Medication visualization and cohort specification

Dingen, D.J.W.G.M.

Award date:
2015

Department of Mathematics and Computer Science
Algorithms and Visualization W&I Group

Medication Visualization And Cohort Specification

Master's Thesis

Dennis J.W.G.M. Dingen

Supervisors:
prof. dr. ir. Jack J. van Wijk
prof. dr. ir. Johan B.A.M. Arends
dr. ir. Joos C.A.M. Buijs

Eindhoven, September 2015

Abstract

Kempenhaeghe is an expertise centre for epileptology, sleep medicine, and neurocognition. Doctors gather information about epilepsy patients in order to respond with adequate surgery and/or medication. Per patient, all data on symptoms, treatments, events and notes are stored in an information system in the form of a record that is known as an *Electronic Health Record (EHR)*. These records ease the monitoring of the progress of the patient over time. Since the records are stored textually, however, both the effectiveness and efficiency of the process of gathering deep(er) insight leave much room for improvement. This holds for a doctor working with the EHR of a patient and also for a clinical researcher studying the EHR data of cohorts of patients.

For Kempenhaeghe we propose a modernized version of their Electronic Health Records system that harnesses the power of interactive visualization to improve the clinical interaction. On the clinical side, it offers a flexible method for specifying a cohort of patients that can also be explored using interactive visual techniques, aiding the researchers at Kempenhaeghe with their studies. Finally, a bridge is proposed and partly implemented between these two modes of operandi of the system, to the benefit of both.

Preface

For me to work on a project in healthcare is rewarding, because in the years prior to commencing with this project, I had been challenged in the area of health myself. In those years, while I was forced to pause working on my Master's degree in Computer Science, I learned to better appreciate the healthcare I received when I was dependent on it. Hence, contributing to it, albeit indirectly, is a nice way to express my appreciation.

After my healing process was finally completed, prof. Jack van Wijk welcomed me back and offered me the graduation project, which I gratefully accepted, that led to this thesis. His patience, encouragement, support and guidance were, and remained, very motivational and valuable. Therefore, I underline what many others have said in praise and have written in their acknowledgements about prof. van Wijk. He is a dependable, knowledgeable and enjoyable supervisor who knows how to get the best out of you.

I am also very pleased with the support and collaboration on the side of Kempenhaeghe, stakeholder in this joint project, in the persons of prof. Johan Arends (TU/e, Kempenhaeghe) and Dr. Emmeke Wammes - van der Heijden (Kempenhaeghe, St. Anna hospital). The meetings were always constructive, encouraging and sociable. On a regular basis, Dr. Wammes - van der Heijden provided the necessary domain knowledge and made sure that the (practical) value the system we have designed was maintained. During the progress meetings, prof. Arends provided valuable feedback and pointed out more opportunities, making sure that I could go as far as reasonable possible. Besides their knowledge, Dr. Wammes - van der Heijden and prof. Arends shared sources and experience with me, all of which I gladly absorbed.

Furthermore, I thank Paul van der Corput for helping me on my way with this project and in general all people at the university for their advice and tips. Part of my official exam committee, as external member, was Dr. Joos Buijs, who I like to thank for his participation and for posting his master's thesis template on his blog. For the acquirement of basic web development skills required for this project I pay tribute to my colleagues at Code Yellow B.V., in particular Burhan Zainuddin and Rob Wu, for sharing their knowledge on web development stacks and Javascript with me.

Finally, I am grateful for the support of my parents, the rest of my family and friends, especially, to my brother Angelo and my friends Ron and Marcel for their support, testing and proofreading. Last but not least, I thank my girlfriend Simone for her support and understanding. She really was the wind beneath my wings.

Contents

Chapter 1

Introduction

Kempenhaeghe is an expertise centre for epileptology, sleep medicine, and neurocognition. Doctors gather information about epilepsy patients in order to respond with adequate surgery and/or medication. Per patient, all data on symptoms, treatments, events and notes are stored in an information system in the form of a record that is known as an *Electronic Health Record (EHR)*. These records ease the monitoring of the progress of the patient over time. Since the records are stored textually, however, both the effectiveness and efficiency of the process of gathering deep(er) insight leave much room for improvement. The focus of this project is on elevating that situation by employing interactive visualization techniques during the exploration of the data.

1.1 Focus

With the focus on improving the capabilities of the information system of Kempenhaeghe we immediately take note of an important observation reported by *Rind et al.* [24], after performing an extensive survey on EHR systems. In practice, there is a sharp distinction between the type of EHR systems designed for processing and showing records on a per patient basis, and the type of systems designed for processing and showing data from multiple patient records at once. We refer to these two types as a *Single Patient System (SPS)* and *Multiple Patient System (MPS)* respectively. The MPSs are intrinsically better suited for researchers to conduct clinical research with. For more details on SPSs vs MPSs, see Chapter 2.

In their paper, Rind et al. point out research possibilities on the integration of richer *Visual Analytics* methodology in MPSs, particularly by means of *visual querying*. Lastly, they pose the challenge of developing systems that bridge the gap between SPSs and MPSs. For an overview of solving problems with Visual Analytics see the book edited by *Keim et al.* [15].

Assessment of the situation at Kempenhaeghe showed that doctors mainly work with an SPS and that clinical research is mainly conducted using spreadsheets. This situation gives rise to the following research questions:

1. Can we support the interactive visualization of single patient data over time in an enriched way? Is direct comparison with (groups of) similar patients is possible? What about taking into account temporal aspects?
2. On the clinical research side, can we develop methods and techniques to visually and interactively explore clinical data of groups of patients using temporal patterns associated with events / entities on a higher level?

3. How can we combine interactive visualization techniques for SPSs with techniques for MPSs supporting clinical research into one system?

1.2 Scope

First of all, this project is a follow up on the project by *Paul van der Corput* [7] regarding the visualization of medicine prescriptions. Paul's project focussed on answering questions about the relationship between doctors, patients and medicine prescriptions, which is modelled as a hypergraph, and associated statistics within a specified period of time. See Section 4.2 for more details.

In contrast to Paul's project in which the emphasise was laid on relationships between doctors, patients and prescriptions, in this project the emphasis lays on the perspective from the patient and / or the doctor during the clinical interaction and clinical research. Secondly, the temporal aspect is considered in more depth from both the SPS viewpoint and the MPS, the latter in particular for clinical research. For the clinical research viewpoint this means that a doctor is interested in researching cohorts of patients. A system designed from that viewpoint is limited in providing insight in questions surrounding doctors or medication, although some of such questions can still be answered by creative use of the system. Thirdly, additional patient data on weight / length, blood compounds, adverse effects and epileptic seizures is taken into account.

The resulting system proposed for this project consists of two subsystems. The first subsystem is designed to evolve the capabilities of the SPS, which currently in use at Kempenhaeghe, while the second is an MPS supporting clinical research. Furthermore, some ways of bridging the (conceptual) gap between the two subsystems are proposed and partly prototyped.

1.3 Organization

The background is discussed in Chapter 2. In Chapter 3 the problem is analysed and the research objectives are stated. The state of the art is covered Chapter 4. The solution is presented in Chapter 5 and Chapter 6. A brief overview of the implementation given in Chapter 7. Finally, the evaluation is discussed in Chapter 8 and the work is concluded in Chapter 9.

Chapter 2

Background

In this chapter, the background of EHR systems is covered. The reader is familiarized with relevant notions and concepts from the literature on EHR systems, such that a contextual basis is formed for the following chapters.

2.1 The need for interactive visualization methods

There is an increasing amount of data stored by the medical profession about patients, events, treatments, medicines and more. For the utilization of the stored data, more effective information retrieval methods are required over time, and thereby, richer data structures as well. This alone, however, does not necessarily lead to better insight in the retrieved data, nor to the improvement of the quality of care.

According to the extensive survey by *Rind et al.* [24], the initial EHR systems have been shown to have little positive effects on the quality of care, and have in some cases even decreased the quality of care. This was largely because EHR systems failed to provide cognitive support to healthcare providers, patients and families. In reaction to this issue, the potential of applying *Information Visualization* techniques was explored.

This led to various systems dealing with patient data in a graphical way. Rind et al. compare these systems based upon whether the system is designed to deal with a single record at a time or a collection of EHRs at once, the supported data types, the number of variables it deals with simultaneously and which user intents are supported. These distinctions are detailed in the section below.

In current research, the focus is shifting towards adding more interactive visualization methods [5]. To that end it is suggested by Rind et al. to employ more *Visual Analytics* techniques.

2.2 Single Patient Systems vs Multiple Patient Systems

In their survey Rind et al. distinguish between systems designed for exploring single patient records and systems designed for exploring collections of patient records at once. The former we refer to as a *Single Patient System (SPS)* and the latter as *Multiple Patient System (MPS)*. The distinction is due to the fact that the requirements of these two types of systems differ and may also imply different software architectures.

An SPS is usually designed for dealing with patient data during treatment of the patient, whereas the

MPS is usually designed for comparing patients with other patients or investigating the properties of a group of patients. Therefore, the MPS is intrinsically better suited for conducting clinical research. It is an open research questions to what extent the gap between the two types can be bridged to form a more cohesive EHR systems.

2.3 Data types

There are two basic data types commonly supported by an EHR: *categorical* data and *numerical* data. Categorical data refer to variables that take on a particular value out of a finite group of possible values. If such a group consists of values that are unordered, the variable is called *nominal*. An example of this is the diagnosis of a syndrome. If the group consists of values that have an order, then the variable is an *ordinal* variable. The severity of an event, for instance, may take on the value "low", "medium" or 'high".

Numerical data refer to variables that take on the value of a measurement in the form of a number, like the weight of a patient. A numerical variable can either have a discrete or continuous domain and represent an interval or a ratio.

Since patient data evolve over time, many of the variables change as a function of time. Therefore, a typical SPS system includes a (navigable) timeline for plotting the variables. A typical MPS may include such a timeline for comparing groups of patients, but can also show aggregates.

Some representative visualization methods used for data over time in medical data analysis have been surveyed by *Kosara and Miksch* [17]. Methods for visualization measured (numerical) data over time are illustrated in Fig. 2.1. Besides the common line plot, small multiples of *glyphs* are used to encode multiple facets into a compound of geometric shapes that change over time. Related to this technique is the dynamic star plot, where variables are plotted along multiple axes and the variables are connected via a polygon for small time intervals. Lastly, timeline distortion is a common technique to put certain periods into perspectives or to emphasize temporal patterns.

Methods for visualizing categorical data on events and symptoms over time are illustrated in Fig. 2.2. The first method is to use bars for the indication of the period of an event. Color can be used for categorization and periods can be merged as a form of temporal abstraction. There are the Japanese *Shinkansen* diagrams that were originally used for depicting many different trains visiting many different stations. Again spirals, or in this case, concentric circles can be used for revealing structures during certain fixed time periods. Finally, the *time cube* method is used for showing spatio-temporal data and enables locating events in time and in space as well. In this case, two spatial dimensions are used and one time dimension.

The cylinders in the time cube might represent a location, like for example a ski resort, and the line represents the visitor moving between the resorts. In the medical context, incident types may be used on the base plane allowing it to show events or periods of symptoms for a patient. However, the usage of bars seems much more common in comparison with other methods for visualizing events over time.

The various methods have been evaluated by *Kosara and Miksch* for intuitiveness, focus+context time and finding other relevant aspects, see Fig. 2.3. Lastly, an interesting feature an EHR system may exhibit is called *semantic zoom*, see Fig. 2.4. From top to bottom the data becomes less summarized, revealing more details.

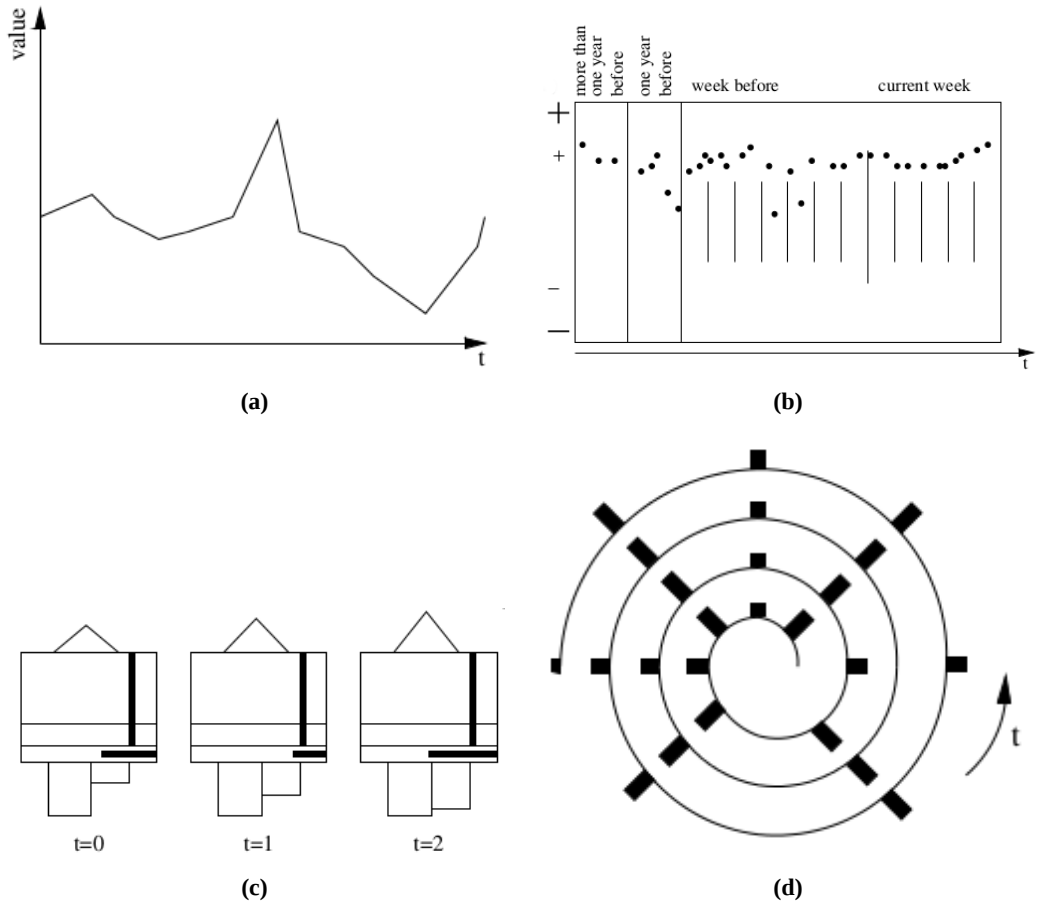


Figure 2.1: Methods for recording data over time (*Kosara and Miksch* [17]). (a) Line plot. (b) Timeline distortion, where larger periods can be summarized. (c) Small multiples of glyphs [12]. The relative size and shapes of the geometric subshapes encode variables. In addition color can be used. (d) The spiral timeline emphasize temporal patterns, by showing data close to each other temporarily, which are otherwise less visible.

