported in a previous study [60].

Ten participants, five medical informaticians, i.e., the information system engineers who work in the medical informatics domain, and five clinicians with different medical training levels, were asked to answer ten clinical questions: five questions

Table 1
Examples of clinical questions used in the evaluation, ordered informally by level of complexity.

Category	Examples of questions
General exploration operators	• What was the mean (i.e., average) annual value of the Hemoglobin (HGB) raw data concept during each of the years 1995, 1996, and 1997? • What was the distribution of the values of the Platelet-state abstract concept during the first and second months (i.e., relative time) following the bone-marrow transplantation (BMT) procedure? • What was the distribution of the aggregate values of the Platelet-state abstract concept during the first month after BMT? Who (i.e., which patient) had the maximal delegate value of the White blood cell (WBC) count raw concept and which patient had the minimal delegate value of Red blood cells (RBC) count?
Temporal association charts	• What delegate value of the HGB-state abstract concept was the most frequent among the patients who previously had a "low" aggregate value of the Platelet-state abstract concept? • What was the distribution of the delegate values of the Platelet-state in patients whose minimal delegate value of the WBC count raw concept was 5000 cells/ml (instead of the previous minimal value)? What were the new maximal and minimal delegate values of the RBC? • What percentage of the patients had previously had a "low" delegate value of the Platelet-state abstract concept during both the first and second month following the BMT?

required the use of the general exploration operators of VISITORS, and five questions required the use of TACs. None of the study participants was a member of the VISITORS development team. The ten questions were selected in consultation with oncology domain experts. They represented typical questions relevant to the monitoring of a group of oncology patients or to the analysis of an experimental protocol in oncology. Examples of the questions are presented in Table 1. The order of the questions (in each of the two categories according to the interaction mode) was permuted randomly across participants. Each evaluation session with a participant started with a 20-min tutorial, which included a brief description of the KBTA methodology [1] and of the general exploration and TAC operators. A demonstration of the general and TAC operators was given, showing how several typical clinical questions can be answered. The scope of the instruction was predetermined and included (after the demo) testing each participant by asking him/her to answer three clinical questions, one of which included the use of TACs. When the participant could answer the questions correctly, he/she was considered ready to perform for the evaluation.

The *functionality* was assessed using two parameters: the time in minutes needed to answer the question, and the accuracy of the answer. The scale determining the accuracy of each answer was pre-defined before the start of the study with the help of a medical expert. The accuracy was measured on a scale from 0 (completely wrong value) to 100 (completely correct value).

To test the *usability* of VISITORS system, we used the system usability scale (SUS) [61], a common validated method to evaluate interface usability. The SUS is a questionnaire that includes ten predefined questions regarding the effectiveness, efficiency, and satisfaction from an interface. SUS scores have a range of 0–100. Informally, a score higher than 50 is considered to indicate a usable system.

7. Results

This section summarizes the evaluation results in terms of the four research questions defined in Section 6.1.

7.1. Overall functionality and usability

7.1.1. Method of measurement

The effectiveness of the users in answering clinical questions using the general exploration operators and TACs was assessed by calculating the overall means and standard deviations of the answer accuracy and of the answer response time. To test the usability of the system, the SUS questionnaire was used. To compare the usability scores of the two groups of participants, a *t*-test was performed.

7.1.2. Results

Table 2 summarizes the response times and answer accuracy levels for all participants.

Most of the participants (9 of 10) successfully answered the clinical questions with a mean accuracy of more than 96 (out of 100); 6 of them had a mean accuracy of 100. One participant had a mean accuracy 90.5. All participants answered the clinical questions in a mean time of less than 3 min.

The mean SUS score for all operators, across all participants, was 69.3 (over 50 is usable). The results of a t-test analysis showed that the mean SUS score of the medical informaticians (80.5) was significantly higher than that of the clinicians (58): [$t(8) = 3.88$, $p < 0.01$].

7.1.3. Conclusion

Based on the results of the VISITORS evaluation, we may conclude that, after a short training period, the participants were able to answer the clinical questions with high accuracy and within short period of time. The SUS scores showed that VISITORS system is usable but still needs to be improved.

7.2. Effects of the interaction mode, time line and user group

To answer research questions 2, 3 and 4, i.e., questions regarding the interaction mode effect, effect of the time line, and the user group effect, a joint analysis was performed, as explained below.

Table 2
Accuracy and response times for all participants.

