The Five Ws for Information Visualization with Application to Healthcare Informatics

Zhiyuan Zhang, *Student Member*, *IEEE*, Bing Wang, Faisal Ahmed, I.V. Ramakrishnan, Rong Zhao, Asa Viccellio, and Klaus Mueller, *Senior Member*, *IEEE*

Abstract—The Five Ws is a popular concept for information gathering in journalistic reporting. It captures all aspects of a story or incidence: *who, when, what, where, and why*. We propose a framework composed of a suite of cooperating visual information displays to represent the Five Ws and demonstrate its use within a healthcare informatics application. Here, the *who* is the patient, the *where* is the patient's body, and the *when*, *what*, *why* is a reasoning chain which can be interactively sorted and brushed. The patient is represented as a radial sunburst visualization integrated with a stylized body map. This display captures all health conditions of the past and present to serve as a quick overview to the interrogating physician. The reasoning chain is represented as a multistage flow chart, composed of date, symptom, data, diagnosis, treatment, and outcome. Our system seeks to improve the usability of information captured in the electronic medical record (EMR) and we show via multiple examples that our framework can significantly lower the time and effort needed to access the medical patient information required to arrive at a diagnostic conclusion.

Index Terms—Visual knowledge representation, data fusion and integration, coordinated and multiple views, focus and context, health informatics, electronic medical record (EMR), electronic health record (EHR)

1 INTRODUCTION

A central task in information visualization is to find the appropriate visualization paradigm for both the data and the problem scenario at hand. Many such visual information mappings exist [13], but it is well understood that there is no one method that can encode all aspects of a given scenario, once sufficiently complex, and so the concept of multiple coordinated views has become an established paradigm [20]. Fluid interaction among these views via cross filtering [32] and brushing [6] is the key to successful information (and data) exploration. Providing overview and detail-on-demand [31] is equally important—salient information should become available on a whim when requested but just as quickly disappear when no longer relevant. The interface we propose adheres to these well-established eminent requirements.

To structure the information domain and provide a suitable visual mapping for each we utilize the *Five Ws* (*who*, *when*, *what*, *where*, and *why*) of journalistic reporting.

The Five Ws are the elements of information needed to get a full story. They are encountered in many domains, such as a police detective investigating a crime or a market analyst planning an effective marketing campaign. The order in which the information is gathered or interrogated can vary case by case—crucial is only that all Five Ws are ultimately addressed. We believe that this grounding also fosters the new effort of storytelling in information visualization [29]—it will ensure that all aspects are covered in this visual story.

Our work demonstrates the application of the Five Ws to health informatics (HI). We find that most current health care informatics systems, if not all, lack the basic concept of information visualization—overview and detail-on-demand—making it difficult to get a quick and effective assessment of a patient's state of health. Information is poorly organized and hard to obtain, and this has been blamed as the prime reason for the slower than expected adoption of the Electronic Medical Record (EMR) [40]. It applies to both acute clinical encounters in emergency room (ER) scenarios as well as to doctor-collaborative diagnosis and treatment plan development. Progress has been made in terms of temporal patient-centric event organization [2], [22] and other statistical dimensions, but these have rarely been linked and comprehensively organized. We propose to use the Five Ws as a means to establish a comprehensive multifaceted assessment of the patient and his (her) history. We then associate each such W with a dedicated, linked visual encoding that can represent and communicate it to the other Ws in effective ways.

In this paper, Section 2 summarizes related work. Section 3 elaborates on the requirements of an EMR system. Section 4 focuses on the analysis of the information components and relations. Section 5 deals with the identification of suitable visual encodings of these. Section 6 outlines how our system could be integrated into the hospital workflow. Sections 7 and 8 present specific case

- *Z. Zhang, B. Wang, and K. Mueller are with the Visual Analytics and Imaging Lab at the Computer Science Department, Stony Brook University, Stony Brook, NY 11794. E-mail: {zyzhang, wang12, mueller}@cs.sunysb.edu.*
- *F. Ahmed and I.V. Ramakrishnan are with the Applied Logic Lab at the Computer Science Department, Stony Brook University, Stony Brook, NY 11794. E-mail: {faiahmed, ram}@cs.sunysb.edu.*
- *R. Zhao is with the Computer Science Department, Stony Brook University, Stony Brook, NY 11794. E-mail: zhao@cs.sunysb.edu.*
- *A. Viccellio is with the Department of Emergency Medicine, Stony Brook University Hospital, Stony Brook, NY 11790. E-mail: Asa.Viccellio@stonybrook.edu.*

Manuscript received 11 July 2012; revised 4 Feb. 2013; accepted 27 May 2013; published online 3 June 2013.
Recommended for acceptance by S. Miksch.
For information on obtaining reprints of this article, please send e-mail to: tvcg@computer.org, and reference IEEECS Log Number TVCG-2012-07-0131.
Digital Object Identifier no. 10.1109/TVCG.2013.89.

1077-2626/13/$31.00 © 2013 IEEE Published by the IEEE Computer Society

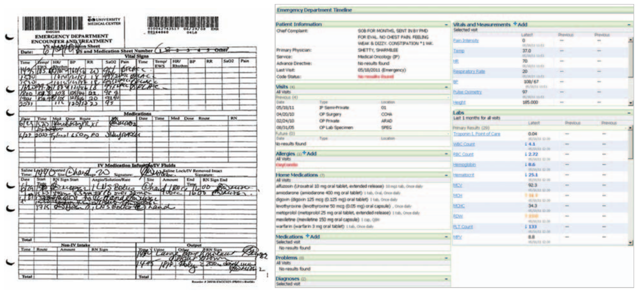
studies and an evaluation. Sections 9 and 10 offer a discussion, conclusions, and outlook on future work.

2 Related Work

A number of approaches for the visualization of medical patient records have been proposed and new systems are likely to emerge as the EMR—also referred as Electronic Health Record (EHR)—is adopted widely. A frequent paradigm is to organize the patient records along the time axis. Early work is that of Powsner and Tufte [23] who construct *graphical summaries*—ensembles of scatterplots visualizing different relevant medical variables such as glucose levels and temperature over time. These are then annotated by comments and expertise indicators of the commenter. More recent work in this direction includes the Critical Care Patient Data Visualization (CDDV) [10] and the VIE-VISU [15] systems, as well as LifeLines [22] in which health records are distinguished by their inherent aspects, such as problems, symptoms, tests/results, diagnosis, treatments and medications, and so on. Color is used to indicate severity or type, and a level of detail mechanism allows one to zoom into patient records. A number of other systems, such as VisuExplore [27] or the framework described by Kosara and Miksch [17] have also embraced this type of patient data visualization. Particularly interesting in this context is the Midgaard system [3], which makes effective use of illustrative abstractions to gradually transition between broad qualitative overviews of temporal data (for example, blood pressure) to detailed, quantitative time signals. These techniques are further elaborated on by Aigner et al. [2]. While most systems arrange the different time-varying medical variables into a set of horizontal tracks, the approach by Ordonez at al. [21] uses starplots in which spokes are mapped to variables. Each time interval then corresponds to one complete polyline wrap around the starplot, conveying temporal continuity. Line color is mapped to time of data acquisition, and so, by looking at the ensemble of wraps users can easily visualize dynamic patterns of change that may exist in the multivariate data.

Conversely, the focus of our work is not so much the visualization of time-varying phenomena in a patient's medical variables, for example, monitor a progressing disease or follow overlapping, concurrent intervals of different timelines for several coexisting problems or therapies, and so on. Our framework could easily adopt one of the existing techniques enumerated above and integrate it as a module dedicated for this purpose. Further, our focus is also not the interactive exploration and knowledge-based interpretation of large quantities of time-oriented clinical data (see, e.g., [30]). Rather, we aim to provide a system by which physicians can easily log and retrieve medical data and information of any kind, and see relations among these items as a whole. This is where we believe the Five W's scheme has merit.

Not all, but many medical data have a direct relation to human anatomy. For these information items, a template of a human body is an intuitive means to provide an index into the corresponding anatomy. Midgaard [3] provides such a utility. Likewise, Ropinski et al. [24] gather closeups of acquired radiological data around a volume-rendered full body. Another frequent paradigm are flowcharts, as used in clinical algorithm maps [12] and others [11], [26], where patient records are visualized as a logical execution sequence of plans. These methods typically operate without temporal alignments. Aigner and Miksch [1] integrate these visualizations into coordinated views. Similarly, we provide a body map to show the *where*.



(a) (b)

Fig. 1. Evolution of the medical record. (a) Paper based. (b) Electronic. Images were shrunk to hide the patient's information.

As an attempt to decrease cognitive load in clinical routine, Workman et al. [34] replace standard medical plots by elaborate glyphs—called *knowledge-enhanced graphical symbols (KEGS)*—that encode deviations from normal. They show that this can improve accuracy of interpretation. While we do not use glyphs, we also make ample use of simple graphical representations to encode medical information for fast cognition.

While some aspects of our framework were recently summarized in two short workshop papers [36], [37], many details, such as color design, temporal encoding, filtering, our rating facility, user feedback, case studies, evaluation, and performance were not discussed there.

3 EMR Systems and Medical Codes

As mentioned, the adoption of EMR systems has been much slower than expected. This becomes immediately evident when one compares a conventional paper-based medical record (Fig. 1a) with a typical commercial display of an electronic medical record (Fig. 1b)—the information organization is rather similar! There are still separate boxes with textual information, they are just now in form of tabbed windows which can be scrolled and clicked on to obtain more detail. Indeed, handwritten notes are now replaced by easier-to-read printed text and browsing through paper document folders is replaced by more convenient scrolling and mouse-selection operations. This clearly is an advance, also in terms of portability, but the opportunity to reformat the digital information into more effective displays is largely being missed, and this leads at least partially to the current frustration with EMRs. Furthermore, a significant problem is also that there are no provisions for scalability—an increase in data and information simply leads to more scrolling and more diverse and deeper selection hierarchies.

The existing problems with current EMR systems have been rigorously studied in [40] from a usability standpoint.

The study finds that the key principles such a system should obey are simplicity, naturalness, consistency, minimization of cognitive load, efficient interactions, forgiveness and feedback, effective use of language, effective information presentation, and preservation of context.