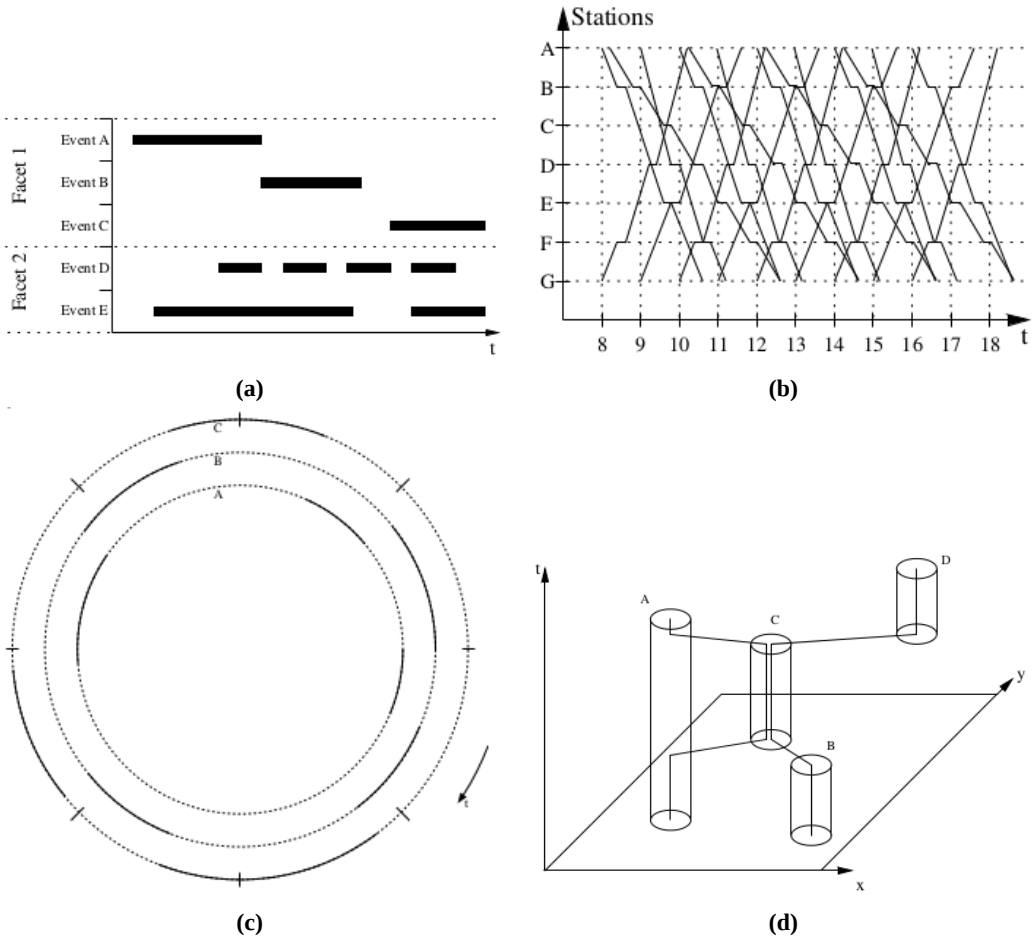


Figure 2.2: Methods for showing data on events over time (*Kosara and Miksch* [17]). (a) Bars. (b) Japanese Shinkansen trains diagram. (c) Concentric circles for revealing structures in a fixed time period, similar to the time spiral. (d) The time cube, where cylinders may represent locations or periods of events and lines encode the pattern an entity generates.

Method	Intuitiveness	Focus+Context Time	Focus+Context Data	Combining Values	Seeing	Developments	Finding	Patterns	Discovering	Intervals
Chart	•					•		•		
Graphical Summary	•	•		•		•				
VIE-VISU		•		•		•				
Spirals	•	•							•	

(a)

Method	Intuitiveness	Focus+Context Time	Focus+Context Data	Seeing	Developments	Finding	Patterns	Discovering	Intervals
LifeLines	•		•		•		•		
Graphical Summary	•	•							
Shinkansen							•		
Circles	•								(•)
Time Cube					•		•		
Spirals	•	•							•

(b)

Figure 2.3: Comparison of visualization methods (*Kosara and Miksch* [17]). •: Positively supports the feature or exhibits the property, (•): Partly supports the feature or exhibits the property, " ": no support or does not have the property. (a) Methods for visualizing measured (numerical) data. (b) Methods for visualizing categorical data.

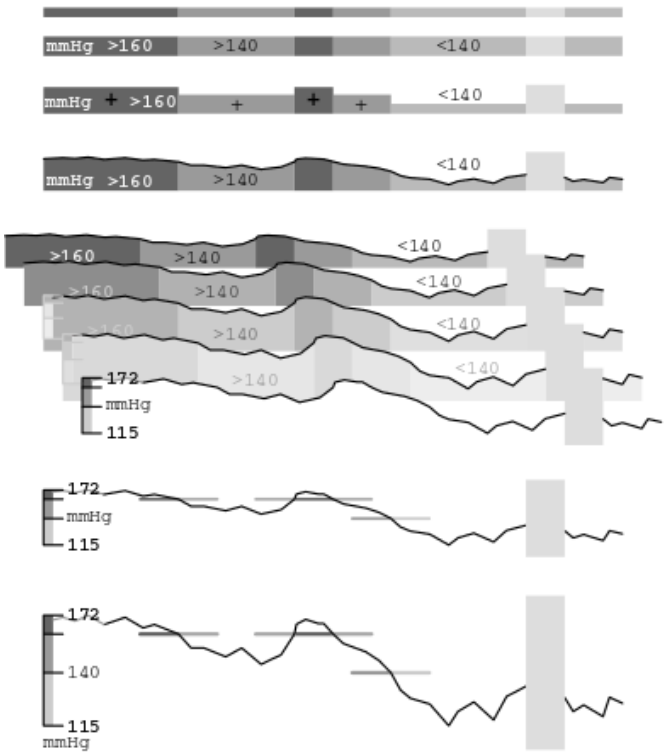


Figure 2.4: Semantic zoom. Less details are shown at the top than at the bottom, as one "zooms in semantically", *Bade et al.* [1].

2.4 Number of variables

Whether we are dealing with an SPS or MPS, in the end the system has to show data and only a limited number of variables fit on the screen at any given moment. An SPS may show a timeline with medication and weight over time and a MPS may show the correlation of two variables in a scatter plot. The actual number of variables that can be shown on the screen simultaneously is a distinguishing property of EHR systems. The more variables are visible the better the overview and / or correlation finding becomes, which implies less searching time.

2.5 User intent model

For comparing the interaction features, Rind et al. have extended the user intent model by *Yi et al* [37] with 20 subintent categories:

1. **Select:** Mark a subset of the dataset as interesting.
    - to keep track of selected items for a short term while the visualization is changed
    - to manage groups, for example adding or removing patients to groups
2. **Explore:** Show a different part of the dataset.
    - to navigate in time (e.g., panning and zooming the time axis)
    - to add or remove parameters to the visualization
    - to add or remove patients to the visualization
3. **Reconfigure:** Rearrange the visual layout of the dataset.
    - by repositioning items manually (freely or by some constraints)
    - by sorting items along an axis
    - by other adjustments of an axis (e.g., alignment to a relative timescale, distortion to see some items in focus and some in context)
    - by applying another technique to avoid occlusion (e.g., 3D camera movement)
4. **Encode:** Change the way each item of the dataset is represented.
    - by switching to a different visualization technique or opening it in a different view
    - by varying visual encoding (e.g., map outcome to item color, encode severity as item size)
5. **Abstract/Elaborate:** Show less or more detail.
    - by abstraction of one or more parameters
    - by temporal data binning (e.g., aggregate parameter values either by fixed time intervals or for as long as the have the same value)
    - by showing details of items (e.g., in a tooltip)
6. **Filter:** Show or highlight something conditionally.
    - by patient status without considering time or development over time
    - by development over time like event sequences (e.g., surgery after stroke) or value trends (e.g., increasing cholesterol) without time constraints
    - by time constraints (e.g., relapse within 3 weeks after discharge, surgery in May 2009)

7. **Connect:** Show related data.
    - to show patient/patient group relationship
    - to brush items in other representations
    - to brush items for other variables at the same point of time or of the same patient

Most EHR systems support the majority of the user intents. In next sections, several prominent EHR systems are highlighted.

2.6 Analytic focussing strategies

A taxonomy of analytic focussing strategies for coping with big data was presented by *Shneiderman et al.* [26]. By only taking the relevant records, event types, or events from a large dataset, the detection of meaningful patterns becomes easier. The ten strategies are:

1. **Goal-driven record extraction:** Narrowing down the sets of record to only those of interest for further study.
2. **Goal-driven event extraction:** If a large portion of the record in a database is required for the study, an analyst may start with a small portion of the associated events and gradually add more.
3. **Temporal windowing:** Filter out events in time using an absolute time intervals or relative ones.
4. **Random sampling of records:** If the previous strategies fail, the next reasonable strategy may be to search for prevalence of patterns in a set of (randomly) sampled records.
5. **Temporal folding:** A long stream of temporal data may be broken down into, for instance, weekly records, revealing patterns otherwise not revealed.
6. **Grouping event categories (aggregation):** To reduce data size and highlight global patterns, multiple data events within a specific class can be grouped into one.
7. **Selecting sentinel events in a stream:** Only selected events, like the 1th, 10th or 1000th are shown, while the rest is removed. Multiple records can be aligned on their sentinel events for easy comparison, which is a popular technique that was proposed by *Wang et al.* [31], illustrated in Fig. 2.5.
8. **Converting multiple point events into a single interval event:** A sequence of "normal" numerical data values may be replaced with a more meaningful single interval event indicating a period of "normal" values.
9. **Converting multiple interval events into a single longer interval:** Consecutive events having intervals that overlap or with small gaps between them can be replaced with a longer interval with a more abstract meaning.
10. **Identifying hidden complex events:** In many domains, high-level events may consist of many smaller events. An analyst may tweak the granularity level of the abstraction and see its effects on the visualization. A pattern simplification strategy is either domain specific or goal driven.

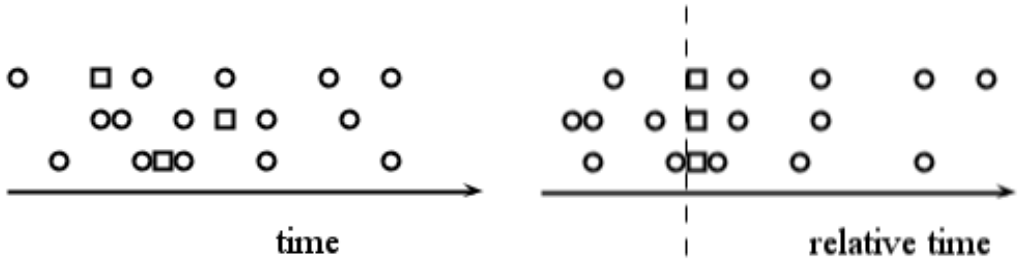


Figure 2.5: Sentinel events technique by *Wang et al.*[31]. (a) Before. (b) After aligning on sentinel events, represented by squares. The similarity in the temporal pattern following the square becomes more visible.

The authors state that the strategies are useful for the extraction of data from a database and remain useful after loading the data in interactive Visual Analytics tools (a *multistep* strategy). Additionally, they state that the accumulated experience of analysts (within their particular application domains) can be used to develop systems featuring automatic recommendation techniques or partially automated workflows.

Chapter 3

Problem statement

The staff of Kempenhaeghe wants to modernize their EHR system. More effective and efficient insight in a patient's EHR leads to better treatment conditions and improves the clinical interaction between doctors and patients. In Section 3.1 both the current and intended future situation at Kempenhaege are discussed. Section 3.2 details the data Kempenhaeghe has collected on the patients. Finally, in Section 3.3 requirements are formulated.

3.1 Current situation and goal

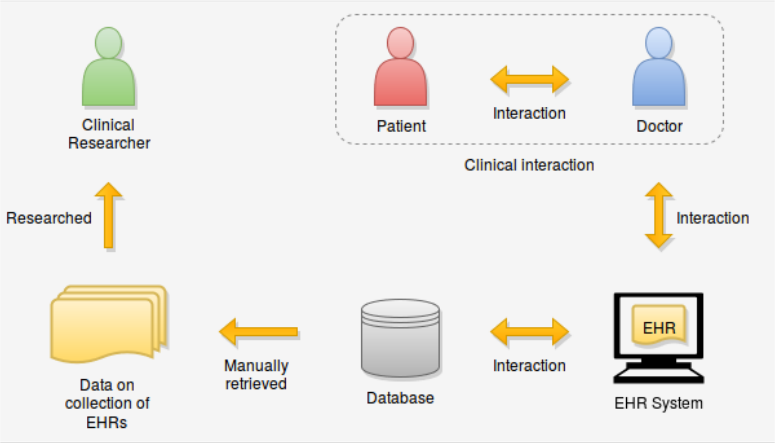
Doctors at Kempenhaeghe are working on treating patients with sleep disorders and epilepsy, which involves diagnosing syndromes and the registration of seizures, side effects of medicines, the level of compounds in the blood and the well-being of the patient over time. A patient may visit the doctor at Kempenhaeghe or report on an event that has occurred. This interaction between the doctor and the patient is called *clinical interaction*.