		General exploration operators	Temporal association charts	Overall
Accuracy score (0–100)	Mean accuracy	99.5 ± 1.6	97.9 ± 3.4	98.7 ± 2.4
	Range of mean accuracy per question across all participants	[97.5...100]	[95.0...100]	[95.0...100]
	Range of mean accuracy per participant across all questions	[95.0...100]	[90.5...100]	[93...100]
Response time (min)	Mean time	2.2 ± 0.2	2.7 ± 0.4	2.5 ± 0.2
	Range of mean time per question across all participants	[2.2...2.4]	[2.4...3.0]	[2.2...3.0]
	Range of mean time per participant across all questions	[2.0...2.6]	[2.2...3.6]	[2.2...3.0]

7.2.1. Method of measurements

The effects of the interaction mode (i.e., general exploration operators and TACs), time line (i.e., absolute and relative times), and the group of participants (i.e., medical informaticians and clinicians) on the dependent variables of response time and accuracy of answering were examined using two different three-way ANOVA tests with repeated measures (one for each dependent variable). The interaction mode and the time-line type were within-subject independent variables, and the group of subjects was a between-subjects independent variable. Since we did not find statistically significant differences among the response times to different clinical questions (and among resultant accuracy levels of answering these questions) of the same time line using the same interaction mode, the mean value of the response time (and of the accuracy) of the clinical questions of the same time line was used as the dependent variable.

7.2.2. Results

The results of the analysis of the accuracy showed no significant effects ($p > 0.05$) of the interaction mode, time line, or of the group of participants. With respect to the response time, the results of the analysis showed that the only significant effect was the main effect of the type of interaction mode [$F(1, 8) = 11.08$, $p < 0.05$], i.e., a mean of 2.2 ± 0.2 min for answering the clinical questions when using the general exploration operators of VISITORS, and a mean of 2.7 ± 0.4 min for answering the clinical questions using the TACs. There was no significant difference in the response time either of the time line or of the group of participants.

7.2.3. Conclusion

Interaction mode, time line and user type did not affect the accuracy of the answers to clinical questions. The mean time needed to answer the clinical scenarios using the TACs was significantly higher than that for the general exploration operators of VISITORS, but it was still less than 3 min. The time line and the user type did not affect the response time of the answers to the clinical questions.

8. Discussion

8.1. Contributions and advantages

The major contribution of the VISITORS system is the provision of a comprehensive environment for intelligent, i.e., knowledge-based, investigation of time-oriented data for multiple patients, including the specification and retrieval, *visualization*, *exploration*, and *analysis* of the time and value associations among both raw and abstract clinical concepts. Based on the results of the current study, the VISITORS system might be described as an "*intelligent equalizer*" for data interpretation and exploration, which results in a uniform performance level, regardless of the patient-specification complexity, the exploration question or mode, or the user type. A similar insight has emerged from our previous study, regarding assessment of the usability and functionality of the OBTAIN language expression-specification tool [4]. In that experiment, both medical informaticians and clinicians have constructed complex expressions quickly and efficiently, although the clinicians found the interface less usable than the informaticians.

The VISITORS system can access diverse types of temporal abstraction knowledge and clinical data. Moreover, the VISITORS system is quite generic and can support exploration within non-medical domains, for example, in the information security domain [62].

Somewhat surprisingly, exploration of data of multiple patients using a relative time line (i.e., relative to a meaningful reference event, or to another clinically significant time point) did not seem to be more difficult to either clinicians and informaticians than exploration using an absolute (i.e., calendrical) time line. A possible explanation might be that there is no semantic difference between "the first day of December" and "the first day after bone-marrow transplantation"; thus, it is equally easy for people to explore data within absolute and relative time lines. Furthermore, clinicians are trained to think along relative time lines, such as "the fourth month of pregnancy" or the "second day of therapy". However, using TACs to answer questions, although resulting in equal accuracy levels, required significantly higher response times. These results probably reflect the highly unusual TACs interface and data mining parameters (i.e., support and confidence), which unlike the relative time line, are qualitatively different from the general exploration operators and exploration interfaces. A possible reason for the lack of significant differences in the accuracy scores when using different interaction modes was that the evaluation included a relatively small group of participants and questions. However, it should be noted that the variance among accuracy scores was quite low for both interaction modes, all of the participants achieving scores above 90. Thus, the absence of a significant effect could not be attributed to random differences and high variability in each interaction mode.