We believe effective and robust information organization and integration via well-established criteria is a key to achieve these requirements. A hierarchy is a convenient data structure for this purpose, and the standard codes commonly used for billing in hospitals offer such a robust and hierarchical information organization. These codes are International Classification of Diseases (ICD), Current Procedural Terminology (CPT), and National Drug Code (NDC). ICD describes the condition or disease being treated—the diagnosis. CPT describes medical services and procedures performed by doctors for a particular diagnosis. NDC codes the administered drugs. As an added benefit, by building our visualization framework on top of this ubiquitous medical code infrastructure we also facilitate a seamless integration into existing hospital systems which use these codes ubiquitously to index medical facets.

4 THE FIVE W'S SCHEME

We shall first discuss the conceptual information organization of our system, in terms of structuring the Five Ws.

4.1 The *Who* and *What*

The *who* and *what* information helps doctors to quickly assess the history and status of the patient. It describes the patient in terms of:

- *Symptoms and diagnosis* include the patient's symptoms, injuries, and any diagnosed diseases. All of this information can be encoded using the ICD code.
- *Procedures and treatments* include patient tests and examinations, treatments administered, and drugs prescribed. This type of information can be encoded using the CPT code or the ICD-procedure code and NDC code.
- *Data* include test and examination results, review of systems, vital signs, and social and family history. The codes for these are part of the procedure code and yield information on what the patient already has.
- *Temporal information*. A time stamp or interval
- *Severity*. A value characterizing deviation from normal.

Our system encodes this information in two ways—hierarchically organized by medical codes and sequentially in form of relations and causations.

4.2 The Where

The *where* information refers to the location of the *who* and *what* information within the confines of the patient's body. While not all information can be localized that way, for the information that can, we encode it in a body outline map onto which information items are linked to their appropriate body locations.

4.3 The When and Why

The *when* and *why* show a case under (doctor) collaborative diagnosis/treatment, or an entire life span. This is best conveyed by a sequential chain that emphasizes causal or temporal ordering. Such a chain stresses causal relationships and encourages causal reasoning done by the physician. It also aims to model the standard medical workflow:

1. observe symptoms and possibly browse history data,
2. prescribe and evaluate tests results,
3. form hypotheses and possibly acquire more data,
4. cast diagnoses, and
5. prescribe treatments.

These steps may all be executed within one patient visit or they may prolong over some period of time, but the overall workflow is always engaged. The fifth step may include a referral to another doctor, which then starts another workflow (back-linking to the previous).

The *why* is represented by relationships. Doctors have the option to create links between different medical entities, using their medical knowledge. For single chains, the system simply connects the event chain one by one.

5 ENCODING THE FIVE WS

Fig. 2 shows our system's user interface along with the two types of cooperating displays it offers. In the following, we first provide an overview and then discuss each display in detail. The displays are:

- A hierarchical radial (patient overview) display (Fig. 2a) with an integrated body outline primarily for the *who* and *where.* It allows doctors to quickly survey and focus on details of the patient's medical history in a fact-centered and anatomy-referenced fashion, presenting symptoms, diagnoses, procedures, treatments, and data along with a *time occurrence histogram* (the *when*).
- A sequential (diagnostic reasoning) display (Fig. 2b) primarily for the *when* and *why*. It enables doctors to see and augment the medical records in the context of the diagnostic workflow—visit, symptom, test/data, diagnosis, and treatment.

The *what* is part of both displays (in form of the various nodes) and is context-sensitive. The two displays are linked, such that operations on either one will be reflected in the other. Thus, one can quickly switch between the (possibly evolving) sequential diagnostic reasoning flow and radial overview displays. The radial display is also able to communicate causal relationships, but in the context of the entire history of the patient. Our user interface provides various facilities for filtering, sorting, selection, and searching, which are available for both displays.

5.1 Hierarchical Radial Display

There are in fact three radial displays, one for symptoms and diagnoses, one for procedures and treatments, and one for data Each uses the appropriate standard medical billing codes as an organizational model. For example, the "symptom and diagnosis" display is organized by ICD9 code, a very detailed and readily available medical hierarchy. We are currently adapting our framework to the new ICD10 code, which is an expansion of the ICD9 code.

We use a tree data structure to store the code hierarchy information. For each symptom or diagnosis the patient has,

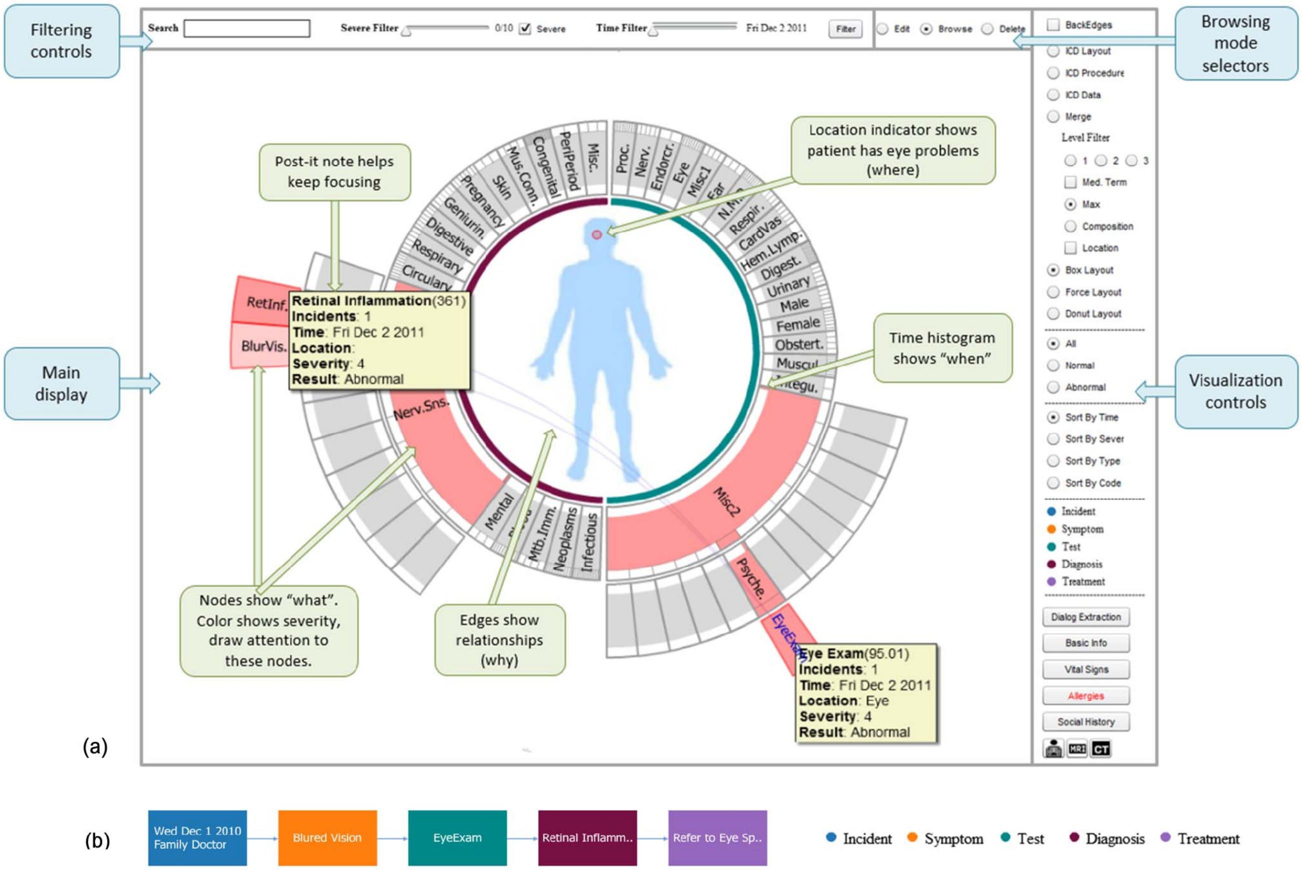


Fig. 2. The two coordinated displays of our system. (a) The radial (patient overview) display with integrated body map, along with the user interface. (b) The corresponding sequential (diagnostic reasoning) display using the same color coding. The user interface is identical to the radial display but removed here to save space.

we find the node n in the tree with the corresponding ICD9 code, and insert the new item as a child for node n. For example, if the patient has *bacterial meningitis* (ICD9 = 320), we first build an incident (medical facet) node m for this diagnosis to store its information (time, severity, result, and so on). In the tree, we find the node n with code 320, which is [320 *bacterial meningitis*]. Then, we insert m as a child of n. Next, we update all ancestors of n with the new inserted incident node's information, such as number of incidents that fall into this category, severity, and so on. This also updates the time history histograms. By doing this for all symptoms, diagnoses, and procedures, the tree will always be current and contain the patient's entire history.

5.1.1 Visual Design

There are many methods to visualize hierarchies [9]. We chose a space-filling paradigm because it can be better restricted to occupy a given space than overlapping visualizations, such as node-link diagrams. For space-filling visualizations, we had the choice between rectangular and radial displays. Treemaps [16] is a popular member of the former category. We ultimately chose a radial one—the sunburst [25]—because it allowed us to easily integrate a body map into the center and so make the map equally accessible to all nodes. This presented a clear justification for us to use a radial over a Cartesian layout which has been shown by in [8] to bear some advantages in terms of accuracy and ease of reading. On the other hand, we do follow the study's other guideline—to encode the more important dimension in sectors (as opposed to rings).

The sunburst is a radial hierarchical space-filling diagram. Nodes in the sunburst layout are drawn as solid areas (either wedges or bars), and their placement relative to adjacent nodes reveals the relationships in the hierarchy. Because the nodes are space-filling, the angle for each node can be used to encode additional information, such as number of incidents in our case. The sunburst layout has been widely used to visually encode hierarchical structures, such as documents [7] or software systems [18].