To support doctors with their work, Kempenhaeghe currently offers an EHR system that interacts with the patient database, as shown in Fig. 3.1a. Since the EHRs are stored in tabular format in the database and the current EHR system offers little graphical support tools for gaining insight in an EHR, doctors spend a considerable amount of time scrolling through tables in search for a piece of data. Moreover, the temporal component of the data is neglected. Although there data is sorted on data, there is still no decent temporal overview that allows the doctor to easily interpret variable values in context, by relating it to other variables and events.

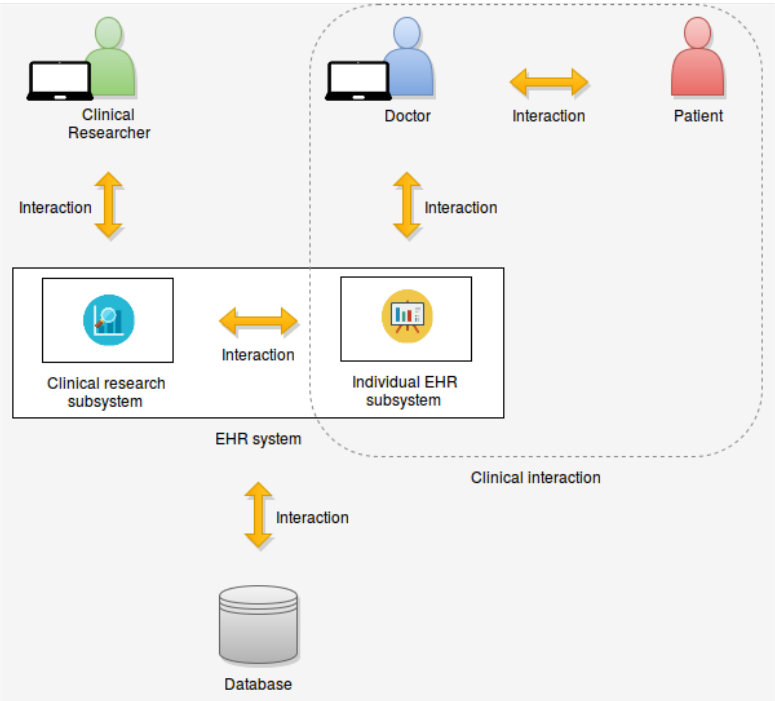
Besides doctors, there are clinical researchers that retrieve collections of EHRs for their studies. In the absence of a support system, the researchers typically retrieve the data from the database in spreadsheet format. This leaves work to the researcher while studying the data. The manual reconstruction of the temporal aspect of the data leads to an increase in study time and the testing of hypotheses pertains a lot of manual work.

The current situation at Kempenhaeghe signifies the need for a better EHR system for the doctor, as well as the clinical researcher. The doctor needs an EHR system that is capable of showing a complete temporal overview of the EHR of a patient, which can be interactively explored. The clinical researcher needs an interactive system for specifying cohorts of patients that can easily be inspected, compared and checked for protocol adherence. The benefit of the fusion of the two systems into one overall system has

to be exploited as much as possible. Fig. 3.1b shows the intended improved situation. For some use cases see Section 8.1.

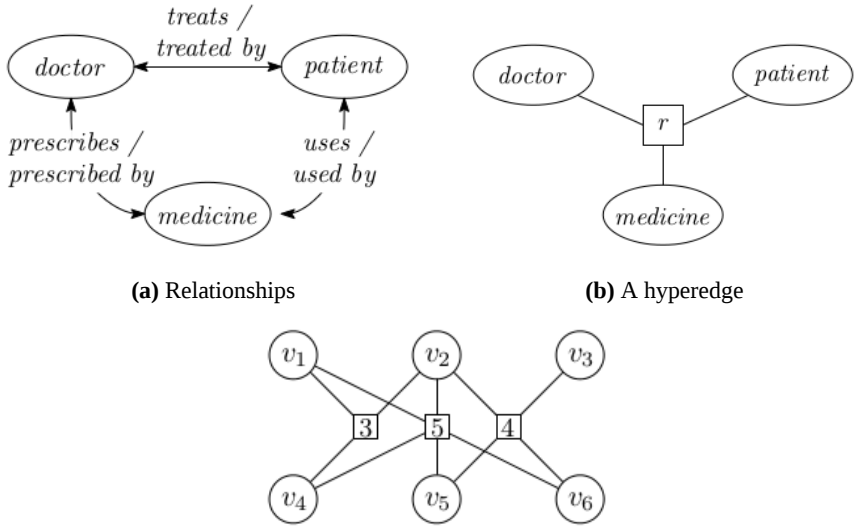


(a) Current situation at Kempenhaeghe.



(b) Intended new situation at Kempenhaeghe.

Figure 3.1: Diagrams of the current and intended situation at Kempenhaeghe. (a) The clinical interaction between a doctor and a patient takes place mainly without involving the EHR system. Researchers manually retrieve data on collections of EHRs from the database for their studies. (b) The clinical interaction now involves the EHR system and researchers have a system in support of their studies. The new system offers high accessibility via modern interfaces and supports easy cooperation between researchers and doctors.



(a) Relationships

(b) A hyperedge

(c) A hypergraph

Figure 3.2: Hypergraph model for entity relationships. Image courtesy of *Paul van der Corput* [7]. (a) The many-to-many relationship(s) between doctors, patients and medicines. (b) The corresponding hyperedge. (c) An example of a hypergraph, where each hyperedge connects multiple vertices.

3.2 Data

The data provided by Kempenhaeghe include EHRs of nearly 33,000 patients and comprises over 1,800 doctors, over 100,000 prescriptions of about 1,000 of the 2,000+ registered medicines, nearly 300,000 seizures, over 400 adverse effects (registered for only 300+ patients) and over 200,000 blood compound measurements. Additionally, nearly 9,000 height measurements and almost 27,500 weight measurements are present. Treatment reports were excluded in compliance with privacy regulations, but an artificial sample was given.

3.2.1 The (extended) hypergraph

With the focus mainly on the relationship between patients, doctors and medicines, *Paul van der Corput* [7] identified the *hypergraph* as a suitable model of that relationship, see Fig. 3.2. In the model, the entities are represented as vertices and connected via an *hyperedge* forming a hypergraph. Because a doctor may be involved in more than one prescription for a single patient, the hypergraph is not a *3-uniform hypergraph*.

In contrast to the work by Van der Corput, we do take into account the data on seizures, adverse effects, blood compounds, height and weights. This extends the hypergraph model, because in addition to relationship tuples of the form (patient, doctor, medicine), there are tuples possible where the patient-doctor pair is related to an entity of class non-medicine. In fact, one can construct a *multipartite graph*, where the partitions reflect the distinct entity classes.

On the most abstract level the entities are related as illustrated in Fig. 3.3. For the full *Entity Relationship Diagram (ERD)*, see Appendix A.

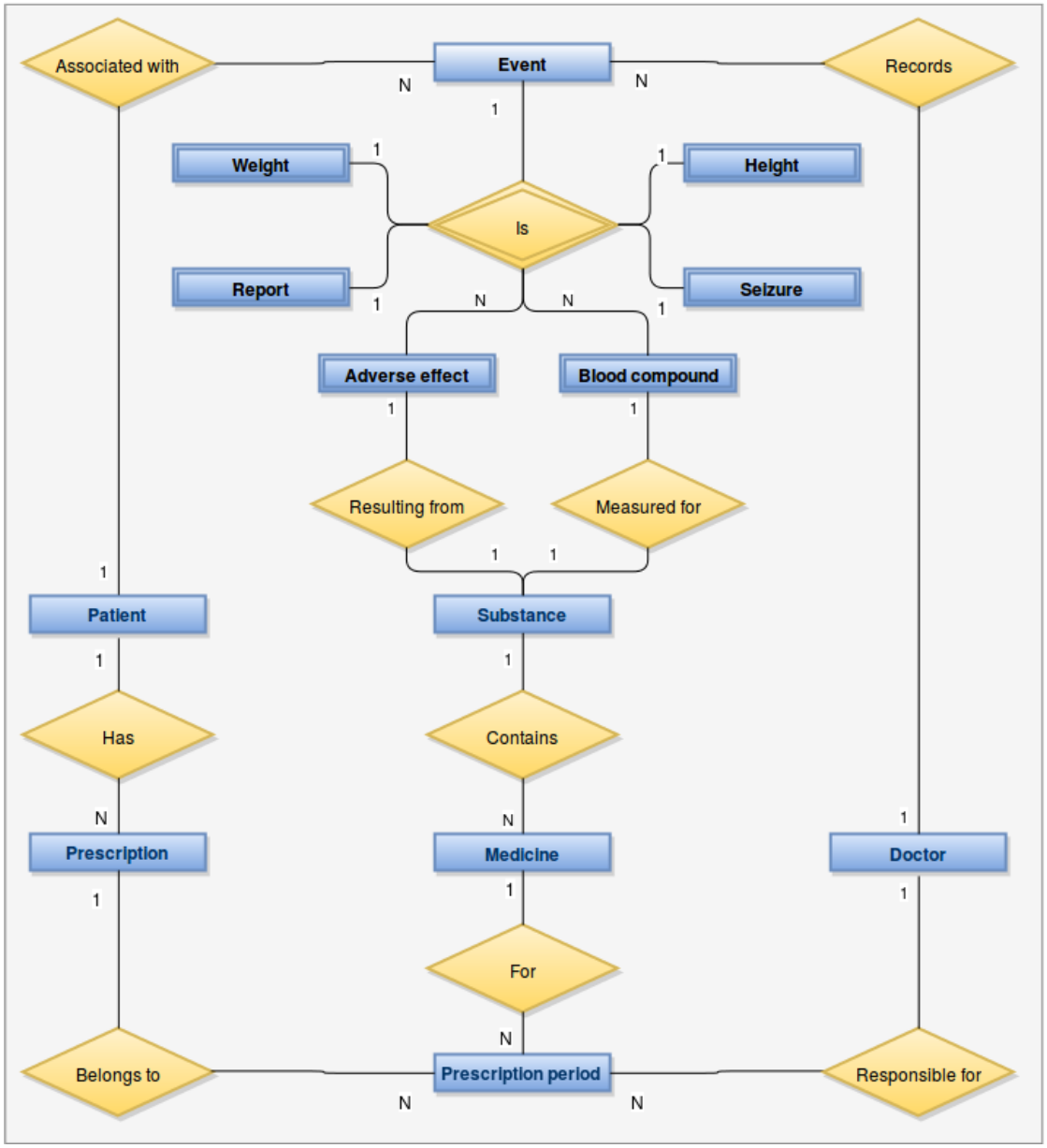


Figure 3.3: The simplified ERD.

3.2.2 Entities detailed

The properties recorded for each entity are listed below and the ones that are not self-explanatory are clarified.

Patient

The patient record holds the: code; name; age (birthdate); gender; IQ class, and main treatment location. The current EHR system at Kempenhaeghe works with birthdates, but for matters of privacy, the data only provides the age.

Doctor

The doctor record holds the: code; name; and functional group. For our purposes, the doctor is mainly part of the context information, because the focus is on the patient and medication.

Substance

The substance record holds the: code; name; and defined daily dose. To compare the dose of one substance with the dose of another substance on a daily basis in a standardized manner, the *Defined Daily Dose (DDD)* is invented. A DDD is the average maintenance dose per day for a substance determined by the *World Health Organization* [20]. Hence, a given dose divided by the DDD is used as the unit of comparison. This also circumvents the problem of dealing with differing units of measurements among dosages.

Medicine

The medicine record holds the: code; name; dose; unit for the dose; ATC code; substance; and anti-epileptic flag. The *ATC code* field classifies the ingredients of the medicine according to the *Anatomical Therapeutic Chemical Classification System*, controlled by the *World Health Organization* [20]. Next, the *substance* is stored, which is the main active ingredient of the medicine. Finally, the *anti-epileptic flag* field records whether the medicine is classified as an anti-epileptic medicine.

Prescription

A prescription record is actually split into two parts, namely the *prescription* and one or more *prescription period(s)*. A prescription record holds the: patient; medicine; substance; start date; and end date. An accompanying **prescription period** records the: related prescription; commencement date; dose duration; daily total dose; unit (of dosage); time (1 through 5) and dose (1 through 5).

One can observe that the overall period is recorded in the prescription and that the dose changes and daily details are recorded in a prescription period. On at most five different times per day a certain dosage is administered to a patient. The start date of first prescription period coincides with the start date prescription, except for "rescue medicines" that are administered in critical situations, for which no prescription periods are recorded.

Adverse effect

The adverse effect record holds the: patient; doctor; medicine; date; serious flag; and notes. An adverse effect is record by a doctor for a patient that is using a particular medicine on a specific date. It is indicated if the adverse effect is considered serious and notes are taken.

Blood compound

The blood compound record holds the: patient; doctor; substance; date; time; value; unit; and study. Blood samples are studied to determine the amount of a given substance in the blood. In some cases, it is necessary to measure the amount of additional substances besides the substance under consideration. For our purposes, these additional substances are referred to as *metabolites*. In the record on blood compounds, the *substance* field indicates the substance, of which the amount is measured, and the *study* field is used to indicate the substance that is actually studied.