8.2. Limitations and future work

Although the user interfaces of the VISITORS system were initially considered by the users as quite complex, most participants, during the population specification and exploration evaluations successfully finished all the evaluation tasks. After the training session and the learning of the principles of the system, the system was considered less complex than before. However, the participants noted that the system provides more functionality than will usually be exploited by typical clinicians or even clinical researchers. Thus, in the future, the main user interface should be simplified by providing only the basic functions, while more complex features (such as time-range constraints regarding the start or end of time intervals) would be enabled only in an advanced mode.

The VISITORS system can also be improved by adding capabilities for interactive specification of the patient population during the exploration process (in the current state, selection and retrieval of patients is a separate process). The visual exploration operators can be enhanced by adding more intelligence to the semantic zoom, e.g., increasing the temporal granularity level of the display (e.g., from the day to month) will cause a display of higher level abstractions when appropriate. Furthermore, "typical day" (modal day) or other granularity visualizations can be added, which is very important for periodic domains such as diabetes. Finally, a visual comparison of several population groups, based on one or more specific concepts (for clinical trials or quality assessment tasks), would be quite useful. We intend to explore all of these options in our future work.

To summarize, we conclude that intelligent retrieval and exploration of longitudinal data for multiple patients using the VISITORS system is feasible, functional, and usable. Future work is needed to extend the visual exploration operators and to assess the value of such extensions to the users.

Acknowledgments

This research was supported by Deutsche Telekom Labs at Ben-Gurion University of the Negev and the Israel Ministry of Defense, BGU award No. #89357628-01. We thank all the clinicians and medical informaticians who contributed their time to the evaluation. We thank Ms. Efrat German for her work on the Tempura system, and Mr. Ido Hacham and Mr. Shahar Albia for their work on the Multi-TOQ system.

Appendix A. BNF syntax for a *Select Patients* expression

select-patients-expression: <data-base> <knowledge-base> <patient-constraints>
patient-constraints: <demographic-constraints>* <time-and-value-constraints>+ | demographic-constraints>+ <time-and-value-constraints>*
demographic-constraints: <selection-condition>+
time-and-value-constraints: <local-constraints>+ [<global-pairwise-constraints>+]
selection-condition: <attribute-name> <min value> <max value>
local-constraints: <concept-name> <value-constraints > <time point-constraints> [<duration-constraints>] [<relative-time-constraints>] [<proportion-constraints>] [<statistical-constraints>]
global-pairwise-constraints: <value-pairwise-constraints>* [<temporal-pairwise-constraints>*]
value-constraints: <min-value> <max-value>
necessary-context: <context-name>
time point-constraints: <start-point, end-point> [<earliest-start-point> <latest-start-point>] [<earliest-end-point> <latest-end-point>]
duration-constraints: <min-duration> <max-duration>
relative-time-constraints: <relative-start-point> <relative end point> [<relative-earliest-start-point> <relative-latest-start-point>] [<relative-earliest-end-point> <relative-latest-end-point>]
proportion constraints: <min-threshold> <max-threshold>
statistical constraints: <individual-patient-delegate-function> <population-delegate-function> <value-relation> [<delta>]
value-pairwise-constraints: <first-concept name> <second-concept name> <time-relation> < individual-patient-delegate-function>
temporal-pairwise-constraints: <first-concept-name> <second-concept-name> <first-boundary-time point> <second-boundary-time point> <value-relation> [<difference>]
individual-patient-delegate-function: <maximal> | <minimal> | <mean>
population-delegate-function: <maximal> | <minimal> | <mean>
value-relation: <great-than-equal> | <great-than> | <less-than> | <less-than-equal>
time-relation: <before> | <after> | <starts> | <ends> | <within> | <meets> | <overlaps> | <equal>
difference: <min-difference> <max-difference>

Appendix B. BNF syntax for a *Select Time Intervals* expression

select-time-intervals-expression: <data-base> <knowledge-base> <interval-constraints>
interval-constraints: <granularity> <concept-constraints>* [<time-constraints> | <relative-time-constraints>]
granularity: <seconds> | <minutes> | <hours> | <days> | <months> | <years>
concept-constraints: <concept-name><population-thresholds> <value-constraints> <individual-patient-delegate-function>
time point-constraints: <start-point, end-point> [<earliest-start-point> <latest-start-point>] [<earliest-end-point> <latest-end-point>]
relative-time-constraints: <relative-start-point> <relative end point> [<relative-earliest-start-point> <relative-latest-start-point>] [<relative-earliest-end-point> <relative-latest-end-point>]
population-thresholds: <min-threshold> <max-threshold>
value-constraints: <min-value> <max-value>
individual-patient-delegate-function: <maximal> | <minimal> | <mean>