Show *Where*. Typically, the root of the tree is displayed in the sunburst center. However given the sole application context—the patient—we replace the standard root node by a human body template. This enables us to intuitively fuse the *who* with the *where* display. The two displays are interlinked, such that nodes in the sunburst point to the appropriate body locations (if such a mapping exists). If an incident (medical facet) has corresponding location information, a red dot is displayed in the body outline; otherwise, it is mapped to a dot outside the body outline (above the head). The intensity is used to encode the severity, which is computed using the same color composition method than for the nodes (see section on Color Design below). Thus, by looking at the body outline, doctors can quickly learn which parts of the patient's body have (or had) diseases and also judge their severity by the color intensity. Hovering on the red dots will popup more details

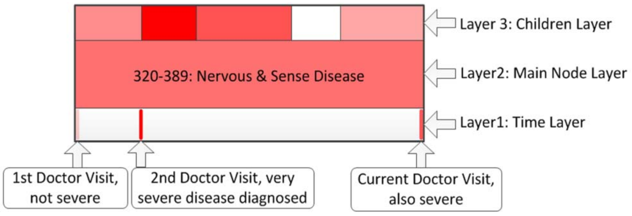


Fig. 3. Node design. Shade of red encodes severity. This node tells us that the patient has a relatively severe disease in the nervous and sensory organs. There have been a total of three doctor visits related to these diseases. The children layer gives more detail on the specific kind of disease within this broad category.

about the injured part, such as name, severity, and how many incidents are related. Clicking the red dot will highlight the corresponding diseases in the sunburst tree.

Node Design and Time Histogram. Each sunburst node has a wedged shape. We further decompose each node into three layers to encode more information, as is shown in Fig. 3. These layers are:

- *Layer 1 (Inner layer)* encodes time information. The length of the node represents the entire patient history, that is, going from left to right (clockwise in the radial layout) we encode the history from the first doctor visit up to now. This *time histogram* allows doctors to see how often the patient received a certain treatment or exhibited a certain symptom, or how long ago this occurred, and so on.
- *Layer 2 (Middle layer)*. The main node layer which is used to display information about the node, such as code, name, and so on.
- *Layer 3 (Outer layer)* encodes the next lower level in the hierarchy. It is meant to give users a quick overview on the subdiseases without showing their real nodes. We provide this layer to make the hierarchy display scalable.

Integrated Display. The three hierarchical radial displays, symptoms/diagnosis and procedures/data, and treatments can be combined to display the entire (medical) picture of the patient as well as the relationship between them. The corresponding subrings are colored accordingly. If two incidents are related with one another, an edge is drawn between them. With these relationships, users can select a node to see what other nodes are related to this node. This exploratory functionality can assist doctors in the medical reasoning process. To get all dependent nodes for one selected node, we use a graph traversal algorithm to compute the dependent closure.

A First Example. Fig. 2a combines the diagnosis view (left half with dark red ring) with the test/data view (right half with green ring)—the color legend is in the visualization controls panel on the right. In the example shown here, a patient visits the doctor complaining about blurred vision. This is the first visit of many and we will return to this case in Section 7.2. The body map in the center shows the anatomical location (here the eye) of the patient's medical problems, and the edges point to the corresponding nodes in the radial display (here from test "eye exam" to diagnosis "retinal inflammation"). Doctors can click on any node to obtain more detail and can then pin this information to the display as "post-it notes" (see Section 5.1.2). The time histogram in the inner node layer has only one bar marker on the far end since this is the first visit of the patient—else there would be more bar markers distributed over the layer ring.

Color Design. When refining our display with our collaborating doctors, we were repeatedly told that one feature they cared very much about was the ability to quickly assess the severity of a symptom or disease. Therefore, in all three layers, the shades of red encode severity information—full red encodes highest severity 10/10 and white encodes no severity 0/10. We used the 0-10 scale because it is often used in the social sciences and in medicine, for example, the Comparative Pain Scale [41].

We use the linear red color scale to shade severities in between 0 and 10. We may also encode severity on a diverging scale—severely low and severely high—color-coded using an appropriate diverging color scale [5]. We employ green-white-red to signal positive and negative outcomes, respectively.

If the node contains multiple incidents in its children, we use color composition to summarize this information. Our interface provides two color composition techniques:

- *MAX* takes the maximum value from all composited severities as the current node's severity and uses it to compute the color. This composition means that if there is one subdisease/subsymptom that has high severity, then its parent category should also be paid attention to.
- *Compositing* computes the color using composited rendering along time. This method fades the colors for past events. Since only one color (red) is used, the color composition can be solved by an alpha blending equation:

$$s = max\left\{\sum_i (s_i w_i); 1\right\}. \tag{1}$$

Here, for each incident i, S_i is severity and w_i is the corresponding weight, which is a function of time. Early incidents have lower weights and more recent incidents have higher weights.

Fig. 4 shows via an example that each technique has its own advantages and disadvantages. The MAX operator can draw a doctor's attention to the most severe diseases, no matter if they occurred a long time ago or just now. This can be good for some long-term severe diseases, such as diabetes. But it may cause misunderstandings for gradually recovering diseases, such as bone fractures. Conversely, the color composition technique takes time into account, it fades diseases that occurred a long time ago and highlights the most recent ones. The two modes are complementary to each other—we observed that our collaborating doctors switched back and forth between the two modes when exploring a medical history.

Edge Coupling. Edges in the integrated view are displayed according to which level of the hierarchy is chosen. Consider Fig. 5 where we show a pair of related nodes in an integrated display and three code levels. The original edges *e1* and *e2* link the nodes corresponding to the specific incidents (the leaf nodes) of level 3 (Fig. 5a). As the user

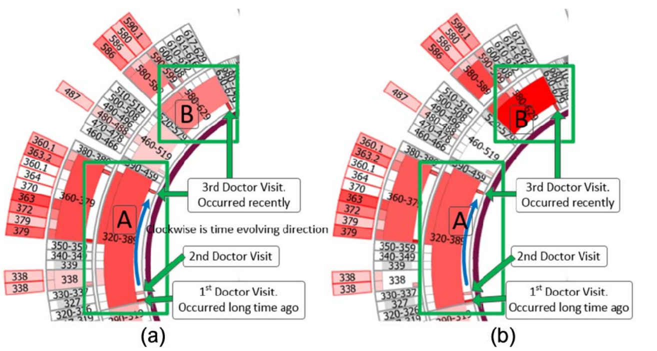


Fig. 4. Color Composition. Node A contains three doctor visits; Node B contains only one but very recent doctor visit. (a) Using the MAX operator. (b) Using color composition. The coloring of (a) suggests that node A has the highest priority. But A's color is determined by a severe disease which, however, occurred in the patient's first doctor visit (a long time ago). That disease might not be as important compared to some less severe diseases that occurred in the patient's last visit (just a few days ago). So, if more recent occurrences are the focus, this disease will be better highlighted using color composition (shown in (b)) which takes time into account. This makes node B more apparent.

collapses, these incident nodes, the edge will link to their parent node (Fig. 5b). If we collapse all of the incident nodes, then *e1* and *e2* will be merged together (Fig. 5c), and the corresponding intensity of the edge increases. Edge bundling [14] is used to reduce cluttering.

5.1.2 Interaction Design and Scalability

Each radial display is either hierarchy-centric or patient-centric. In the hierarchy-centric display (Fig. 6a), each node in the sunburst tree is sized by how many subcategories it has. It focuses more on the hierarchy information represented in the medical codes and serves as an illustration of the complexity of a subsystem and its composition. Conversely, in the patient-centric display, more radial space is dedicated for diagnoses/procedures the patient had activities in. For categories that the patient does not have any activities in, the node will be collapsed to save space for others (see Fig. 6b). This display is most often used as it makes better use of the space.

Multilevel Interaction. The sunburst radial display in Fig. 6b shows three levels of the code hierarchy. Level 1 corresponds to the highest code hierarchy level. Level 2 shows more detailed categories. Level 3 contains the incident nodes, which are the actual medical facets (symptoms/procedures/diagnosis) that the patient has activities in. The user is given the choice to either display the medical code or the corresponding term in each node (see Section 5.1.3). The first level always shows codes/names to provide an overview. Likewise, the leaf nodes also always show the codes/names for detail. The middle levels (second level) show codes/names only when they have incidents (children).

Three default level filters are provided to help users quickly explore these three levels. Users can expand and collapse the nodes interactively. In ICD9, certain conditions can have up to five levels and ICD10 has even more. Our sunburst display is scalable to support these additional levels by extending the underlying data model.

Augmenting the Display with Post-It Notes. Hovering over any node will reveal more information on a yellow *post-it note*. Clicking on the post-it note will pin it to the display for fast recall (Fig. 2a). Using them within a *diagnostic sandbox*, doctors can focus on the post-it noted symptoms/exams/diagnosis. The post-it notes can include data as well, such as an image or a time-plot. We have two versions for each: a thumbnail for pinning and the original size for exploring.

Filtering, Transformations, and Zooming. A time and severity filter is provided to filter out unrelated or unimportant incidents. For example, doctors can specify a time range and a severity threshold. Then, only the incidents that occurred during the specified time range with severity values higher than the specified threshold are shown.

Further supported user interactions help users explore the patient information and see details on demand:

- *Translation, zoom, and rotation*. These interactions manipulate the radial view to put the important features in the center of the window. We found that this helps users to stay focused and promotes ease of reading.
- In the integrated view, users can zoom into one specific hierarchy display by making its angular range larger. This shrinks the angular ranges of the other categories and all nodes are resized accordingly. Like the expand/collapse feature, this interaction helps users to focus on a specific disease of interest.

One concern about these interactions is that they might destroy a doctor's mental map. However, our user study (see Section 8) indicates that most doctors are comfortable with this feature. Also, users can always choose not to use these features if they dislike them.