Seizure

The seizure record holds the: patient; doctor; seizure class; date; time; count; type; and status epilepticus. The *seizure class* field indicates the type of the seizure. Next, the *type* field records whether the temporal distribution of a series of seizures is considered a "cluster". A *status epilepticus* is set if the seizure lasts for more than five minutes, or, if the patient has a series of seizures lasting for more than five minutes, without recovery in between two seizures. In practice, however, this is rarely the case according to the data.

Height and weight

Both the height and weight records hold the: patient; date; and value.

3.3 Research objectives

As mentioned in Section 1.1 three main questions are considered:

1. Can we support the interactive visualization of single patient data over time in an enriched way? Is direct comparison with (groups of) similar patients is possible? How can we take temporal aspects in account for defining a group?
2. On the clinical research side, can we develop methods and techniques to visually and interactively explore sets of EHRs using temporal patterns associated with events / entities on a higher level?
3. How can we combine interactive visualization techniques for SPSs with techniques for MPSs supporting clinical research into one system?

In order to answer these questions we need to be more specific with the underlying research objectives. Hence, the following objectives are to be met.

1. The development of an interactive visualization method for gaining a quick overview and insight into the EHR of any patient that:
   (a) shows graphical navigable display of medication over time,
   (b) including all (important) events, and,
   (c) offers the ability for searching on events using keywords.
2. The development of a method for comparing a patient (group) directly with another specified, possibly similar, patient group, such that:
   (a) the group definitions by *Paul van der Corput* are integrated,
   (b) a way to investigate a single patient is included,
   (c) the statistical underpinning is (reasonably) considered, and,

   (d) can we generalize the method, for use in other disciplines?

3. The augmentation of the previous method for temporal patterns for:

   (a) the specification of patient groups based on medication, and,

   (b) testing adherence to prescription protocols.

4. Contemplate and / or implement a bridge between the two methods above as an example of how techniques for SPS and MPS can be integrated.

We add the non-functional requirement for the system to be as intuitive as possible. Lastly, in modern practise, *ease of integration* is a highly valued non-functional requirement of a software solution. In adherence to that requirement the prototype is produced in the form of a web application.

Chapter 4

State of the art

Many EHR systems have been developed over time and some leading ones are described in this chapter. This is be no means a complete overview, but the selected ones serve as a representative sample, mostly extracted from the work by *Rind et al.* [24].

4.1 SPS

A classic SPS system is *Lifelines* that is developed by *Plaisant et al.* [22]. The main component of the system is shown in Fig. 4.1. Events and episodes are indicated with bars on a shared time axis, which the user can pan and zoom for more details. If a user clicks on an event, a details panel shows data related to that event. Many later systems adopted the approach of Lifelines.

A more recent system is *Knave II* [25], shown in Fig. 4.2. It offers features similar to Lifelines, but also adds a domain ontology browser and a search mechanism. The domain ontology browser enables the user to dynamically select concepts and data for the visualization. The panels around the charts have buttons for toggling some options in the charts, like highlighting points of measurement on the curve. It is also possible to flip between absolute and relative time scales. The timeline is drawn for each chart separately. The system is widely reported on in literature.

Furthermore, an effort made by *Microsoft* resulted in the *Health Common User Interface* [18], shown in Fig. 4.3. A simple ontology selector is present at the top. Various charts are shown above a common timeline. The scrollbar is used to scroll the chart of choice into view. On top are some general buttons to reset the views and toggle common chart options, like the display of data values. The background of a chart is used to convey information, like common or important ranges of values to put the measured values in context. The visible time period is selected at the top, which may cause the rescaling of the charts. Each chart has a button besides it to fit the chart in the visible area by rescaling the vertical axis. A chart can also be collapsed to save space for other charts in the visible area.

Another part of the system is displayed in Fig. 4.4. The basic idea behind that part is to use small windows showing multiple charts. A scroll bar enables the user to view the chart of choice. Again we have the time period selector on top and a scroll bar at the bottom to pan the timeline. Each window has a level of detail selector to enable semantic zooming. The windows cannot be moved in relation to each other and the medication values are not directly comparable regardless of the level of detail. Note that the charts are put on a single timeline, which, according to Rind et al., is considered best practice.

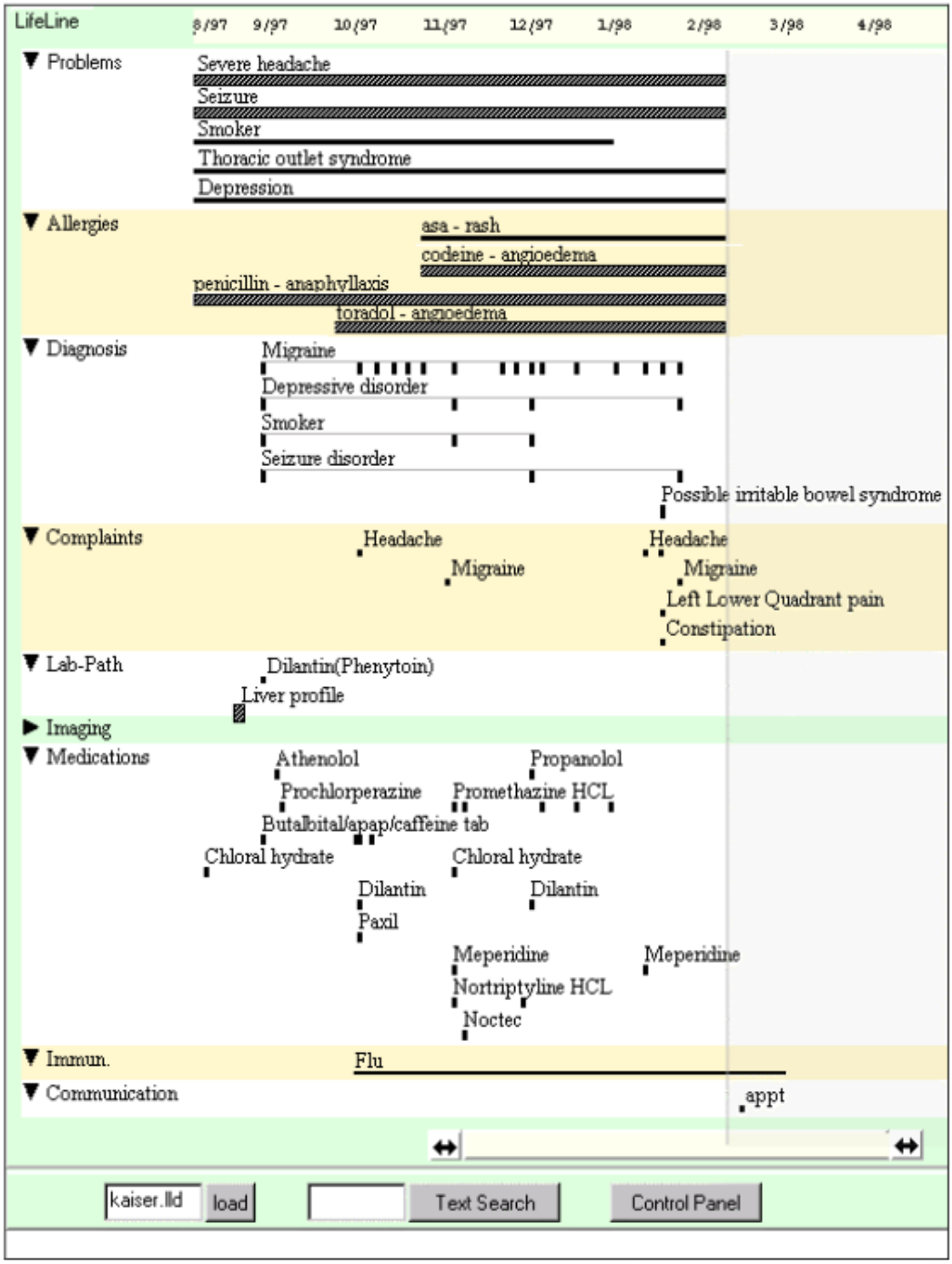


Figure 4.1: Lifelines system. Events are categorized and shown over time.

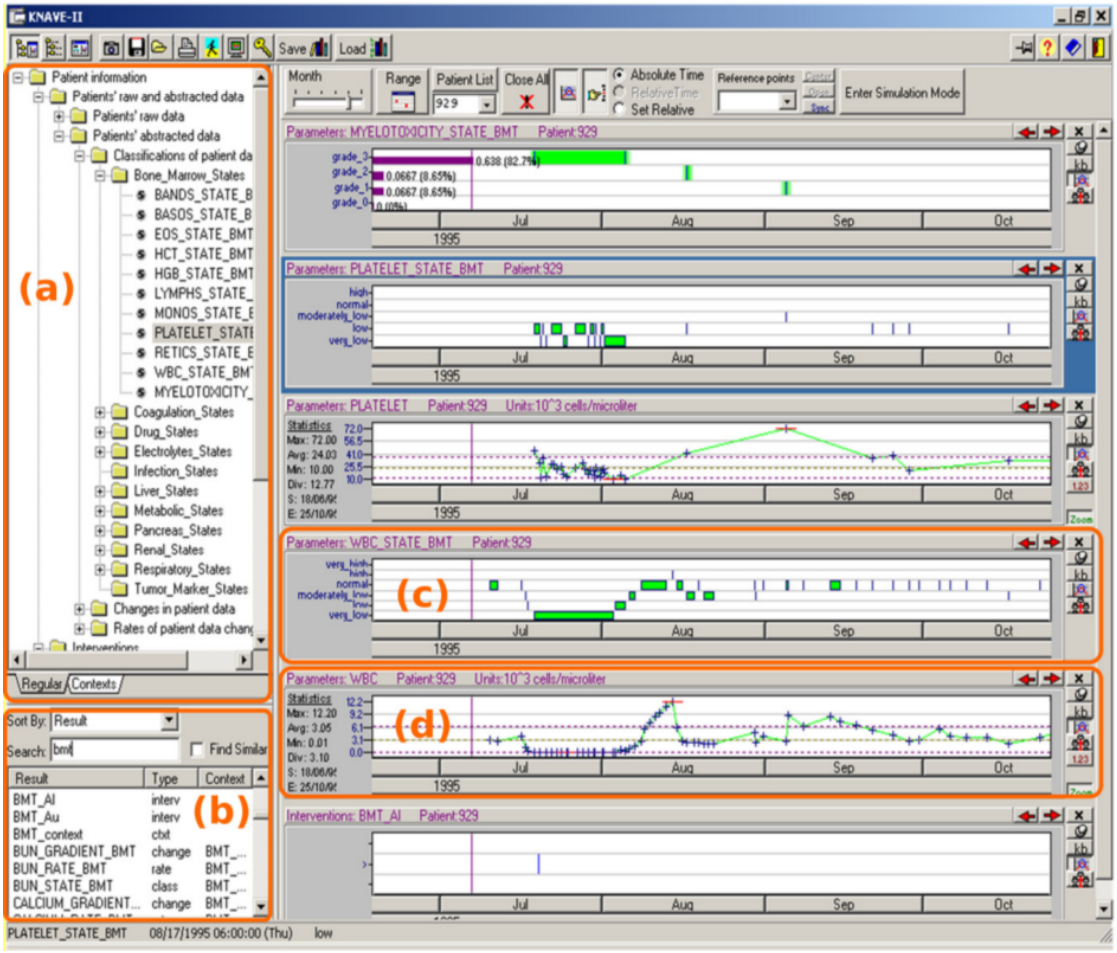


Figure 4.2: Knave II system by *Shahar et al.* [25]. (a) Domain ontology browser for selecting concepts and raw data for visualization. (b) Search panel for concepts. (c) Abstract data on ordinal time scale with durations. (d) Numerical data displayed with line plots. Image by *Rind et al.* [24].

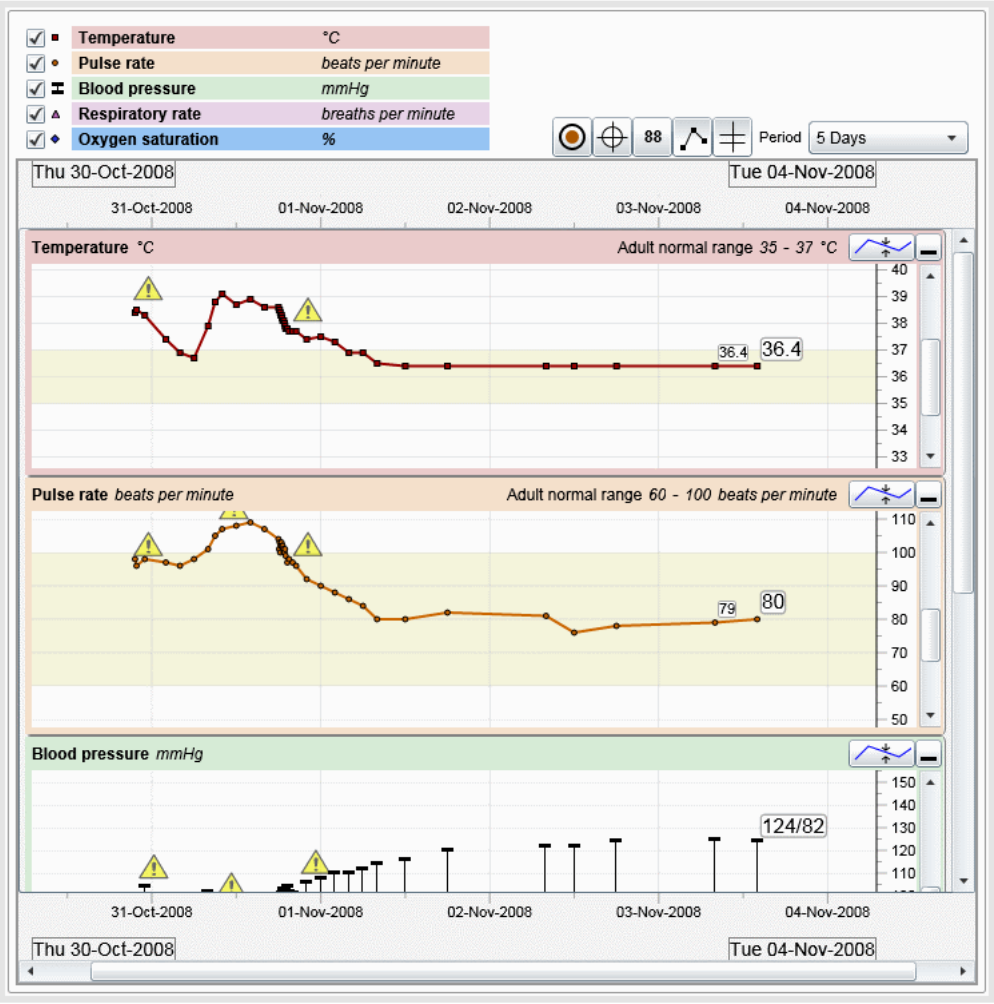


Figure 4.3: The Health Common Interface system by Microsoft.