Appendix C. BNF syntax for a *Get Patient Data* expression

get-patients-data-expression: <data-base> <knowledge-base> < concept-name > <patient-list> [<interval-list>]

References

[1] Shahar Y. A framework for knowledge-based temporal abstraction. Artif Intell Med 1997;90(1–2):79–133.
[2] Shahar Y, Goren-Bar D, Boaz D, Tahan G. Distributed, intelligent, interactive visualization and exploration of time-oriented clinical data and their abstractions. Artif Intell Med 2006;38(2):115–35.
[3] Martins S, Shahar Y, Goren-Bar D, Galperin M, Kaizer H, Basso L, et al. Evaluation of an architecture for intelligent query and exploration of time-oriented clinical data. Artif Intell Med 2008;43(1):17–34.
[4] Klimov D, Shahar Y, Taieb-Maimon M. Intelligent selection and retrieval of multiple time-oriented records. Journal of Intelligent Information Systems; in press. doi:10.1007/s10844-009r-r0100-0.
[5] Hearst M. User interfaces and visualization. In: Baeza Yates R, Ribeiro-Neto B, editors. Modern information retrieval. NY: ACM Press; 2000. p. 257–324 [Chapter 1].
[6] Goldstein M, Hoffman B. Graphical displays to improve guideline-based therapy of hypertension. In: Izzo Jr JL, Black HR, editors. Hypertension primer. 3rd ed., Baltimore: Lippincot, Williams & Wikkins; 2003.
[7] Silvent A-S, Dojat M, Garbay C. Multi-level temporal abstraction for medical scenario construction. Int J Adapt Control 2005;19:377–94.
[8] Miksch S, Horn W, Popow C, Paky F. Utilizing temporal data abstraction for data validation and therapy planning for artificially ventilated newborn infants. Artif Intell Med 1996;8:543–76.
[9] Cooley J, Tukey J. An algorithm for the machine calculation of complex Fourier series. Math Comput 1965;19(90):297–301.
[10] Miksch S, Seyfang A, Popow C. Abstraction and representation of repeated patterns in high-frequency data. In: The fifth workshop on intelligent data analysis in medicine and pharmacology (IDAMAP-2000), workshop notes of the 14th European conference on artificial intelligence (ECAI-2000); 2000.p. 32-9.
[11] Combi C, Chittaro L. Abstraction on clinical data sequences: an object-oriented data model and a query language based on the event calculus. Artif Intell Med 1999;17:271–301.
[12] Kowalaski R, Sergot M. A logic-based calculus of events. New Gener Comput 1986;4:67–95.
[13] Chittaro L. Information visualization and its application to medicine. Artif Intell Med 2001;22(2):81–8.
[14] Plaisant C, Milash B, Rose A, Widoff S, Shneiderman B. LifeLines: visualizing personal histories. In: Proceedings of ACM CHI '96 Conference. New York, NY, USA: ACM Publisher; 1996. p. 221–7. ISBN:0-89791r-r777-4.
[15] Plaisant C., Mushlin R., Snyder A., Li J., Heller D., Shneiderman B. LifeLines: using visualization to enhance navigation and analysis of patient records. Revised version appeared in American Medical Informatics Association Annual Fall Symposium, 1998: 76–80.
[16] Wang T, Plaisant C, Quinn A, Stanchak R, Shneiderman B, Murphy S. Aligning temporal data by sentinel events: discovering patterns in electronic health records. In: Proc. ACM Conf. on human factors in computing systems (CHI2008), ACM. New York, NY, USA: ACM Publisher; 2008 . p. 457–66. ISBN:978-1-60558r-r011-1.
[17] Spenke M. Visualization and interactive analysis of blood parameters with InfoZoom. Artif Intell Med 2001;22(2):159–72.
[18] Falkman G. Information visualisation in clinical Odontology: multidimensional analysis and interactive data exploration. Artif Intell Med 2001;22(2):133–58.
[19] Chittaro L, Combi C, Trapasso G. Data mining on temporal data: a visual approach and its clinical application to hemodialysis. J Vis Lang Comput 2003;14(6):591–620.
[20] Aigner W, Miksch S, Müller W, Schumann H, Tominski C. Visual methods for analyzing time-oriented data. IEEE Trans Visual Comput Graph 2008;14(1): 47–60.
[21] Navathe S, Ahmed R. Temporal extensions to the relational model and SQL. In: Tansel AU, Clifford J, Gadia S, Segev A, Snodgrass R, editors. Temporal