5.1.3 Node Labeling: Medical Code and Terms

The display nodes can be alternatively labeled by the ICD code number or by the corresponding text (as shown in

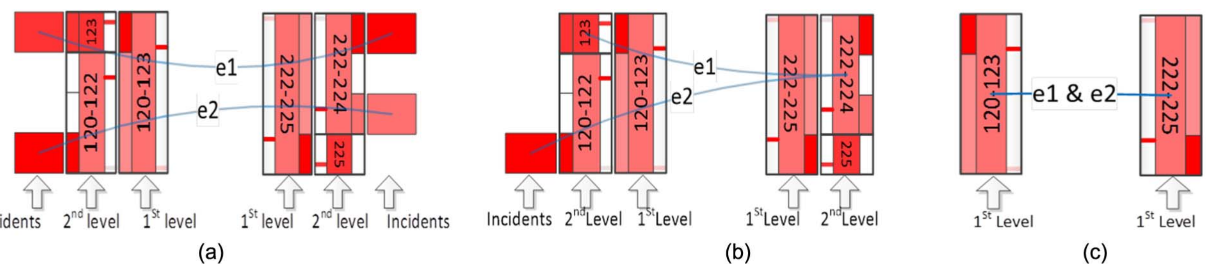
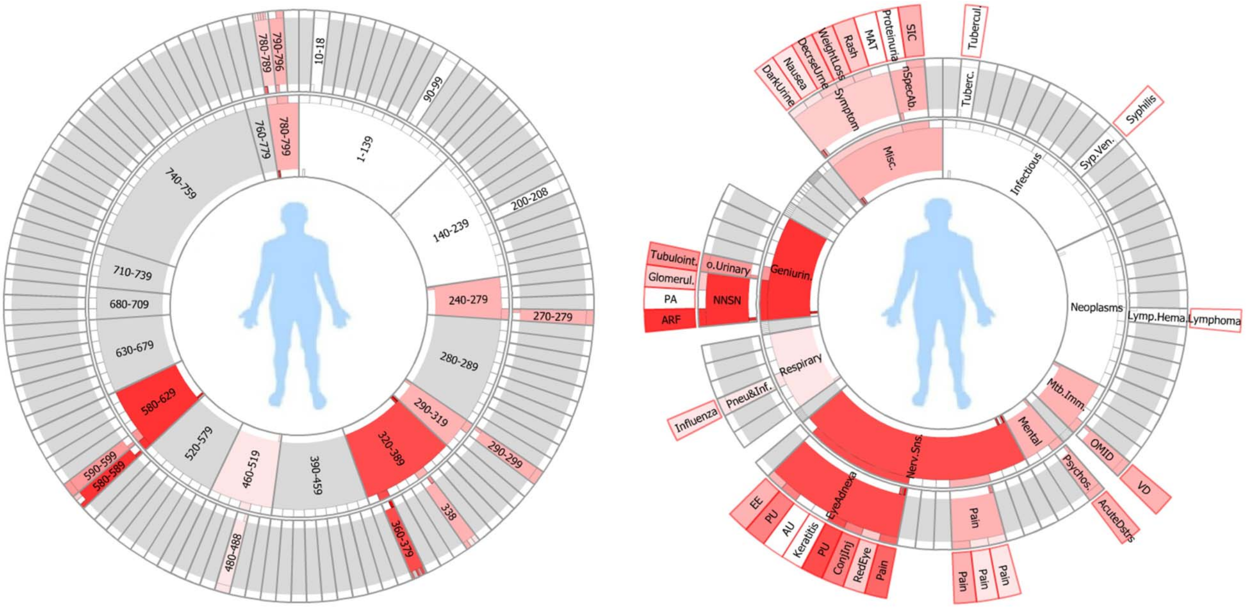


Fig. 5. Node collapse strategy. (a) Original edges e1 and e2 link the incident nodes; (b) Node [222-224] and [123] collapse. Edges linked to their children now link to themselves. (c) Only the first level nodes are shown, then e1 and e2 merge into one single edge. The opacity of the edge is computed by compositing those of e1 and e2.



(a) (b)

Fig. 6. Different layouts for the sunburst display with two alternative node labeling schemes. (a) The hierarchy-centric layout. The nodes are labeled with the ICD9 codes but labeling with the corresponding medical terms is also possible. (b) The patient-centric layout. Here, the nodes are labeled with the corresponding (abbreviated) medical terms as available in the ICD9 description.

Fig. 6). The interface provides a button by which a user can quickly switch from one representation to the other.

A challenge is to find good abbreviations for the nodes' labels. The ICD9 medical terms can have very long strings. For example, code "049.9" stands for "Unspecified non-arthropod-borne viral diseases of central nervous system." To make the display visually manageable, we aim for a string length of 10 characters or less. This ensures that no label extends much beyond its node's border. We found that while some standard medical abbreviations exist, for example, *CNS* for "central nervous system," abbreviations for others are less obvious. Our current approach uses standard techniques to shorten the strings. First, we remove all stop words such as "the" and "of." Then, depending on the length of the remaining words, we employ the following increasingly aggressive techniques:

- *Contract* omit some or all interior portions of the word but retain its first and possibly last letters or elements.
- *Abbreviate* cut the word after some characters and then terminate the remainder by a period.
- *Acronym* only retain the first character of each word, turn them upper case and concatenate without blanks.

These strategies have helped to shorten all strings to the desired length of 10 or less. Yet, the result is not always satisfactory. We have, therefore, devised an interface that doctors can use ad hoc to define a better abbreviation for a term that they think is not well represented.

5.2 Sequential (Diagnostic Reasoning) Display

The sequential display is used mainly to demonstrate *what*, *when*, and *why* information. Usually, the diagnostic workflow is: Patient visits doctor → patient complains of symptoms → doctor orders tests → doctor renders one or more diagnoses (valid or not) → treatments are given → outcome is observed. Thus, a sequential display can show this reasoning chain very well.

The medical records are organized by an underlying graph data structure. Each node corresponds to one stage of the workflow—visit, symptom, test/data, diagnosis, and treatment. Edges represent relationships. Each stage corresponds to a column of equal-colored nodes. This achieves a well-structured design that reduces the overhead for visual search and, therefore, lowers cognitive load. An earlier version of this display [36] used a force-directed layout in which the various types of nodes could appear anywhere on the canvas and were only distinguished by color. We abandoned this display since our medical collaborators found it difficult to work with.

5.2.1 Visual Design

A node is displayed as one elongated box—we found that this better utilizes the rectangular screen, better fits the text, and has better scalability compared to a circular shape. All of our medical collaborators agreed on this. If two nodes are related with one another, an edge is drawn to link them together. As an example, a very simple sequential display is shown in Fig. 2b. It incorporates the test and diagnosis nodes of the radial display of Fig. 2a, and the other elements of the visit.

More complex chains can have many edges. We use edge bundling to reduce the cluttering that may occur. Further, in some cases the current doctor refers the patient to see another specialist (which is the treatment in this case), or current symptoms are caused by previous described drugs (which can be a form of diagnosis). In these situations, back-edges appear. Back-edges are shown in different color (red) to make them easy to see. Fig. 7 shows an example for a back-edge drawn to indicate a side effect (see caption). We find that the equivalent radial display shows the back-edges

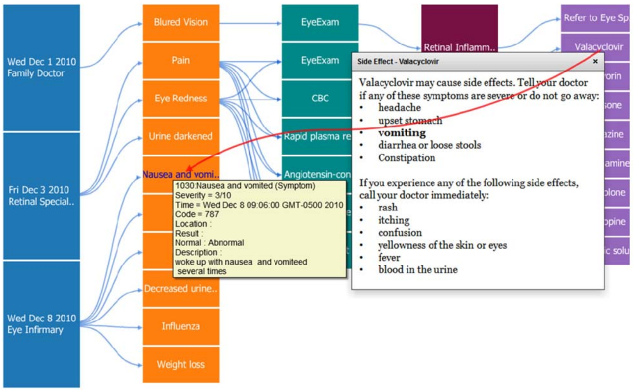


Fig. 7. Some features of the sequential display: information text window, post-it note, and red back-edge. A back-edge from a treatment to an incident can denote a referral, while a back-edge from a treatment to a symptom can denote a side effect as is the case here—the drug *Valacyclovir* prescribed at a previous visit (December 3) causes the current visit's (December 8) symptoms of nausea and vomiting.

Fig. 8. Severity and uncertainty rating bar variations for sequential display nodes. The location of the vertical black line below the node indicates whether the data item is a two-sided rating (black line in the middle) or a one-sided rating (black line on the left). The length of the bar encodes the value and its saturation encodes the uncertainty of that value. The text inside the boxes above explains the semantics that our system uses in more detail.

especially well because they go against the flow of (cross) the other edges. Our system currently does not have a specific column or radial display for treatment outcome, as a gauge of effectiveness. Rather, outcome is logged and can be monitored by examining the corresponding treatment node, which is back-linked to the other nodes in the diagnostic workflow.

Rating. All incidents can be rated by the physician on the fly using a popup with a slider. To encode the rating, one option is to use the same method as was used in the radial display—color. This would make for a consistent encoding. However, there is a conceptual difference in how the two displays are used. The radial display is meant to provide an overview where color (in particular shades of red) can quickly guide the viewer to the more severe nodes. Conversely, the sequential display is for diagnostic reasoning where quantitative assessments are to be made. According to Bertin's levels of organization [4], color and brightness are selective and ordered but only size is quantitative. With this in mind, we use different types of visual severity encodings for the two displays: color (saturation) in the radial display and length (of a rating bar) in the sequential one. This rating bar is positioned below the corresponding display primitive and the semantics of the ratings determines its color and variation. Fig. 8 illustrates the various schemes that we will further explain below. The rating uses standardized levels to gain independence from scaling issues and so provide for a scale-neutral node display. Our medical experts indicated that they are able to translate these ratings into actual values using their medical knowledge. But, hovering over the node will display the actual values.

Symptoms, Tests, and Diagnoses are rated in terms of severity, that is, the deviation from normal according to some scale. As mentioned, we adopt the Comparative Pain Scale [41] of 0-10. Severity is encoded in a *severity bar*, which is gray with full length at first, meaning the node has not been processed by the doctor. After the doctor looks at the node and sets its severity or normality, the severity bar will have the same color as the node, which our doctors agreed to be the most aesthetic. The bar's length is weighted by the severity level. Our system supports two types of severity:

- *Two-sided (often used for data)*. The normal value is in the center and deviations are either too low or too high.
- *One-sided (rates severity of symptoms and diagnoses)*. The normal value is on the left and the most severe value is encoded as a bar with full length.

Treatments are rated by their outcome—whether the treatment has a positive (successful) effect on the patient or a negative effect (unsuccessful, causing side effects). Green color encodes positive effects, and red encodes negative effects. The length of the rating bar means how successful/poor the positive/negative effect is.

Uncertainty. The uncertainty or confidence in a diagnosis or test is encoded as the saturation of the corresponding node—full saturation means no uncertainty (full confidence) and low saturation means high uncertainty (no confidence). For example, if the doctor thinks that the patient may have a 50 percent possibility of suffering from lung cancer, then the saturation of the node is reduced by 50 percent. This rating is also given by the doctors interactively.