Figure 4.4: Another part of the Health Common Interface by Microsoft.

4.2 MPS

First of all we look at the preceding work for this project by Paul van der Corput [7] that resulted in the *Three Table View*, shown in Fig. 4.5 that uses the "row relation glyph" (Fig. 4.6). The design was focussed on providing insight in the relation between the three main entities: doctors, patients and medicines. Each entity has a table for all its instances and relations between entities are shown via a glyph in front of a row.

With the focus on the three entities simultaneously, this system differs from the existing EHR systems that typically focus on the perspective of the patient. For our purposes, the perspective of the patient is the most important one, because we want to improve the clinical interaction and provide support for clinical research.

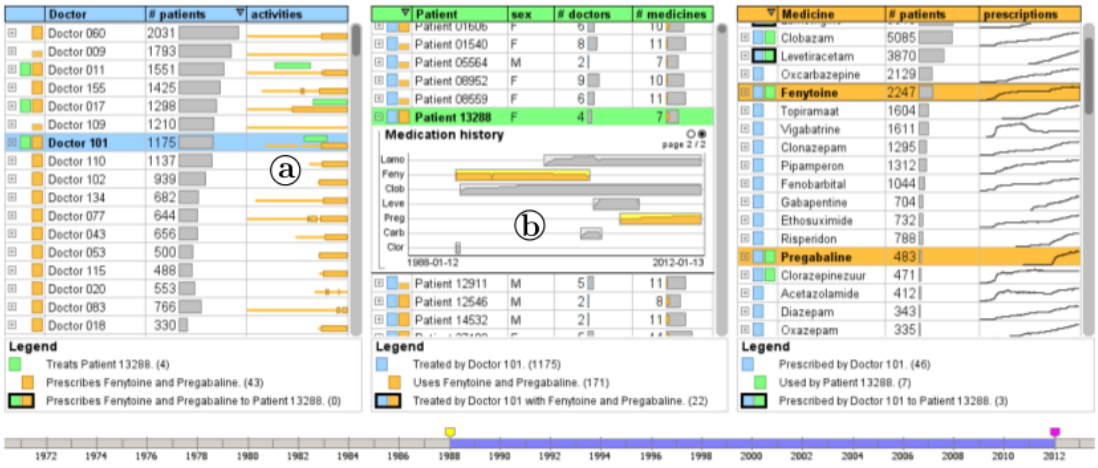


Figure 4.5: The three table view. On the bottom a time period is selected that filters the existing relationships over time. (a) Time of treatment by a doctor of the selected patient, while prescribing the selected medicines, is shown using Gantt charts. (b) The prescription history of the selected patient. (Paul van der Corput [7])

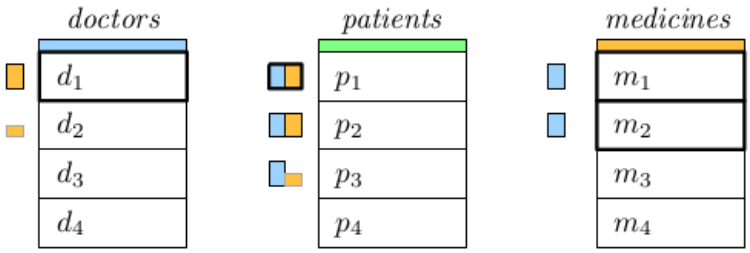


Figure 4.6: Row relation glyphs. Rows with a thick border are selected rows. The glyph color indicates the relation between doctors, patients and medicines. (d_1) A doctor who has prescribed all selected medicines. (d_2) A doctor who has prescribed one of the two selected medicines. (p_1) A patient who has been prescribed m_1 and m_2 by doctor d_1. (p_2) A patient who is treated by d_1 and has been prescribed m_1 and m_2, but not all by d_1. (p_3) A patient who is treated by d_1 and has been prescribed either m_1 or m_2. (m_1, m_2) Medicines that are prescribed by d_1. (Paul van der Corput [7])

Analogously to what Lifelines was for SPSs, the *Lifelines 2* system, designed by *Wang et al.* [32], is a classic example of an MPS. See Fig. 4.7 for an impression of the system. The *point* event sequences of multiple patients are shown stacked vertically on a timeline that can be panned. A period selector is present for setting the granularity of the timeline.

The filter panel has options for filtering and ranking of the records. Furthermore, it is possible to align sequences on event type and nth occurrence. Sequences can also be filtered by specifying a temporal sequence filter, of which an example is depicted in Fig. 4.8a. The sequence filter uses a form for expressing events linearly in time, starting from first specified and ending with the last specified. It is possible to explicitly specify the absence of events in a event sequence.

A filtered group can be saved and later be loaded in the comparison mode of Lifelines 2, shown in Fig. 4.8b. The distribution of variables of two groups are stacked vertically. Charts can be normalized and the time granularity is configurable. Besides manual specification of filters to define a cohort, a automated approach is possible, using a similarity measure. A good example of a system offering this features is called *Similan* [36].

The Lifelines 2 system has evolved into the *LifeFLow* system by *Wongsuphasawat et al.* [35], see Fig. 4.9. This systems has a main component showing a graphical summarized overview of all event sequences. In this case the sequences are aligned on the "arrival" event, indicated by a blue color. Thereby, the Lifelines 2 systems is integrated as a component for a more detailed visualization of the selected patients.

Another possibility to show data of multiple patients is to simply plot the data in one chart and make use of temporal abstraction if necessary. Consider the *VISITORS* system [16], shown in Fig. 4.10. It is basically the Knave II system evolved into an MPS system. Now there is a way to select patient cohorts using boolean expressions using forms and graphical widgets for specifying temporal constraints (more on this in Section 4.3). The actual data of a cohort is plotted over time using the typical charts, but the system summarizes or aggregates the data before showing it graphically.

An example of a system for visualizing a cohort of patients over time is *Outflow* [34], shown in Fig. 4.11a. A cohort of patients is repeatedly split, depending on the occurrence of an event. In this way the patients "flow" via pathways from left to right on the screen. The rectangular geometry of each splitting point reflects the size of the group via its height and the average duration of the split (the time between two events) via its width. Color is used for coding the different outcomes at a splitting point.

A more recent system is *CAVA* by *Zhang et al.* [38] that embeds the Outflow system as a component in its interactive process of cohort specification and exploration. Fig. 4.12 shows a basic use case of the system. A user starts with a cohort overview consisting of several visualizations. The system automatically generates some filters, but manual filters can also be defined. After selecting some filters, the user proceeds with the temporal pattern view, in which the pathway of patients of interest can be selected. Finally, detailed patient information is obtained using a tabular view.

Finally, we briefly mention two systems that show a snapshot of multivariate patient patterns. The first is the *Gravi++* system by *Hinum et al.* [11], see Fig. 4.13. On the screen black squares are drawn that represent aspects of patient data and pictograms represent patients. The higher the tendency of a patient regarding an aspect, the more it is attracted towards the associated square.

The second system system is called *TimeRider* [23]. Animated scatter plots with optional traces are used to visualize trends in patient cohorts. By varying the color shape and size of the marks, up to three more variables can be encoded in the visualization.

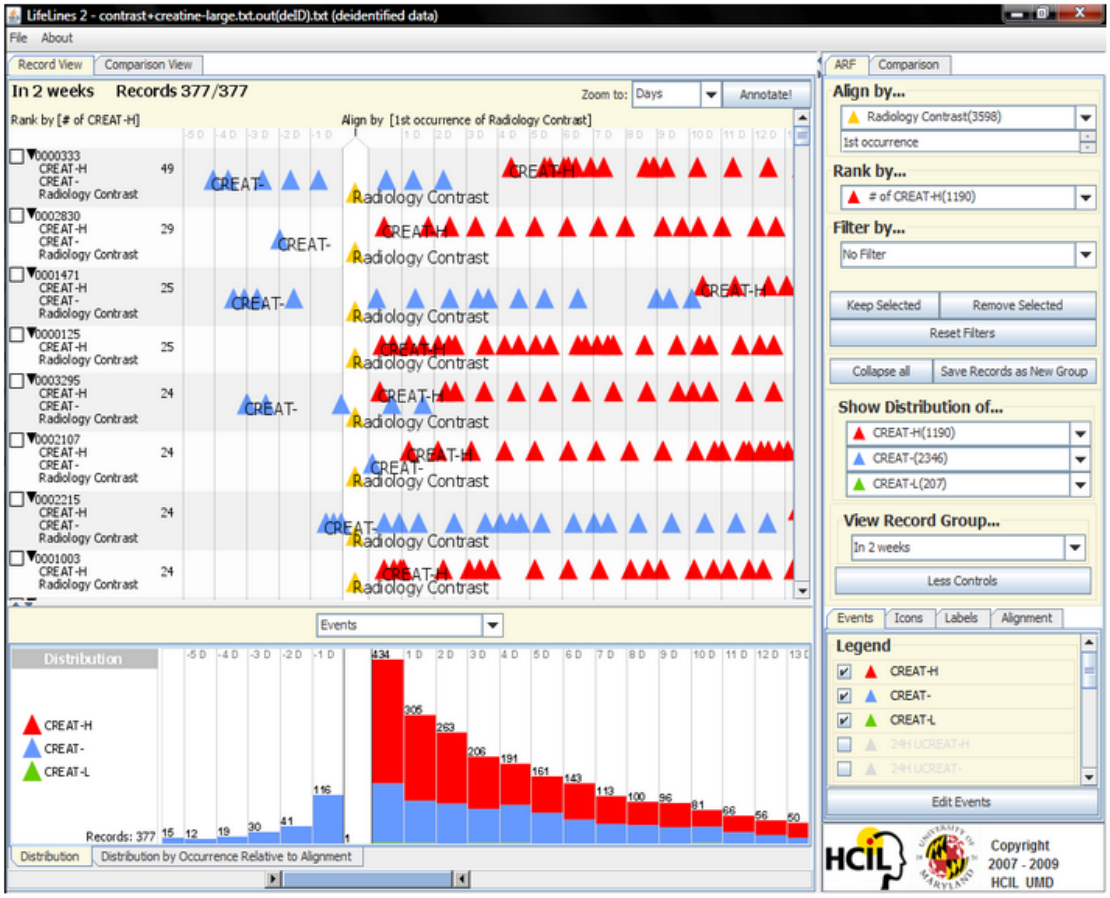
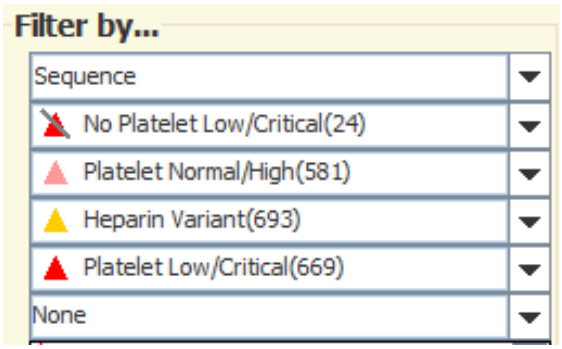
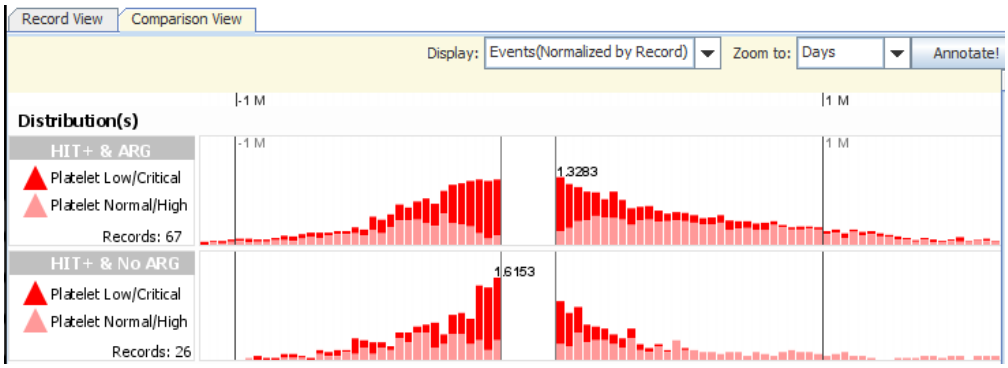


Figure 4.7: Lifelines 2 system. The top left component shows the point event sequences of a patients vertically stacked. These sequences are aligned using a sentinel event (yellow) aiding the user to compare the sequences. On the bottom distributions are shown over time for a selected event type. On the right is a filter panel to filter out a group of patients and specify the sentinel (*Wang et al.* [32]).



(a) Event sequence filter.



(b) Comparison of distribution of events for multiple groups.

Figure 4.8: Lifelines 2 sequence filtering and group comparison features (*Wang et al.* [32]).