Databases: Theory, Design, and Implementation. Redwood City, CA: Benjamin/Cummings; 1993 . p. 6–27.
[22] Sarda N. HSQL: A historical query language. In: Tansel A, Clifford J, Gadia S, Jajodia S, Segev A, Snodgrass R, editors. Temporal Databases: Theory, Design, and Implementation. Redwood City, CA: Benjamin/Cummings; 1993. p. 110–40.
[23] Snodgrass R. The TSQL2 temporal query language. Hingham, MA: Kluwer; 1995, ISBN 0-7923-9614-6.
[24] Combi C, Pinciroli F, Cavallaro M, Cucchi G. Querying temporal clinical databases with different time granularities: the GCH-OSQL language. In: Gardner RM, editor. 19 Annual symposium on computer applications in medical care. Philadelphia: Hanley & Belfus; 1995. p. 326–30.
[25] Combi C, Cucchi G. GCH-OSQL: a temporally-oriented object-oriented query language based on a three-valued logic. In: Proceedings of the Fourth International Workshop on Temporal Representation and Reasoning, 1997 (TIME'97), 119-126, 1997. ISBN:0-8186-7937-9.
[26] Combi C, Cucchi G, Pinciroli F. Applying object-oriented technologies in modeling and querying temporally-oriented clinical databases dealing with temporal granularity and indeterminacy. IEEE Trans Inform Technol Biomed 1997;1(2):100–27.
[27] Das A, Musen M. A temporal query system for protocol directed decision support. Methods Inform Med 1994;33(4):358–70.
[28] Combi C, Montanari A, Pozzi G. The t4sql temporal query language. In: Mario J, Silva, Alberto HF, Laender, Ricardo A, Baeza-Yates, Deborah L, McGuinness, Bjorn Olstad, Oystein Haug Olsen, Andre O, Falcao, editors. Proceedings of the sixteenth ACM conference on information and knowledge management, CIKM 2007. 2007. p. 193–202.
[29] Combi C, Montanari A. Data models with multiple temporal dimensions: completing the picture. In: Klaus R i, Andreas Geppert, Moira C o, editors. CAiSE, volume 2068 of LNCS:187-202, 2001.
[30] Hibino S, Rudensteiner E. A visual multimedia query language for temporal analysis of video data. Multimedia database systems: design and implementation strategies. Kluwer Academic Publishers; 1995, 123–59.
[31] Silva S, Catarci T, Schiel U. Formalizing visual interaction with historical databases. Inform Systs; 2002;27(7):487–521.
[32] Dionisio J, Cardenas A. MQuery: a visual query language for multimedia, timeline and simulation data. J Visual Lang Comput 1996;7:377–401.
[33] Hochheiser H, Shneiderman B. Visual specification of queries for finding patterns in time-series data. In: Proceedings of discovery science; 2001. p. 441–6.
[34] Hochheiser H, Shneiderman B. Dynamic query tools for time series data sets: timebox widgets for interactive exploration. Inform Visual Spring 2004;3(1): 1–18.
[35] Chittaro L, Combi C. Visualizing queries on databases of temporal histories: new metaphors and their evaluation. Data Knowl Eng 2003;44(2):239–64.
[36] Chu W, Chih-Cheng H, Cardenas A, Taira R. Knowledge-based image retrieval with spatial and temporal constructs. IEEE Trans Knowl Data Eng 1998;10(6):872–88.
[37] Bresciani P, Nori M, Pedot N. QueloDB: a knowledge based visual query system. In: Proceedings of the 2000 international conference on artificial intelligence IC-AI 2000, vol. III; 2000.
[38] Dumas M, Fauvet M-C, Scholl P-C. Handling temporal grouping and pattern-matching queries in a temporal object model. In: Georges Gardarin, James C, French, Niki Pissinou, Kia Makki, Luc Bouganim, editors. Proceedings of the 1998 ACM CIKM international conference on information and knowledge management. 1998. p. 424–31.
[39] Clifford J, Croker A, Grandi F, Tuzhilin A. On Temporal Grouping. Temporal Databases. In: Clifford J, Tuzhilin A, editors. Recent Advances in Temporal Databases, Proceedings of the International Workshop on Temporal Databases, Zurich, Switzerland, 17-18 September 1995. Springer, Workshops in Computing; 1995. p. 194–213.
[40] Moon B, Lupez F, Vijaykumar I. Efficient algorithms for large-scale temporal aggregation. IEEE Trans Knowl Data Eng 2003;15(3):744–59.
[41] Goralwalla I, Leontiev Y, Özsu T, Szafron D, Combi C. Temporal granularity: completing the puzzle. J Intell Inf Syst 2001;16(1):41–63.