Interaction Design and Scalability. Hovering over a node brings up scrollable text windows that provide relevant information, such as data details, side effects of a treatment, narrative text from a patient report, and so on. Also, similar to the radial display, "post-it notes" with a more detailed description can be pinned at any location, possibly reduced to just show a few pieces of information, such as the rating and the full name of a disease (which appears as an abbreviated label in the node). Finally, any data associated with a node can be brought up by clicking on the corresponding data icon in the box.

The nodes may be sorted by any data field: time (date), doctor, severity, and so on. Scalability is achieved by 1) muting unselected nodes and their links and possibly completely collapsing them, 2) aggregating related nodes into a single box where these groupings can be defined by similarity in the sorting variable, such as data, temporal, doctor, and the like (one typical grouping might be a visit with a specific doctor), and 3) filtering with a global slider by severity, time interval, and so on. Fig. 9 presents an example.

We use edge bundling to reduce clutter. Since back-edges are drawn on top of other nodes/edges, too many of these tend to clutter the display. Hence, the back-edges are not shown by default. Users can turn on those edges to see the transitions. Also, in browsing highlight mode (Fig. 9ii), users can hover on any node and then only the nodes that are related to the selected one and their corresponding

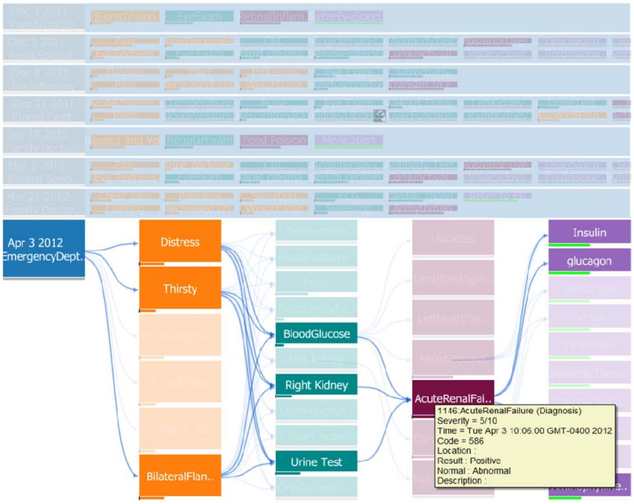


Fig. 9. Sequential (diagnostic reasoning) display. (i) Node-collapse to focus on the most recent visits—a total of more than 100 incidences is shown. (ii) In the browsing highlight mode, the doctor has selected one of the diagnoses which highlights all related nodes and branches but fades out the others. The same type of highlighting also occurs when filtering.

edges will be highlighted. The same type of highlighting also occurs when filtering is applied. The effect is real time and it allows users to quickly browse complicated displays by moving over nodes and branches, which then highlights the associated elements and muting others. Our video demonstrates this function in action.

5.3 Implementation Details

The user interface is implemented using Action Script and the Flare visualization toolkit [39]. The back-end server is implemented in Java and Java EE. It connects the front-end interface with the structured database that stores the patient histories. Each patient record contains: visit ID, type, code, description, time, and so on, and if available, severity, uncertainty, body location, and IDs for related reports or diagrams. The relationships are built by the physicians during the exam and are stored in a separate table. The communication between the interface and the server is achieved with the help of BlazeDS.

6 Integration into Hospital Workflow

We see at least two places in a hospital workflow in which our system can proof useful: 1) as a diagnostic assistant in the patient-doctor encounter to help doctors learn about the patient history and support clinical decision making, and 2) providing medical coding support in the hospital's coding and billing departments.

6.1 Diagnostic Assistant

When doctors perform a diagnosis, it is essential to have a good understanding of the patient's history. Further, one doctor often takes charge of several patients, particularly in busy emergency room scenarios. Therefore, the smaller time that is spent on learning a patient's history, the more efficient actions will be taken. And as a result, the more patients will be taken care of. Our system allows doctors to quickly get insight into important issues like:

- What were the most severe symptoms this patient had, now, recently, and in the past?
- What tests have been done related to these symptoms, and what were the results? Were there treatments that the patient did not respond well to, or not at all?
- What were the diagnoses rendered in these tests? What were the outcomes? What were the reasoning chains that led to these diagnoses? Were there ruled-out diagnoses?
- What medications were prescribed in the past and when? What side effects do they possibly have, and might they have something to do with the present symptoms?

With respect to the last issue, because our system is connected to online databases, a doctor can quickly research information on drugs and their side effects and other information on possible treatments and causes.

6.2 Medical Coding Support

Medical coding is the transformation of report-based narrative descriptions of diseases, injuries, and medical procedures into numeric or alphanumeric designations—the ICD, CPT, and NCD codes—that are used to bill patients and insurance. Hospitals typically have certified staff for this task: medical coders. Medical coding is not without challenges, and we shall list the most relevant next:

- *Poor doctor's handwriting*. This challenge is trivially overcome by computer-based input.
- *Mismatched terminology*. Cases can exist when doctors use a different terminology than the one formulated in the ICD or CPT codes. So coders looking at an operative note might expect a certain descriptive word for a procedure and if they would not find it, they will code that the procedure was not done. However, the doctor might have described the procedure in different terminology, and so the procedure would go unbilled.
- *Unbundling*. This describes the fraudulent process of breaking apart (fragmenting) codes that are inclusive of other codes. An example is coding two units of CPT 67,311 (strabismus surgery, recession or resection procedure, one horizontal muscle) instead of one unit of CPT 67,312 (strabismus surgery, recession or resection procedure, two horizontal muscles).
- *Upcoding/undercoding*. The former is the fraudulent practice, in which provider services charge for higher CPT procedure codes than those actually performed, resulting in a higher payment. Since the rules are fairly complex, just to be safe a doctor may deliberately bill for work on lower level codes even if more services were performed, leading to loss in revenue.
- *Not coding the diagnosis to the highest level*. Some ICD codes need a fourth or fifth digit to be accurate and correct, many coders tend to use the highest level to save time.
- *Incomplete reporting*. Physicians may not report on everything they did although they may have performed it.

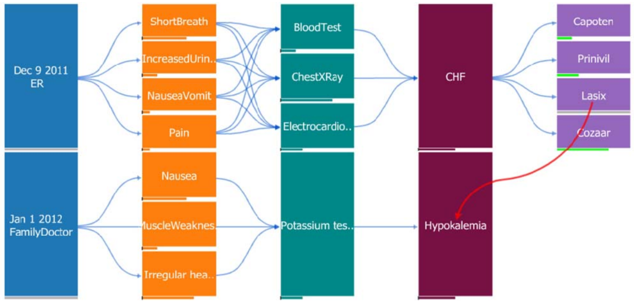


Fig. 10. Diagnosing a case of Hypokalemia with the sequential display. The cause is a recently prescribed medication: *Lasix*.

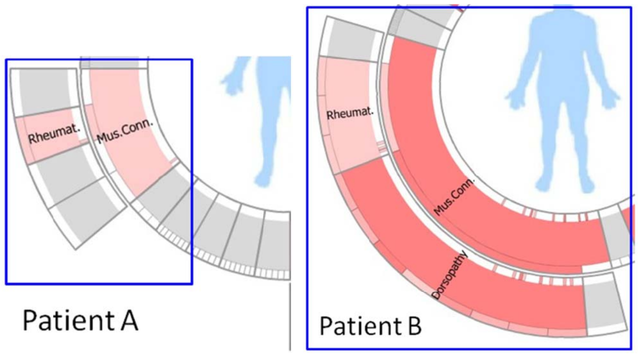


Fig. 11. Radial displays for two patients reporting to the ER with back pain. The relevant area is the lower left quadrant labeled Mus. Conn. From the time histograms, we see that patient A had no incidence of back pain before, while patient B is a frequent sufferer.

Since users can easily switch between medical terms and ICD codes for the nodes, our system can be used by both doctors and coders—the underlying (medical code based) hierarchy is identical. Doctors tend to be less familiar with the actual ICD codes and so they typically use the medical term labeling almost exclusively. Coders on the other hand make use of both medical terms and medical codes. By using the same display infrastructure for both reporting and billing the possibility of the problems due to mismatched terminology is greatly reduced. Further, our system also provides medical coders with a much better overview about the services performed and what services may have been performed and so helps avoid revenue loss due to incomplete reporting. Coders can quickly and better see relationships of treatments and procedures and so avoid upcoding, undercoding, unbundling, and other reporting errors that often lead to lengthy and costly struggles with insurers.

7 Usage Scenarios

We have explored a few usage scenarios to demonstrate the effectiveness and efficiency of our system. One scenario is reconstructed from a complex medical case involving a number of physicians. The others are based on routine medical cases that occurred at our home institution.

7.1 Scenes from Daily Clinical Practice

For this part of the study, we identified four sample scenarios from daily emergency room practice. They were compiled by our collaborating ER physician who has 25 years of professional experience. This ER doctor has long been looking for an interface where "information is right there when I need it" and came to us highly frustrated with the current state of the art.

We compare our prototype with a state-of-the-art commercial EMR system. Here, we are mainly interested in gauging how efficient each can provide insight into a patient's medical situation. As a quantitative measure for this capability, we count the number of mouse clicks needed to uncover a specific piece of information [40]. We analyze the four scenarios using both our prototype and the commercial system from the hospital (with similar screens in Fig. 1b), accessing a (patient deidentified) copy of our university hospital's database. For the latter two scenarios, we only briefly summarize the possible interaction without figure references. In the following, we shall motivate each scenario by a specific clinical task.

7.1.1 Diagnostic Medical Reasoning

We choose a patient who has been admitted to the ER with serious nausea and irregular heartbeat. A test result points to a low potassium level. A deeper look at the patient's recent history reveals that he was diagnosed of congestive heart failure (CHF) in the past, and prescribed *Lasix* as a preventive medication. The doctor knows that low potassium level can be a possible side effect of *Lasix*, and so the current diagnosis could be related to it. An alternative medication is prescribed. To obtain this information from the hospital's health IT system, the required number of mouse clicks n_c is (at least) 9**:**

n_c = [go to patient details] + [problems & diagnoses] + [medication list] + [first med.] + [details first med.] + [second med.] + [details second med.] + [third med.] + [details third med.] + click X different rows = 9 + X.