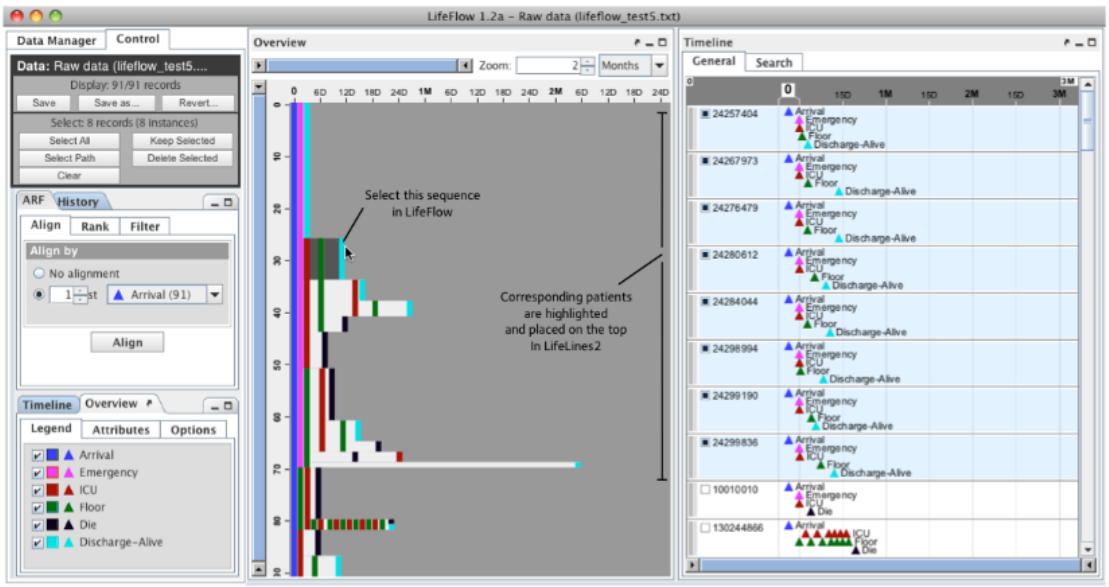


Figure 4.9: LifeFlow system. An graphical summary of all events sequence is shown in the center. The Lifelines 2 system is integrated for visualizing more details of the selected patients. There is support for using sentinel events (*Wongsuphasawat et al.* [35]).

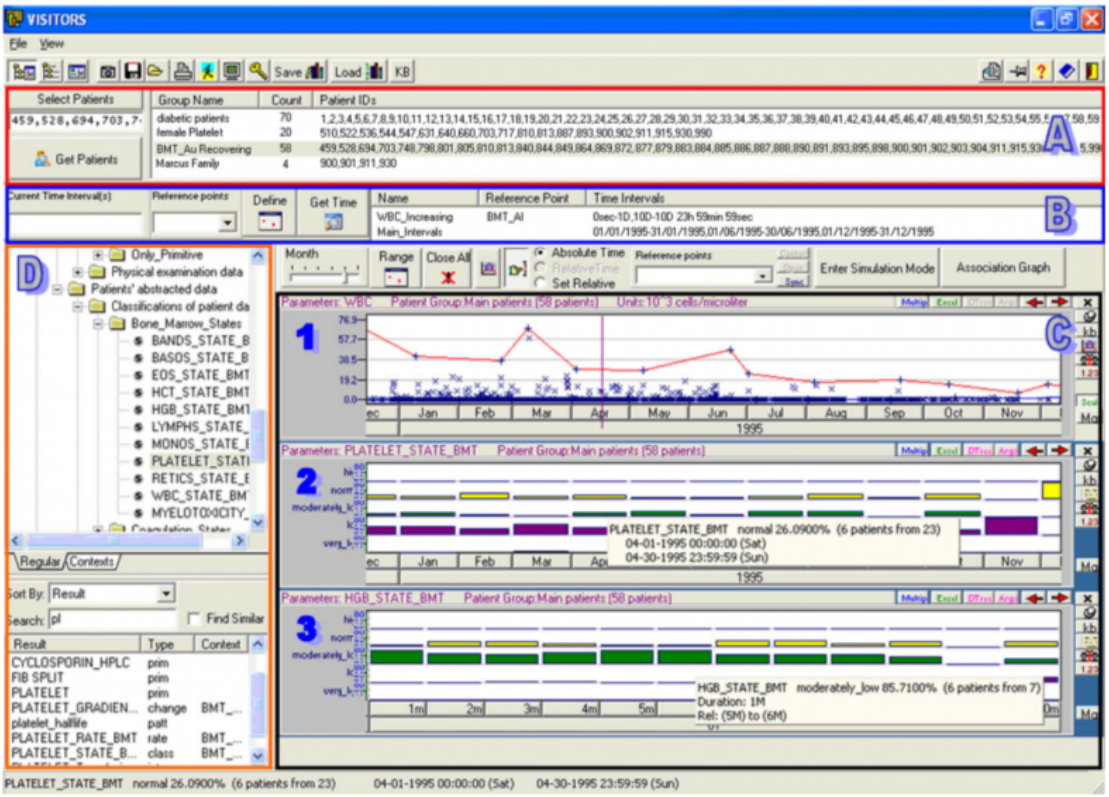
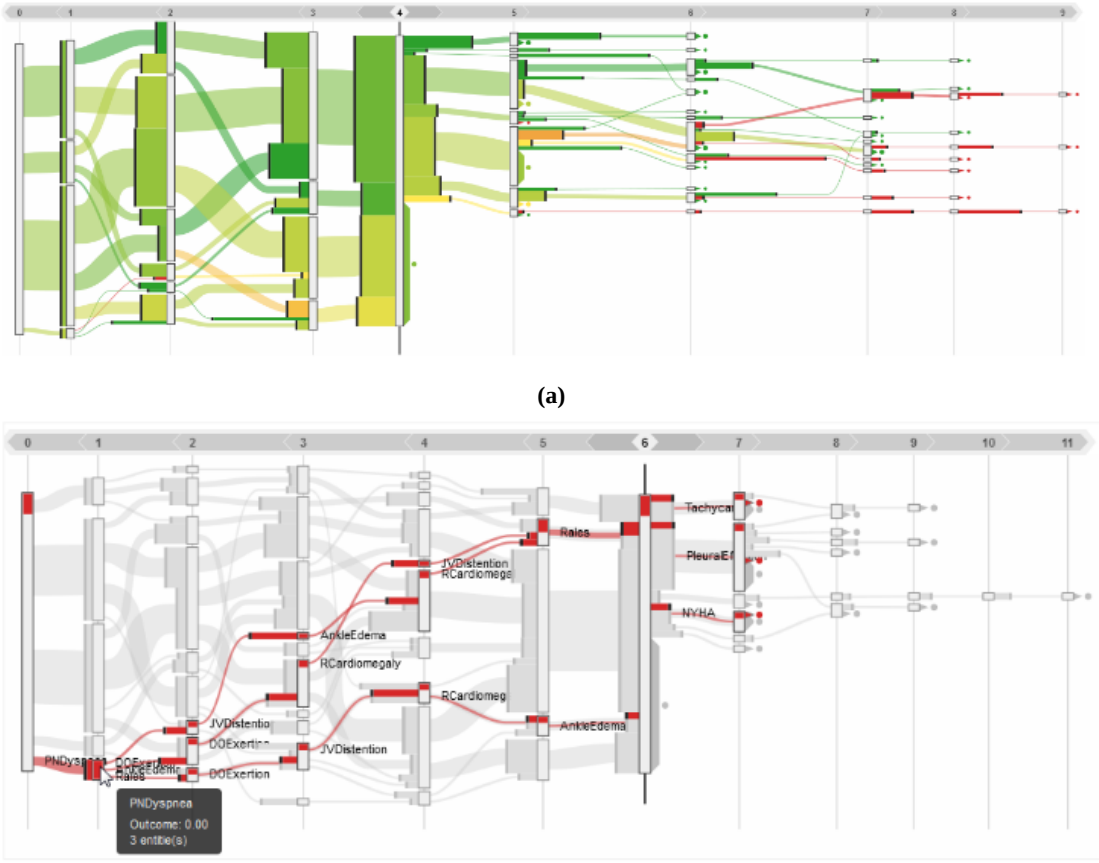


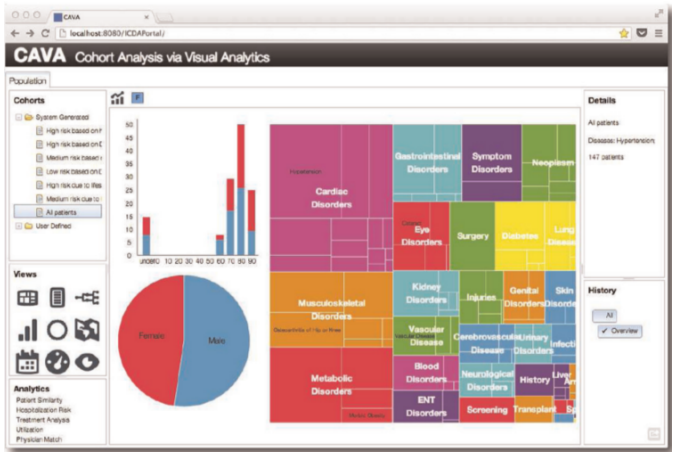
Figure 4.10: The VISITORS system [16]. (A) Patient cohorts. (B) List of time intervals. (C) Raw data and temporal abstraction of the current time interval. (D) The ontology browser including the abstractions (*Rind et al.* [24]).



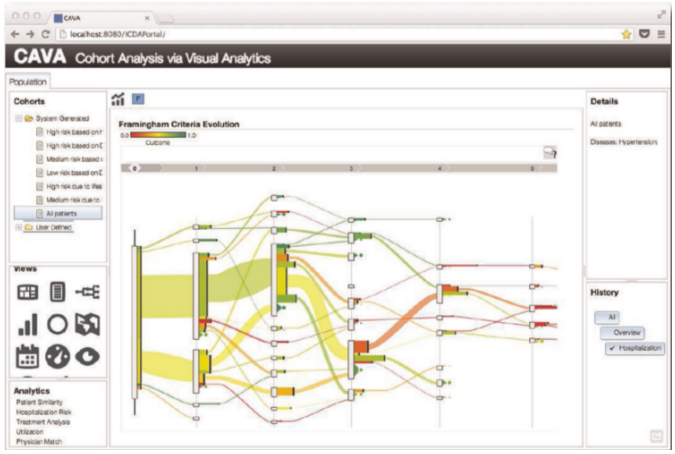
(a)

(b)

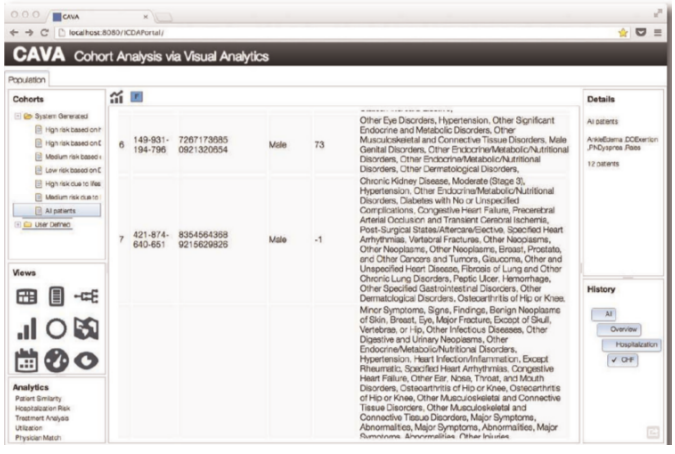
Figure 4.11: Visualization by the Outflow system. (a) From left to right, a cohort of patients is split repeatedly, depending on the event that occured. The height of a bar indicated the patient count and the width the average duration. (b) A brush is applied to highlight a selected path in the flow (*Wongsuphasawat et al.* [34]).



(a)



(b)



(c)

Figure 4.12: CAVA system. (a) The cohort overview. (b) After applying cohort filters, the users switches to a temporal pattern view to identify a high-risk patient pathway. (c) Finally the user switches to a tabular view for detailed patient information (*Zhang et al.* [38]).

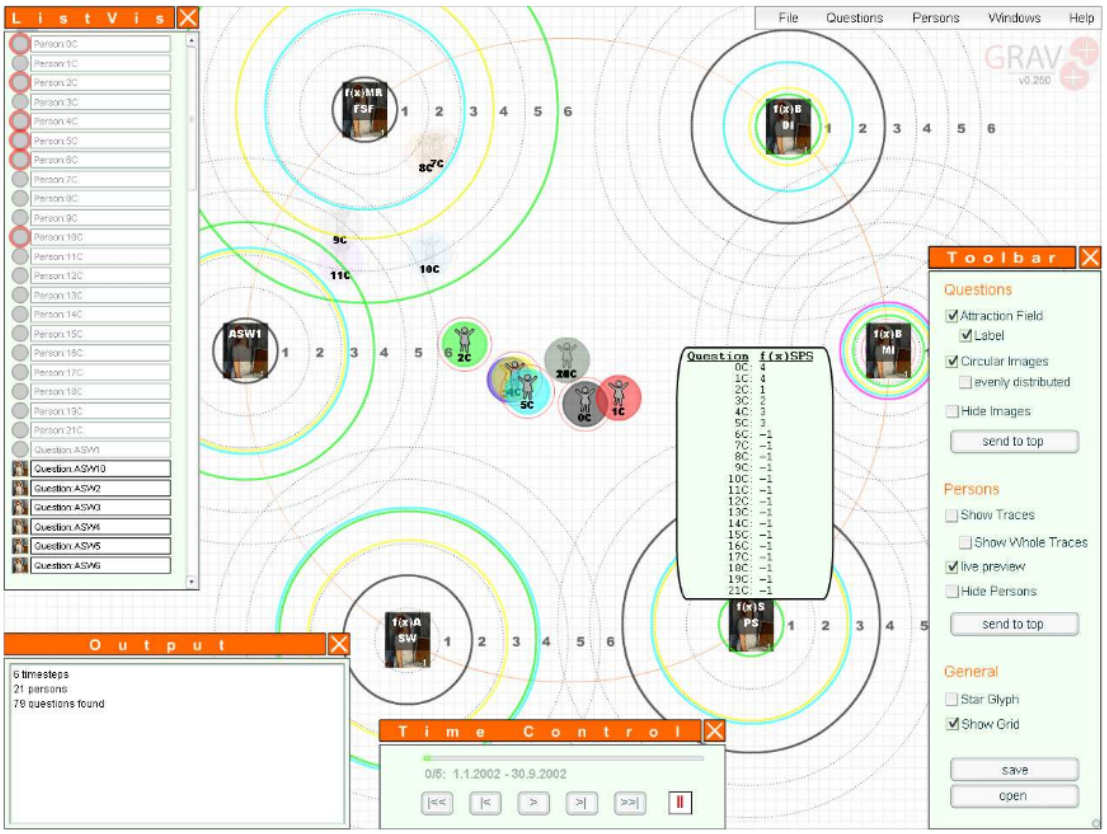


Figure 4.13: The Gravi++ system. A typical application case in which patients, represented by pictograms, are attracted from the center towards the black attractor squares (*Hinum et al.* [11]).