[42] Goralwalla I, Leontiev Y, Özsu T, Szafron D, Combi C. Temporal granularity for unanchored temporal data. Proceedings of the seventh international conference on Information and knowledge management, November 02–07, 1998, Bethesda, Maryland, United States. New York, NY, USA: ACM Publisher, 414–23, ISBN:1-58113r-r061-9.
[43] Clifford J, Rao A. A simple, general structure for temporal domains. Temporal Aspects Inform Systs 1987;17–28.
[44] Dal Lago U, Montanari A. Calendars. Time granularities, and automata. Proceedings of the 7th international symposium on advances in spatial and temporal databases, 12–15, 2001, Lecture notes in computer science, volume 2121/2001. Springer Berlin/Heidelberg, 279–98, ISBN:978-3-540-42301-0.
[45] Combi C, Pinciroli F, Pozzi G. Managing different time granularities of clinical information by an interval-based temporal data model. Methods Inform Med 1995;34:458–74.
[46] Bettini C, Mascetti S, Pupillo V. A system prototype for solving multi-granularity temporal CSP. In: Recent advances in constraints, revised selected papers from the workshop on constraint solving and constraint logic programming (CSCLP) volume 3419 of lecture notes in computer science. Springer; 2005. p. 142–56.
[47] Bettini C, Wang X, Jajodia S. Solving multi-granularity temporal constraint networks. Artif Intell 2002;140(1/2):107–52.
[48] Bettini C, Mascetti S, Wang X. Mapping calendar expressions into periodical granularities. Proc. of the 11th international symposium on temporal representation and reasoning (TIME). IEEE Computer Society; 2004, p. 96–102, ISBN:0-8186-7937-9.
[49] Nigrin D, Kohane IS. Temporal expressiveness in querying a time-stamp–based clinical database. J Am Med Inform Assoc 2000;7:152–63.
[50] Narayanan A, Shaman T. Iconic SQL: practical issues in the querying of database through structured iconic expressions. J Visual Lang Comput 2002;13(6):623–47.
[51] Sacchi L, Larizza C, Combi C, Bellazzi R. Data mining with temporal abstractions: learning rules from time series. Data Min Knowl Discov 2007;15(2): 217–47.
[52] Boaz D, Shahar Y. A distributed temporal-abstraction mediation architecture for medical databases. Artif Intell Med 2005;34(1):3–24.
[53] Shahar Y, Musen M. Knowledge-based temporal abstraction in clinical domains. Artif Intell Med 1996;8(3):267–98.
[54] Spokoiny A, Shahar Y. Active database architecture for knowledge-based incremental abstraction of complex concepts from continuously arriving time-oriented raw data. J Intell Inform Syst 2007;28(3):199–231.
[55] Spokoiny A, Shahar Y. Incremental application of knowledge to continuously arriving time-oriented data. J Intell Inform Syst 2008;31(1):1–33.
[56] Shahar Y. Knowledge-based temporal interpolation. J Exp Theor Artif Intell 1996;11(1):123–44.
[57] Klimov D, Shahar Y, Taieb-Maimon M. Intelligent interactive visual exploration of temporal associations among multiple time-oriented patient records. Methods Inform Med 2009;48(3):254–62.
[58] Moskovitch R, Shahar Y. Temporal data mining based on temporal abstractions. In: ICDM-05 workshop on temporal data mining; 2005.
[59] Friedman C, Wyatt J. Evaluation methods in medical informatics. New York: Springer; 1997.
[60] Chakravarty S, Shahar Y. Specification and detection of periodicity in clinical data. Methods Inform Med 2001, 40(5):410–20. [Reprinted in: Haux R, Kulikowski C., editors. Yearbook of Medical Informatics 2003, Stuttgart: F.K. Schattauer and The International Medical Informatics Association, forthcoming].
[61] Brooke J. SUS: a "quick and dirty" usability scale. In: Jordan PW, Thomas B, Weerdmeester BA, McClelland AL, editors. Usability evaluation in industry. London: Taylor and Francis; 1996.
[62] Shabtai A, Klimov D, Shahar Y, Elovici Y. An intelligent, interactive tool for exploration and visualization of time-oriented security data. In: Conference on computer and communications security. Proceedings of the 3rd international workshop on visualization for computer security; 2006. p. 15–22.