Each mouse-click changes the screen, breaking the mental flow. Conversely, in our system the doctor has the choice of using either the radial or the sequential display. Fig. 10 shows the latter, with just the most recent patient visits summarized. With one glance, the doctor learns about the patient's CHF history and the four medications that were prescribed. Seeing *Lasix* as one of the medications on the list and connecting this fact with the current finding of hypokalemia (low potassium), the doctor quickly gets the answer. A red back-edge is drawn to indicate this causal relationship, and an alternative medication is prescribed (not shown). Hence, with our system the number of mouse clicks is just 1 (to select/filter the recent events from the sequential display). No screen ever changes—the doctor maintains full overview at all times.

7.1.2 Detection of Substance Abuse

Back pain is frequently reported in the ER, and narcotics are often prescribed without much examination of patient history since it takes too much time with current EMR systems. However, at many occasions, patients either simulate their back pain to obtain narcotics for street sale or own personal abuse, or they have fallen victim to chronic pain which should be treated via alternative ways.

Fig. 11 shows the radial displays for two patients, A and B, who both complain of severe back pain and request narcotics to relieve this pain. Back pain falls into the large category of *musculoskeletal and connective* (labeled *Mus. Conn.*). By looking at the time histograms, the doctor quickly sees that patient A did not have any back pain before, while patient B has had regular hospital visits for chronic back pain. The appropriate courses of action are now taken. Obtaining this insight with our interface took a simple glance at the radial display. For the commercial system, on the other hand, the same information required $\mathrm{n_c} = 6$ mouse clicks.

7.1.3 Assembling and Connecting Information

Another problem with current systems is that related information is difficult to connect and assemble. For example, patients frequently carry dozens of medications, prescribed by different physicians. Yet current systems make it difficult to track what a given medication was actually prescribed for. This can have dangerous consequences when a medication has numerous uses (take, for example, Inderal, which treats hypertension, migraine, hyperthyroidism, and angina). These uncertainties often lead to overprescriptions of medications and unexplained adverse effects. By combining the treatment section with the diagnosis section in our radial display, this type of information can be easily obtained in one glance by following the arcs that link two nodes on opposite sides. Alternatively, one could also use the sequential display for this task as well. On the other hand, for the current systems, using suitable examples from the database, we find that $\mathrm{n_c} = 6 + \mathrm{X}$ mouse clicks are required to extract this insight.

7.1.4 Reconstructing Patient History

The following is a more practical case. The ER saw two patients, A and B, both diagnosed with too low heart rates. Both A and B were on lopressors to treat high blood pressure. A routine intervention would have been to give both a pacemaker, but an extensive click-through session with the EMR system finally revealed that A was recently put on a double dose, while B had received the same dose of medication for five years. The action was thus to reduce A's dose and only give B a pacemaker. The conventional system required $\mathrm{n_c} = 11$ mouse clicks for this, conversely, our displays are specifically designed to hold such time histories and, therefore, can reveal them in one view.

7.2 Collaborative Analysis

A significantly more complex scenario is derived from a case recently reported in the New England Journal of Medicine [35]. From it, we constructed simulated visits for the patient (a 22-year old woman) and the four different doctors involved. We then updated the visual displays by the medical information accordingly. In the following, we present these displays and some of the interactions that might have occurred if the system had been available during these visits. Fig. 12 presents the sequential display that lists the four visits top to bottom, and the radial displays for visit 1, 2, 4, respectively.

Visit 1: Onset—Primary Care Physician. The story begins with the patient visiting her primary care physician, reporting blurred vision in her right eye. The doctor suspects that a retinal inflammation might be the cause and refers the patient to a retina specialist as the immediate treatment. He then annotates this information in the interface, shown in the top row of the sequential display—for clarity we do not draw the back-edge of the referral. The radial display annotated with "visit 1" is shown below. The physician marked the diagnosis of retinal inflammation as fairly serious, using the rating popup. The body map has a moderate circle in the eye region, noting the problems there.

Visit 2: Retina Specialist. The day after next, the patient goes to visit the referred retina specialist. She now has severe throbbing pain behind the right eye and also redness. The eye exam reveals conjunctival injection and posterior uveitis. The urine appears to be darkened. CBC and other lab tests, however, turn out to be normal—the displays are updated as the results arrive a few days later. Based on the eye exam, the doctor prescribes a number of medications. The radial display is updated accordingly. The body map now upgrades the eye marking to a full red circle—highly severe—and it also adds a moderate red circle in the bladder region to indicate the slightly unusual urine color.

Visit 3: Eye and Ear Infirmary. Five days later the situation worsens—the patient feels sick again. She checks into the Eye and Ear Infirmary, complaining of her problems—vomitting and nausea which later resolved but was followed by pain in flank and groin. She also mentions her decreased urine outputs and weight loss. From the visual displays constructed so far, the doctor quickly learns that the lab tests were normal. Now, for each symptom the doctor needs to find some explanation or devise further tests. This reasoning activity is well supported by our interface. Fig. 7 demonstrates the process by ways of the symptom "*Nausea and Vomit*," using the diagnostic chain interface. By looking at the prior chains, he notices that the patient's previous treatments contained *Valacyclovir*. He suspects from prior experience that this medication may have something to do with it. To confirm, he conveniently pops up the medication information and sees that it has indeed the side effect "*Vomit.*" So this may explain why the patient has nausea and vomited, and the doctor draws a red-colored back-edge to link the two. He also sends the patient for a vitrectomy. The displays are updated accordingly and the body map now also shows additional problems the patient is reporting, such as a flu.

Visit 4: Conclusion—Emergency Department. The case escalates to its peak on December 21, 2011 when the patient reports to the emergency department. Additional diagnoses point to problems with the kidneys—the renal system. The ER doctor assigned to the case takes a renal ultrasound and commits it to the EMR system. Then, as it is often the case in hectic ER environments, he gets called away from the patient to take care of another. A new doctor—a renal system expert—gets assigned and she inspects the displays aggregated so far. She knows that these types of kidney problems can stem from either glomerular, tubulointerstitial, or vascular causes. Looking at the patient history, she quickly notices the blurred vision reported recently. This constitutes important evidence in the differential diagnosis that now has to commence—blurred vision is often a symptom in glomerular causes of renal failure. A rapid plasma reagin test is administered but turns out negative, which rules out glomerular causes. Looking back at the sequential display the ER doctor notices that sulfadiazine

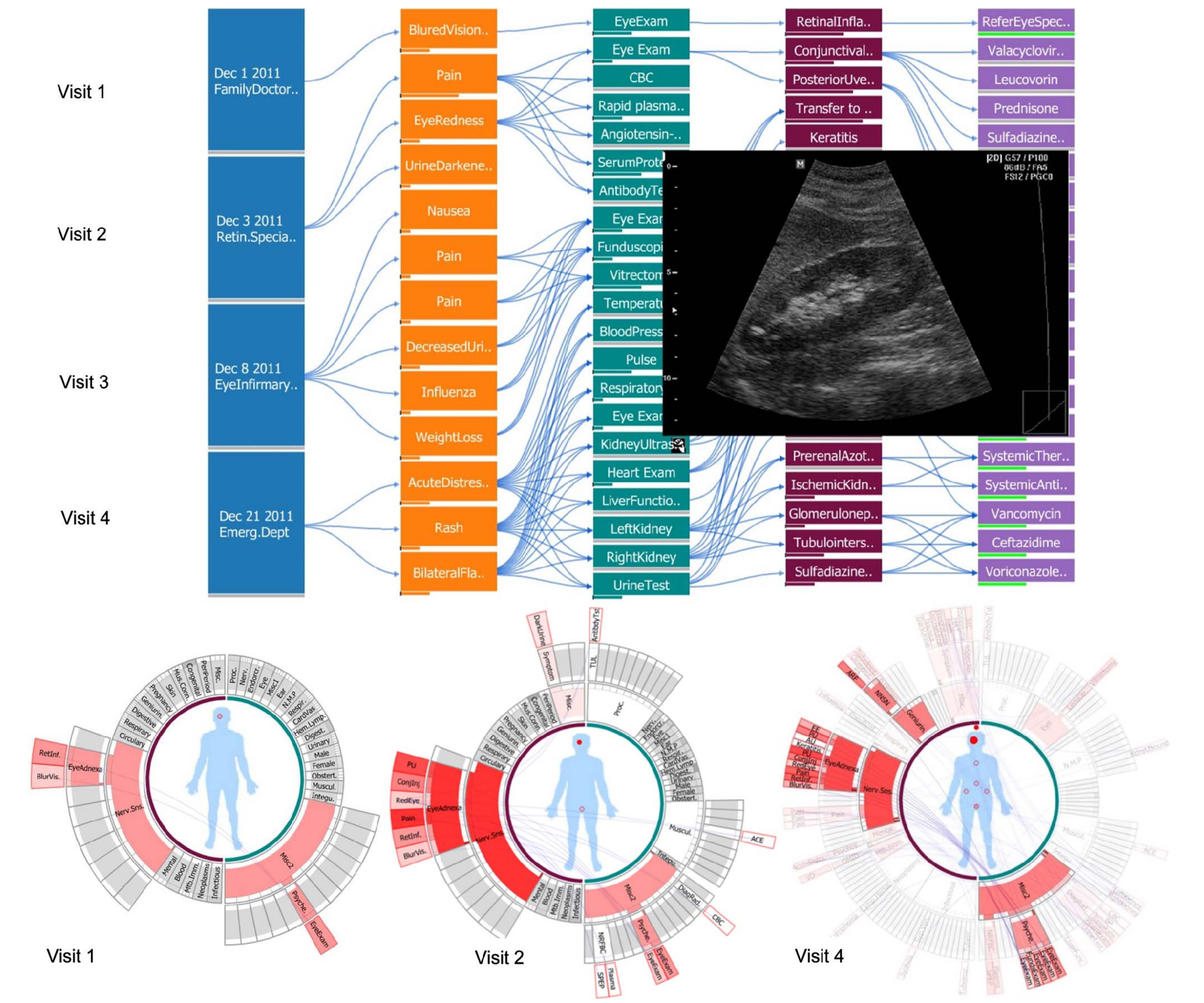


Fig. 12. Complex medical case involving four different doctors in a collaborative diagnosis task. Top: sequential display after the fourth visit. Bottom: the emerging radial displays labeled by visit number.

was prescribed to treat the conjunctival injection—done by the retina specialist on December 3rd (visit 2). This sparks her attention, especially when she sees the flank pain reported on visit 3. She checks if the flank pain could be caused by an obstruction in the kidneys. She uses the sequential display to call up the ultrasound taken by her colleague before (Fig. 12). It appears normal which rules out plain obstruction as a possible reason. But she knows that the flank pain in combination with sulfadiazine can be a clear explanation for the kidney trouble that the patient is reporting. So she immediately stops administration of sulfadiazine and gives hydration instead. Following this treatment, the patient soon returns to normal. The eye also recovers as a result of the vitrectomy taken at visit 3.