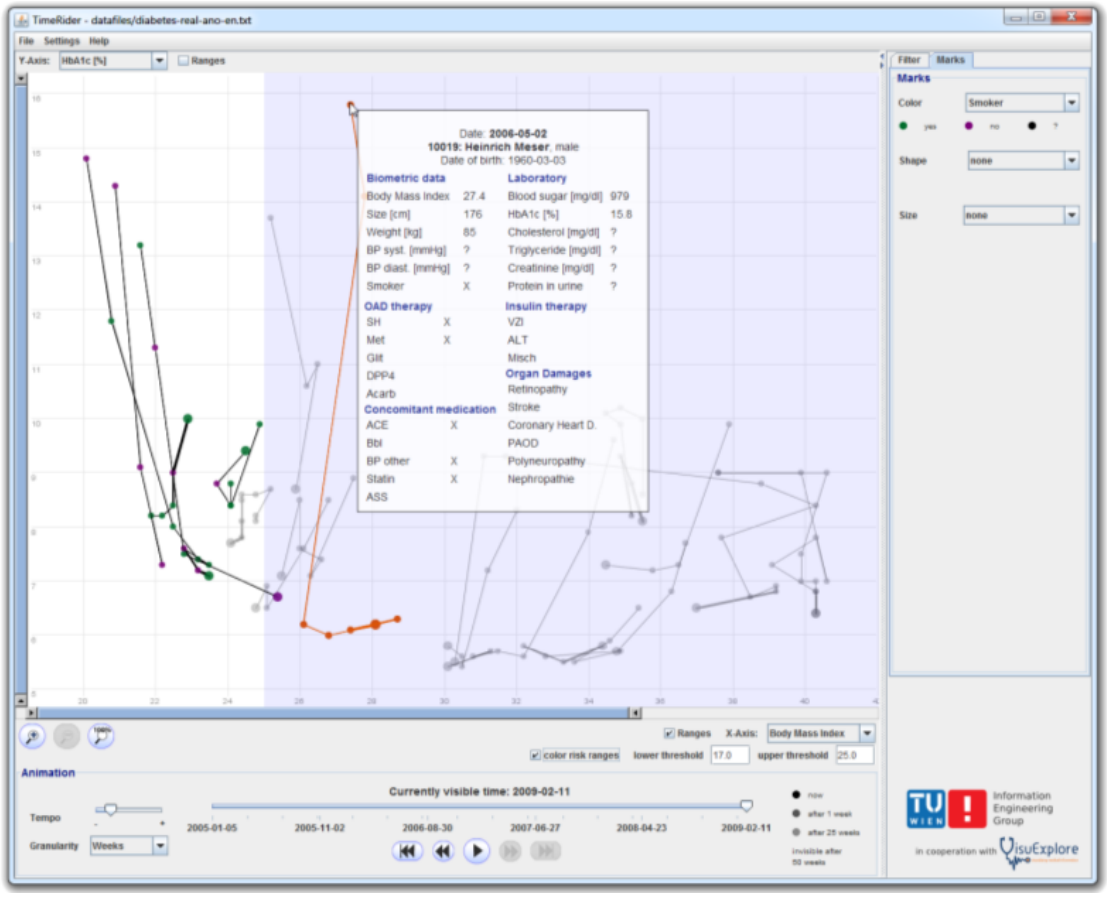


Figure 4.14: The TimeRider system. It employs an animated scatter plot. The green and purple colors, in this case, distinguish patients that are smoking from non-smoking patients. Hovering over a data point triggers an overlay showing the patient details (*Rind et al.* [23]).

4.3 Visual (temporal) querying

A standard user is generally not keen on learning a query language to express a filter. In reaction, forms and visual methods have been used to allow the user to work on a higher level of abstraction. The system is then responsible for translating the expression of the user to a lower level expression, like a query language.

The VISITORS system [16], discussed in Section 4.2, offers the user an interface for selecting patients by specifying temporal and value constraints. The temporal constraints are specified pairwise for two variables at a time, see Fig. 4.15.

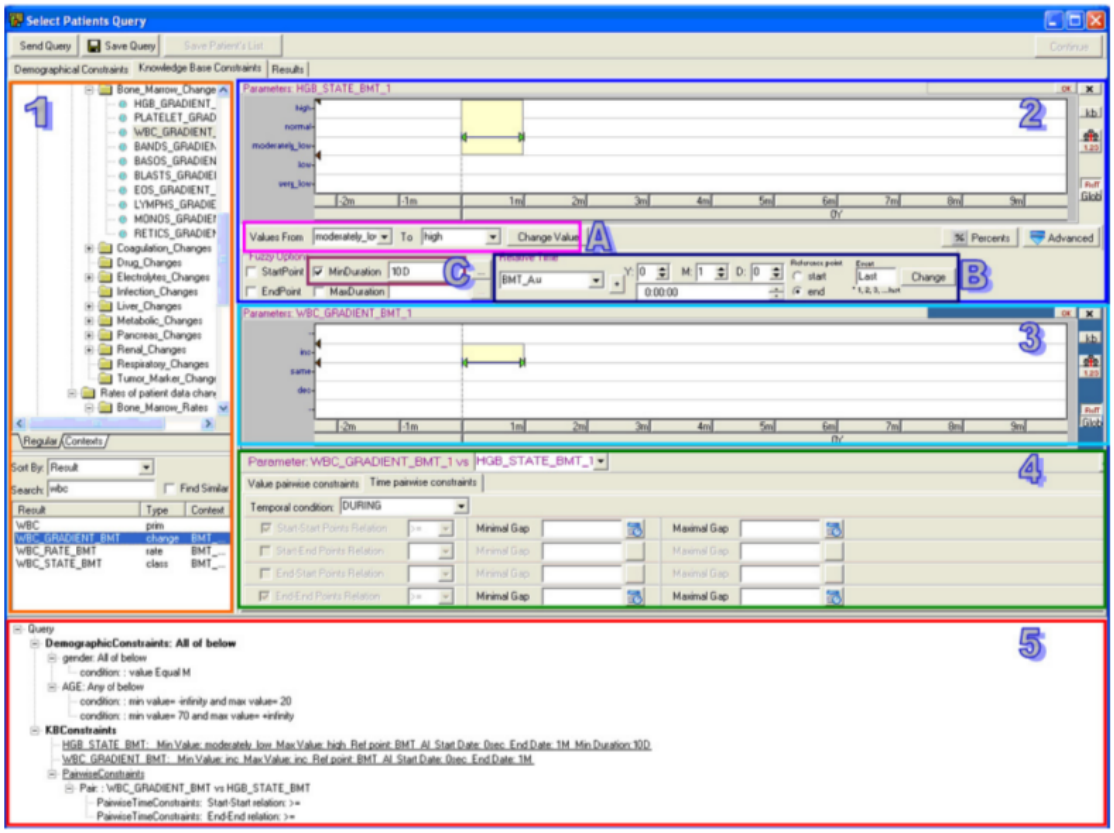


Figure 4.15: The VISITORS UI for patient cohort selection [16]. (1) The domain ontology browser. (2-3) There are two graphical widgets that allow the user to draw a period of time using lines. (A-C) Instead of the graphical widgets, a user can use these forms. (4) This panel specifies the pairwise temporal constraints, like minimal and maximal gaps between the variables. (5) The generated query. Image by Rind et al. [24].

EventFlow [19] allows the user to define slightly more complicated queries, shown in Fig. 4.16. In the graphical query, lines segment indicate periods of events of a specific type. The type is related to the color of the line segment. Gaps between different line segments are used to express a (maximum) period of time between different types of events.

A different approach is taken for the design of the *QueryMarvel* system, see Fig. 4.17. In this case the user specifies an event sequence by adding little boxes, which represent events or time periods, using a comic strip metaphor.

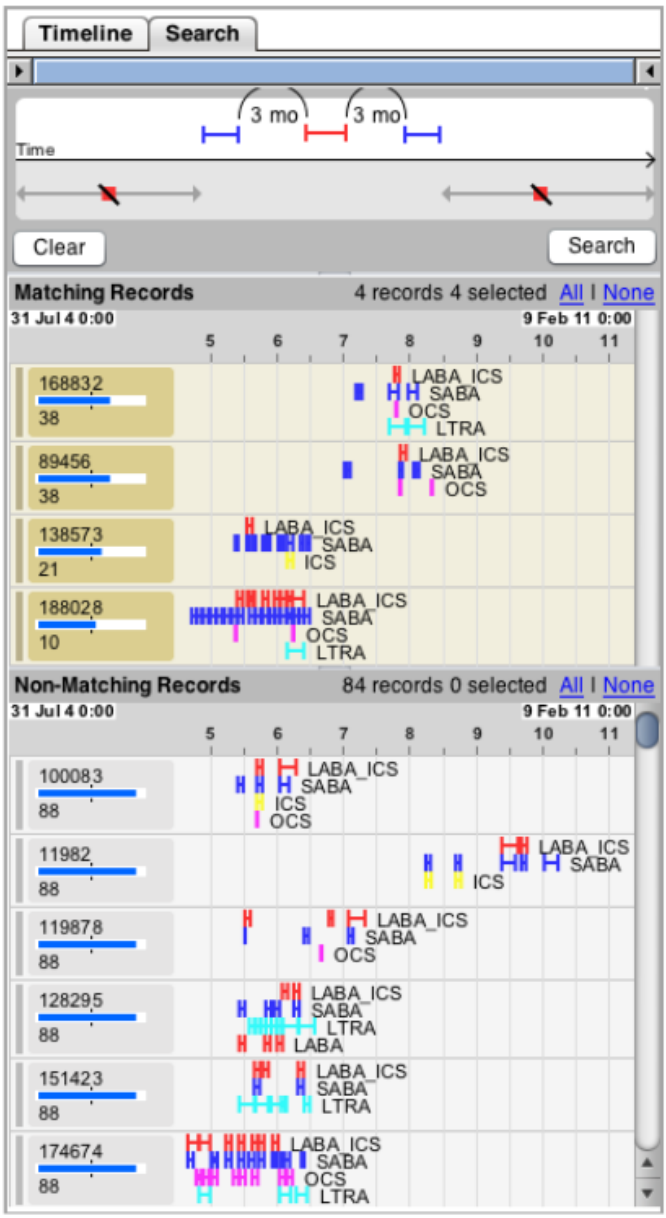
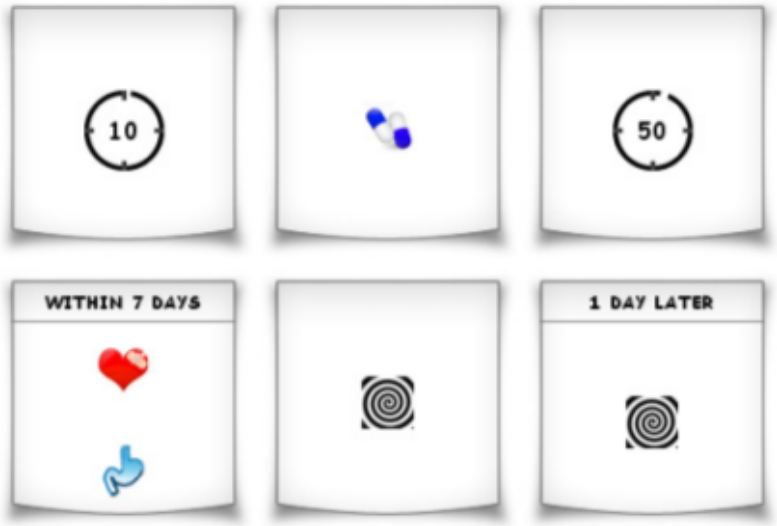


Figure 4.16: The EventFlow system. The graphical query is specified at the top using line segments and time period indicators. In this case the cases where the blue events (SABA" medication) occur within three months of the red events ("LABA_ICS") are in the results, which are shown on the bottom (*Monroe et al.* [19]).

(a) QueryMarvel UI.



(b) Example query.

Figure 4.17: The QueryMarvel system. (a) The following UI components are present in the system: 1. The event picker; 2. Undo / redo, time and negation operations; 3. The content panel; and; 4. The info panel. (b) An example query that specifies "patients who take medicine X between 10 th and 50 th day, and then within 7 days have chest pain and stomach pain at the same time, followed by feeling nausea for two consecutive days" (*Jin and Szekely* [14]).