8 Evaluation

To evaluate the usability and efficacy of our prototype, we interfaced it with an EMR database at a large teaching hospital. We invited six physicians (some were residents) and two health informatics professionals to participate in a pilot user study. None had previous experience with our system. All physicians were familiar with their current EMR systems, and the two HI professionals had much experience in designing and developing HI systems.

We first gave each participant a 6-minute tutorial about our system. We explained the idea behind each layout and the basic functionalities, including the search and filter facilities, the three different interaction modes, the body map, and so on. We prepared two sets of questions. The first set aimed at finding out whether our system can help physicians to quickly and accurately find information. The second set was more focused on design details along with some general questions. All six physicians did both sets of questions while the HI professionals were only given the second set. Our study was conducted with a set of real patients from the hospital EMR database.

8.1 Questions

The first set contained three questions, which were designed to test the efficacy of our system in terms of understanding the patient history and obtaining diagnostic assistance. All three questions had fully defined answers so we could test accuracy. We also recorded the time to find the correct answer.

Q1. What were the most severe diseases of the patient?
Q2. What were the anatomical locations of these diseases?
Q3. What were the symptoms and which are the related tests that had been prescribed?

The second set was designed to test the usability.

Q4. Did you find it hard to read text in the radial layout?
Q5. Was adding links manually in the box-layout helpful?
Q6. Did the collapse, expand, zoom, and rotate interactions in the radial layout affect your mental map?
Q7. Did the system save you time compared to standard systems (could be paper-based or EMR systems)?
Q8. On a scale of 1-10, how would you rate the system?

8.2 Results

For question set 1, all six physicians correctly identified the most severe disease, the anatomical location of this disease, and the related symptoms and tests. Thus, the accuracy was 100 percent.

The time for answering Q1 was between 4-10 s (mean 6.5 s), for Q2 it was between 2-11 s (mean 5.6 s) and for Q3 it was between 3-9 s (mean 5.7 s). For Q2, one physician knew the disease, we did not count his rapid answer (<1 s). Another physician took more than 10 s, stating that since the system was new it took some time to "learn where everything was." But eventually all physicians felt very comfortable with the system.

We found if physicians had the choice between sequential or radial layout, they would prefer the former. For example, for Q1, four doctors tried to find the answer using the sequential layout while two used the radial layout. Five used the severity filter to highlight the most severe diseases while one simply turned on the severity bar and found the longest one.

All physicians except the one who is very familiar with the disease used the body map in the radial layout to answer Q2. Five physicians used the *browse* mode (Fig. 9) to highlight the related symtoms and tests and finished the task quickly. One did not use the *browse* mode and tried to search through all links. This physician spent more than twice the time to identify the relation.

From question set 1, we learned that while all physicians felt that this system was vastly different from what they were used to, they became comfortable with it quickly and could efficiently locate the information we asked them to find.

Question set 2, on the other hand, was not specific to a certain medical case. It was designed to get directions of further system improvements.

For Q4, we were interested in finding out whether the nonhorizontal text in the radial layout was hard to read. One physician indeed wished we could keep the text horizontal, while another physician and one HI professional also noted that it was hard but that it was the natural way of displaying text on a radial layout, and that "your eyes will adjust to it." All others found it to be no problem.

For Q5, three physicians said they would gladly spend time on adding the links because it would save them much time when telling the next physician or the patient what was going on. Two physicians said they were not sure about the usefulness of the links, but they would add them. One physician said that he probably would not use this function. Finally, the two HI professionals said that this was an interesting function and they liked it, but since they were not physicians they could not tell under what circumstances they would use it.

For Q6, the two HI professionals thought the interactions provided a good way to let them focus on what they thought was interesting. Three of the physicians said that it would not affect the mental map because it was easy to figure out where everything was. Two physicians thought the same node should always stay at the same location, and one was not sure. But when we asked the latter three whether it was necessary to remove these interactions, they all said that they would like to keep them, and mentioned that although they did not use them when learning about the patient history, it could be helpful when making a report because they would have control over where to put the important nodes.

For Q7, four physicians thought our system would definitely save them time compared to the traditional EMR system they used. The other two were not sure, stating that they were very familiar with the current system and were satisfied with it. One of them said that a "well written paper report would actually save your time" and that she "already got used to it." But she admitted that very few reports were actually well written. The two HI professionals liked the system exceedingly well.

For Q8, three subjects said they would not be able to give a score unless they used the system intensively. The remaining five scores are: 5; 6; 7; 9 but has the potential to get 10; 7 for the sequential and 9 for the radial layout.

We find these evaluation results to be very encouraging, especially the fact that after a short tutorial most of the participants were comfortable with our system. We also gathered many valuable suggestions. One was that in case of large data, to reduce clutter, we could have some "prefilters" that would first pull records within some time frame, to let physicians retrieve the information that they think is important or interesting and then work on only what is left. Another suggestion was to reduce the size of the body map in the radial display; the body map did not need to be high resolution because doctors were familiar with the location references. This way we could save space for the outer rings to make the texts more readable. Also, among all features, the one that the physicians liked the most was the browsing highlight mode (see Fig. 9). They mentioned that "this feature can really help to explain what was going on with the patient."

8.3 Coding Support

We also demonstrated our system to a group of six medical coders employed at our institution. The group leader has been working on coding for more than 15 years. All the others have more than three years of coding experience.

They were very positive and said that the system could save tremendously in time, 15 s to 1 minute for each code lookup. Usually each coder deals with about 80-100 codes per day, so a rough estimation of the time that can be saved by using the system is 30 minutes to 1.5 hours for each person. Furthermore, they mentioned that the system would lead to a much more accurate coding and so reduce the time required for insurance claims to go through. Finally, they praised our hierarchical radial interface as an excellent platform for training, as the hierarchies are visually well presented and fully interactive—as opposed to the large books that are presently in use.

9 Discussion

Presently, our system enables doctors to 1) quickly browse a patient's medical history via both the radial and sequential interface, and 2) enter new medical information using various input widgets in the sequential interface. For example, doctors can enter the name/description of a medical facet either via a textbox with full typing support or via searchable and scrollable lists of terms associated with standardized medical codes. This input interface is similar to that used in current commercial EMR systems.

We acknowledge, however, that while standardized medical codes do carry a great deal of information, they are not rich enough to capture all possible medical findings. During the exam, it may have been determined that the heart beat was normal, a tumor was benign, or the blood pressure was only slightly elevated. This information can be very valuable for future diagnoses and it is why doctors always resort to the patient's medical reports to get the full picture. Extending our system beyond the constraints of medical billing codes is a current focus of our work. Our first step in that direction was to provide doctors with convenient interfaces for adding links and severity levels. Other information can be accessed with a history browser which has hot links to the actual medical report(s) or image(s) (Fig. 12) associated with the corresponding node.

10 Conclusions

We have presented the Five W's scheme of information gathering and reporting, with a special application to health care informatics. We have shown and evaluated that our framework can significantly lower the time and effort needed to access the patient's medical information, which is essential to arrive at a diagnostic conclusion. Finally, it was interesting to see that our system could also be helpful to medical coders.

For limitation and future work, currently our framework requires that all text strings have a length of 10 or less to fit into the display text boxes. Section 5.1.6 discussed our current partially semiautomated techniques to abbreviate the long medical terms. But a more automated and general approach would be desirable, perhaps one based on clustering of the corpus of all ICD codes. Second, scalability to large data is another issue we have only partially addressed so far (by simple filtering). For example, chronic patients can amass a tremendous amount of EMR records over the years, producing an abundance of nodes and edges that clutter the display and make browsing difficult. Here, we think that compounding filtering sequences into single shortcut buttons, possibly even triggered contextually when clicking on a specific node can provide a viable solution. For example, one button-click could lead to showing only treatments to severe symptoms of a particular branch of anatomy or physiology. Third, we acknowledge that some doctors may feel too busy to rate severities or add links between nodes. Yet, these unrated or unlinked records can still be visualized and filtered as entities, and can so contribute to the patient assessment along with the linked medical report. But we were pleased to see that half of the doctors of our pilot study would be willing to add links and ratings.

Moreover, to aid in medical diagnostics as well as in ratings we could take advantage of the wealth of population statistics. For instance, by incorporating patient cohort analysis [19], [33], [38], doctors could start from a single patient's record, find a similar patient cohort, and then make predictions based on this. Finally, we are also currently exploring other application domains in which a Five Ws-based information organization can help in visualization tasks, such as business analytics.

Acknowledgments

This research was supported in part by US National Science Foundation (NSF) grants 1050477, 0959979, and 1117132, and a seed grant from Stony Brook University Hospitals and CEWIT. The authors thank our former undergrads Arunesh Mittal and Alex Dimitriyadi for early contributions to the work. Finally, they thank Paul Nagy, Gorkem Sevinc, Michael Cohen, and the six physicians and residents and the two health informatics professionals at Johns Hopkins University for their kind collaboration and help in evaluating our system.

References

[1] W. Aigner and S. Miksch, "Supporting Protocol-Based Care in Medicine via Multiple Coordinated Views," *Proc. Second Int'l Conf. Coordinated and Multiple Views in Exploratory Visualization*, pp. 118-129, 2004.

[2] W. Aigner, S. Miksch, W. Müller, H. Schumann, and C. Tominski, "Visual Methods for Analyzing Time-Oriented Data," *IEEE Trans. Visualization and Computer Graphics*, vol. 14, no. 1, pp. 47-60, Jan./Feb. 2008.

[3] R. Bade, S. Schlechtweg, and S. Miksch, "Connecting Time-Oriented Data and Information to a Coherent Interactive Visualization," *Proc. SIGCHI Conf. Human Factors Computing Systems (CHI)*, pp. 105-112, 2004.

[4] J. Bertin, *Semiology of Graphics: Diagrams, Networks, Maps.* Univ. of Wisconsin Press, 1983.

[5] C. Brewer, "Color Use Guidelines for Data Representation," *Proc. Section Statistical Graphics*, pp. 55-60, 1999.

[6] W. Cleveland, *Dynamic Graphics for Statistics.* Springer, 1988.

[7] C. Collins, S. Carpendale, and G. Penn, "Docuburst: Visualizing Document Content Using Language Structure," *Proc. Eurographics/IEEE-VGTC Symp. Visualization (EuroVis)*, pp. 1039-1046, 2009.

[8] S. Diehl, F. Beck, and M. Burch, "Uncovering Strengths and Weaknesses of Radial Visualizations—An Empirical Approach," *IEEE Trans. Visualization and Computer Graphics*, vol. 16, no. 6, pp. 935-942, Nov./Dec. 2010.

[9] N. Elmqvist and J. Fekete, "Hierarchical Aggregation for Information Visualization: Overview, Techniques, and Design Guidelines," *IEEE Trans. Visualization and Computer Graphics*, vol. 16, no. 3, pp. 439-454, May 2010.

[10] A. Faiola and S. Hillier, "Multivariate Relational Visualization of Complex Clinical Datasets in a Critical Care Setting: A Data Visualization Interactive Prototype," *Proc. 10th Int'l Conf. Information Visualization*, pp. 460-468, 2006.

[11] J. Fox and R. Thomson, "Decision Support and Disease Management: A Logic Engineering Approach," *IEEE Trans. Information Technology in Biomedicine*, vol. 2, no. 4, pp. 217-228, Dec. 1998.

[12] D. Hadorn, "Use of Algorithms in Clinical Practice Guideline Development: Methodology Perspectives," *AHCPR Publication*, vol. 9, no. 95 pp. 93-104, 1995.

[13] J. Heer, M. Bostock, and V. Ogievetsky, "A Tour through the Visualization Zoo," *Comm. ACM*, vol. 53, no. 6, pp. 59-67, 2010.

[14] D. Holten, "Hierarchical Edge Bundles: Visualization of Adjacency Relations in Hierarchical Data," *IEEE Trans. Visualization and Computer Graphics*, vol. 12, no. 5, pp. 741-748, Sept./Oct. 2006.

[15] W. Horn, C. Popow, and L. Unterasinger, "Support for Fast Comprehension of ICU Data: Visualization Using Metaphor Graphics," *Methods of Information in Medicine*, vol. 40, pp. 421-424, 2001.

[16] B. Johnson and B. Shneiderman, "Tree Maps: A Space-Filling Approach to the Visualization of Hierarchical Information Structures," *Proc. IEEE Conf. Visualization*, pp. 284-291, 1991.

[17] R. Kosara and S. Miksch, "Visualization Techniques for Time-Oriented, Skeletal Plans in Medical Therapy Planning," *Proc. Joint European Conf. Artificial Intelligence in Medicine and Medical Decision Making*, pp. 291-300, 1999.

[18] G. Langelier, H. Sahraoui, and P. Poulin, "Visualization-based Analysis of Quality for Large-Scale Software Systems," *Proc. IEEE/ACM Int'l Conf. Automated Software Eng.*, pp. 214-223, 2005.

[19] L. Lins, M. Heilbrun, J. Freire, and C. Silva, "VisCareTrails: Visualizing Trails in the Electronic Health Record with Timed Word Trees, a Pancreas Cancer Use Case," *Proc. IEEE Visual Analytics in Health Care (VAHC) Workshop*, 2011.

[20] C. North and B. Shneiderman, "Snap-Together Visualization: Can Users Construct and Operate Coordinated Visualizations?" *Int'l J. Human-Computer Studies*, vol. 53, no. 5, pp. 715-739, 2000.

[21] P. Ordonez, M. desJardins, M. Lombardi, C. Lehmann, and J. Fackler, "An Animated Multivariate Visualization for Physiological and Clinical Data," *Proc. ACM Int'l Health Informatics Symp.*, pp. 771-779, 2010.

[22] C. Plaisant, R. Mushlin, A. Snyder, J. Li, D. Heller, and B. Shneiderman, "Lifelines: Using Visualization to Enhance Navigation and Analysis of Patient Records," *Proc. AMIA Ann. Symp.*, pp. 76-80, 1998.

[23] S. Powsner and E. Tufte, "Graphical Summary of Patient Status," *The Lancet*, vol. 344, no. 8919, pp. 386-389, 1994.

[24] T. Ropinski, I. Viola, M. Bierman, H. Hauser, and K. Hinrichs, "Multimodal Visualization with Interactive Closeups," *Proc. EGUK Theory and Practice of Computer Graphics (TPCG)*, 2009.

[25] J. Stasko and E. Zhang, "Focus+Context Display and Navigation Techniques for Enhancing Radial, Space-Filling Hierarchy Visualizations," *Proc. IEEE Symp. Information Visualization*, pp. 57-65, 2000.

[26] S. Quaglini, M. Stefanelli, G. Lanzola, V. Caporusso, and S. Panzarasa, "Flexible Guideline-Based Patient Careflow Systems," *Artificial Intelligence in Medicine*, vol. 22, no. 1, pp. 65-80, 2001.

[27] A. Rind, S. Miksch, W. Aigner, T. Turic, and M. Pohl, "VisuExplore: Gaining New Medical Insights from Visual Exploration," *Proc. Int'l Workshop Interactive Systems Healthcare*, pp. 149-152, 2010.

[28] G. Savov et al., "Mayo Clinical Text Analysis and Knowledge Extraction System (cTAKES): Architecture, Component Evaluation and Applications," *J. Am. Medical Informatics Assoc.*, vol. 17, no. 5, pp. 507-513, 2010.

[29] E. Segel and J. Heer, "Narrative Visualization: Telling Stories with Data," *IEEE Trans. Visualization and Computer Graphics*, vol. 16, no. 6, pp. 1139-1148, Nov. 2010.

[30] Y. Shahar, D. Goren-bar, D. Boaz, and G. Tahan, Tistributed, Intelligent, Interactive Visualization and Exploration of Time-Oriented Clinical Data and Their Abstractions," *J. Artificial Intelligence in Medicine*, vol. 38, no. 2, pp. 115-135, 2006.

[31] B. Shneiderman, "The Eyes Have It: A Task by Data Type Taxonomy for Information Visualizations," *Proc. IEEE Symp. Visual Languages*, pp. 336-343, 1996.

[32] C. Weaver, "Cross-Filtered Views for Multidimensional Visual Analysis," *IEEE Trans. Visualization and Computer Graphics*, vol. 16, no. 2, pp. 192-204, Mar./Apr. 2010.

[33] K. Wongsuphasawat and D. Gotz, "Exploring Flow, Factors, and Outcomes of Temporal Event Sequences with the Outflow Visualization," *IEEE Trans. Visualization and Computer Graphics*, vol. 18, no. 12, pp. 2659- 2668, Dec. 2012.

[34] M. Workman, M.F. Lesser, and J. Kim, "An Exploratory Study of Cognitive Load in Diagnosing Patient Conditions," *Int'l J. Quality in Health Care*, vol. 19, no. 3 pp. 127-133, 2007.

[35] L. Young, H. Bazari, M. Durand, and J. Branda, "Case 33-2010 — A 22-Year-Old Woman with Blurred Vision and Renal Failure," *New England J Medicine*, vol. 363, 1749-1758, 2010.

[36] Z. Zhang, A. Mittal, S. Garg, A. Dimitriyadi, I.V. Ramakrishnan, R. Zhao, A. Viccellio, and K. Mueller, "A Visual Analytics Framework for Emergency Room Clinical Encounters," *Proc. IEEE Visual Analytics in Health Care (VAHC) Workshop*, 2010.

[37] Z. Zhang, F. Ahmed, A. Mittal, I.V. Ramakrishnan, R. Zhao, and A. Viccellio, and K. Mueller, "AnamneVis: A Framework for the Visualization of Patient History and Medical Diagnostics Chains," *Proc. IEEE Visual Analytics in Health Care (VAHC) Workshop*, 2011.

[38] Z. Zhang, D. Gotz, and A. Perer, "Interactive Visual Patient Cohort Analysis," *Proc. IEEE Visual Analytics in Health Care (VAHC) Workshop*, 2012.

[39] FLARE: http://flare.prefuse.org/, 2013.

[40] "Defining and Testing EMR Usability, Healthcare Information and Management Systems Society (HIMSS),"http://www.himss.org/content/files/himss_definngand testingemrusability.pdf, June 2009.

[41] http://lane.stanford.edu/portals/cvicu/HCP_Neuro_Tab_4/0-10_Pain_Scale.pdf, 2013.

Zhiyuan Zhang received the BE degree from Shandong University, China, and is currently working toward the PhD degree in computer science at Stony Brook University. His research interests include visual analytics and information visualization, with a special focus on healthcare informatics, high-dimensional data visualization, and correlation analysis. He is a student member of the IEEE.

Bing Wang received the BE degree from the University of Science and Technology of China, and is currently working toward the PhD degree in the Computer Science Department at Stony Brook University. Her research interests include healthcare visual analytics and high-dimensional data visualization.

Faisal Ahmed received the BS degree from BUET, and the PhD degree in computer science from Stony Brook University in 2012. His research interests include web mining, web accessibility, and HCI.

I.V. Ramakrishnan is a professor of computer science at Stony Brook University. His research includes AI, in particular computing with logic, machine learning, and their applications in human-computer interaction, information extraction, and accessible IT.

Rong Zhao is the director of software systems division at CEWIT, New York, a computer science faculty at Stony Brook University. His research includes information retrieval, data mining, medical informatics.

Asa Viccellio received the MD degree from Harvard University. He is the clinical director of the Department of Emergency Medicine at Stony Brook Hospital. He received the 2005 National ACEP Colin C. Rorrie, Jr. Award for Excellence in Health Policy.

Klaus Mueller received the PhD degree in computer science from Ohio State University. He is a professor of computer science at Stony Brook University. His research interests include visualization, visual analytics, and medical imaging. He received the US National Science Foundation (NSF) CAREER award in 2001 and the SUNY Chancellor's Award in 2011. He is a senior member of the IEEE.

▷ **For more information on this or any other computing topic, please visit our Digital Library at** www.computer.org/publications/dlib.