4.4 Faceted search

Many systems offer methods to filter an event sequence or to specify a cohort of patients that is of interest. Usually multiple forms to filter characteristics in combination with graphical widgets for specifying temporal constraints are employed. The general principle behind these methods is often what is known as the *faceted search*. A user adds constraints one by one using a form or widget form the constraints that filter the data.

The filtering process using a faceted search basically works a follows [2]. A filter, whether it is of a temporal nature or not, can be expressed as the union of its constituent (boolean) conditions or constraints. Each filter is translated into a query that selects a subset of the data based upon the conditions. Next, the intersection of all these resulting subsets is taken to obtain the final set as the result.

Consider the example in Fig. 4.18. The facets of the patient data that can be filtered on are: a temporal aspect of the data (like for instance registration date), the IQ class, gender and events. A user sets conditions on the facets and thereby specifies a group of patients. Note that all conditions hold *simultaneously* for the patients in the resulting group. The group can be updated interactively by altering the conditions on the facets. In practice, a group of patients can be ranked in a table or its (aggregated) properties can be visualized.

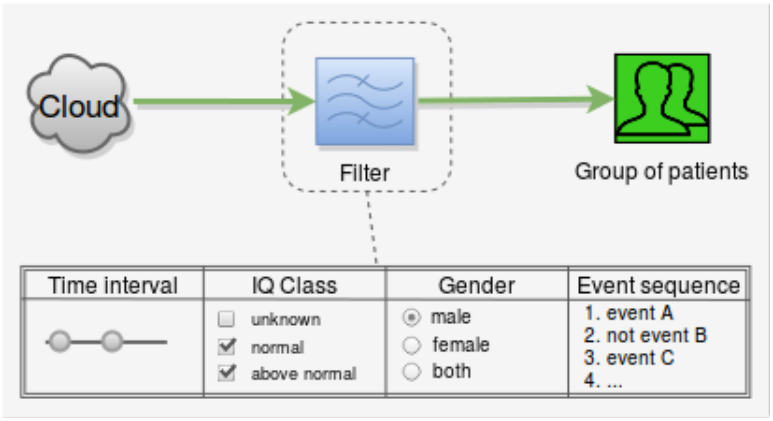


Figure 4.18: Facated search example. The patient in the cloud are filtered after setting the conditions in the filter for the relevant facets of the data. In this example, the filter is configured using a time interval selector, an IQ class selector, a gender selector and event sequence specifier. The conditions can be altered interactively and the resulting group of patients is updated after each change.

4.5 Comparison

In their survey paper Rind et al. provided a comparison between the surveyed systems on their features and using the user intent model (Section 2.5), see Fig. 4.19 and Fig. 4.20 respectively. Note that, as stated before, not all systems in the comparison have been described here, but we leave the comparison as it is for a broader context.
In addition to these comparisons, we estimated the adoption of the analytic focus strategies (Section 2.6) by the MPS systems that have been described in this chapter. The result is shown in Table 4.1.

		categorical data	*numerical data*	*no. of variables per screen*	*one patient*	*multiple patients*
single EHR	LifeLines	●	○	≈ 25	●	
	MIVA	○	●	≈ 5	●	
	WBIVS	●	●	10	●	
	Midgaard	●	●	≈ 15	●	
	VisuExplore	●	●	≈ 10	●	
	VIE-VISU	●	●	15	●	
EHR collection	Lifelines2	●		≈ 10		●
	Similan	●		≈ 10		●
	PatternFinder	●	○	3		●
	VISITORS	●	●	≈ 5	●	●
	Caregiver	●	●	1–6		●
	IPBC	○	●	≈ 3	●	●
	Gravi++	●	○	≈ 6		●
	TimeRider	○	●	2–5		●

●: Full support, ○: partial support, " ": no support.
The number of variables per screen is an estimate based on examples in the original publication.

Figure 4.19: Comparison of systems based characteristics [24].

	3TV	LifeLines 2	LifeFlow	VISITORS	OutFlow	CAVA	EventFlow
Goal-driven record extraction	●	●	●	●	○	●	
Goal-driven event extraction	●	●	●	●	●	●	●
Temporal windowing	●	●	●	●			
Random sampling of records							
Temporal folding							
Grouping event categories (aggregation)				●			●
Selecting sentinel events in a stream	n.a.	●	●	●	○		○
Converting multiple point events into a single interval event	n.a.		●	●			○
Converting multiple interval events into a longer interval event	●			○			
Identify hidden complex events	n.a.			●			○

Table 4.1: Estimation of support for analytic focus strategies by the MPS systems, based upon the descriptions of the systems. ●: Full support, ○: Partial or indirect support, " ": no support, n.a.: not applicable.

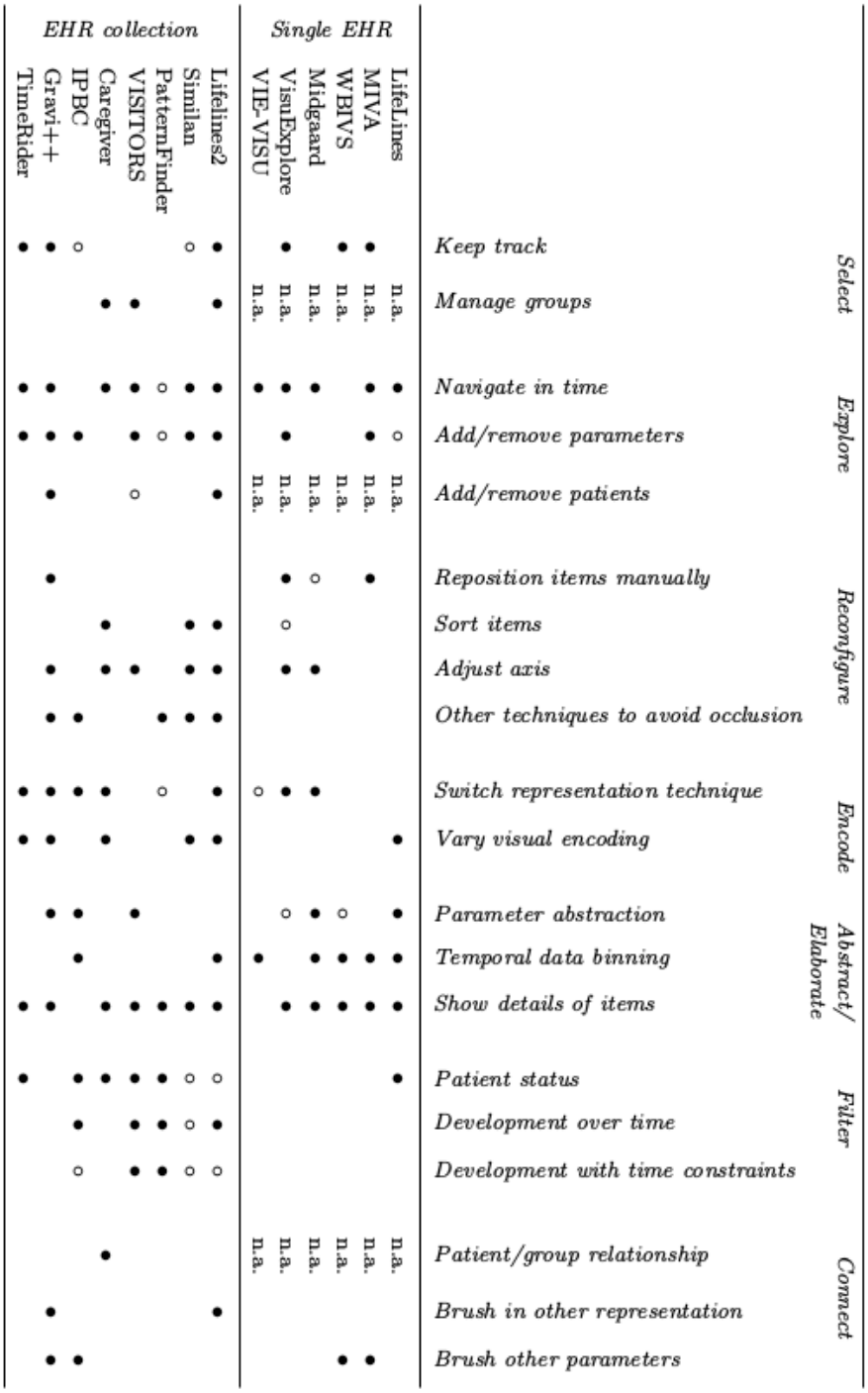


Figure 4.20: Comparison of system based on the user intent model [24].

4.6 Summary

In this chapter we have seen examples of EHR systems of both the SPS and MPS type. A typical SPS shows the status of patient over time, typically, in the form of a navigable timeline on which numerical and categorical data is shown, using lines plots and bars, grouped using the domain ontology. Additional graphical options for the chart are configurable, like the display of value labels. The details of an item are commonly shown in a dedicated panel. Often some basic filter or search features are offered.

An MPS regularly shows aggregates of a cohort of patients and / or a means to compare event sequences of patient within the cohort. Some systems allow for comparison of cohorts, usually via aggregates or abstraction of event sequences. The specification of a cohort is done through the formulation of (temporal) constraints using forms and/or graphical widgets.

The support for the visual expression of a query, especially the ones involving temporal constraints, is evolving. A set of constraints is commonly specified using a form of faceted search and what started with the entry of simple numerical values for defining a temporal constraints, has evolved into graphical drawing of event sequences with associated attributes.

In general we have recognized a trend that a SPS evolves into an MPS. For instance, we have seen the evolution of LifeLines $\rightarrow$ LifeLines 2 $\rightarrow$ LifeFlow and KNAVE $\rightarrow$ KNAVE II $\rightarrow$ VISITORS. The LifeLines approach towards visualizing event sequences is very popular and is adopted on a regular basis.

The basic methods for visualizing the EHR of a single patient are well developed. The same holds for the visual methods for comparing the EHR of groups of patients, including event sequences. The focus is now on adding more complex and interactive methods. Another aspect is the filtering of a cohort of patients that can be used for comparison with other cohorts. To this end, many systems employ a faceted search type of filter mechanism and an open question is whether this can be improved.

Chapter 5

Graphical Electronic Health Record

In this chapter we introduce the system we designed for meeting the research objectives on the SPS side. The main part of the solution is assembled by incorporating concepts already found in the literature, as discussed in Chapter 4. Still we have to translate these concepts to fit the specific case of Kempenhaeghe, detailed in Chapter 3.

The doctors at Kempenhaeghe need quick insight into a patients history, including all relevant events. Currently. there is limited graphical support for this and doctors have to scroll their way through many panels that show the data mainly in tabular format. This limits the overview a doctor has and it is also time consuming. Moreover, it is usually the overview that doctors need as a basis for their reasoning behind what to do next for the patient.

Also, the clinical interaction is far from optimal, due to the amount of time a doctor needs to spend on sorting out data first, before answering a question of the patient. For instance, finding out when an specific event had taken place in the past by scrolling through the notes one by one in a table is an inefficient way of working. Adding a full-text search mechanism will improve the situation for searching events, but once an event is found, the event still needs to be placed into context, and details have to be shown for further reasoning.

Thus, there is a need for searching events and providing context and details quickly for selected events. Conversely, an overview aids the analysis and the correlation of measurements and / or events, which can result in new questions and searches. Additionally, if the system would provide a clear overview, both the doctor and patient could interact with it during the clinical interaction simultaneously, enabling better insight for the patient as well.

The benefit of showing the patient status over time using small multiples of glyphs is that the changing factors in a complex multivariate dataset are rapidly conveyed. The drawback is that for an EHR containing data spanning a significant time interval, many glyphs are needed. Scaling issues and placement issues arise at the cost of the overview. Combining glyphs into fewer glyphs leads to less detailed perception of the data or even to distorted perception of the temporal component of the data. For this reason, we adopt the timeline approach.

The system presented in this chapter, dubbed the *Graphical Electronic Health Record (GEHR)*, addresses these issues. An image of the GEHR is shown in Fig. 5.1, in which the parts are tagged. In the section below, all parts of the systems are discussed in more detail. Most problems are solved using methods

found in literature, but we improve screen space usage with a more flexible UI, combine related variables into one chart and preserve the temporal distribution of events as much as possible when using temporal abstractions.

5.1 Basic patient information bar

The patient information bar shows the patients':

1. code,
2. gender,
3. age,
4. the main treatment location, and
5. the minimum, maximum and average weight.

Given the data, these fields of information were considered useful for this prototype. For matters of privacy, the date of birth of a patient was replaced with the age, determined at the moment the data was handed over.

5.2 Timeline and interval selector

The timeline is the main component on which all events are drawn, shown in Fig. 5.2. At any given moment the timeline spans a specific interval of time, ranging from a minimum span of two weeks to a maximum span that covers all events in the timeline. Changing the interval effectively zooms the timeline. The user can select fixed intervals of multiple years or specify an interval manually. These intervals are specified using the interval selector. Arrows aid the user with shifting the time interval to the left or right. If the user wants to shift the interval only slightly, the timeline can be dragged to the left or right.

5.3 Ontology based selector panel

Being inspired by the VISITORS [16] system, we have included a vertical panel for toggling and selecting items, based on the ontology (Fig. 5.3). The user is enabled to toggle the medication that is shown in the medication portlet on the timeline. This saves screen space for the more relevant medication at that moment. The medication that has been prescribed as "rescue medication" has no information about associated prescriptions periods in the database and is therefore disabled by default.

Furthermore, the seizures, weights and reports are sorted on date and can be selected. If selected, the timeline is panned automatically, using the currently selected time span, such that the date associated with the item is in the center of the view. This puts the item itself into view, which is highlighted. On advice of the domain expert, the seizures are sorted in a nested way, first on year, then month and finally day. Lastly, the panel is horizontally and vertically resizeable. The main motivations for these selectable items are that it is something quicker to find a specific item in a list for a given date, and, that the doctors are already familiar to work with data in tabular format (hence a smoother transition to the new system).

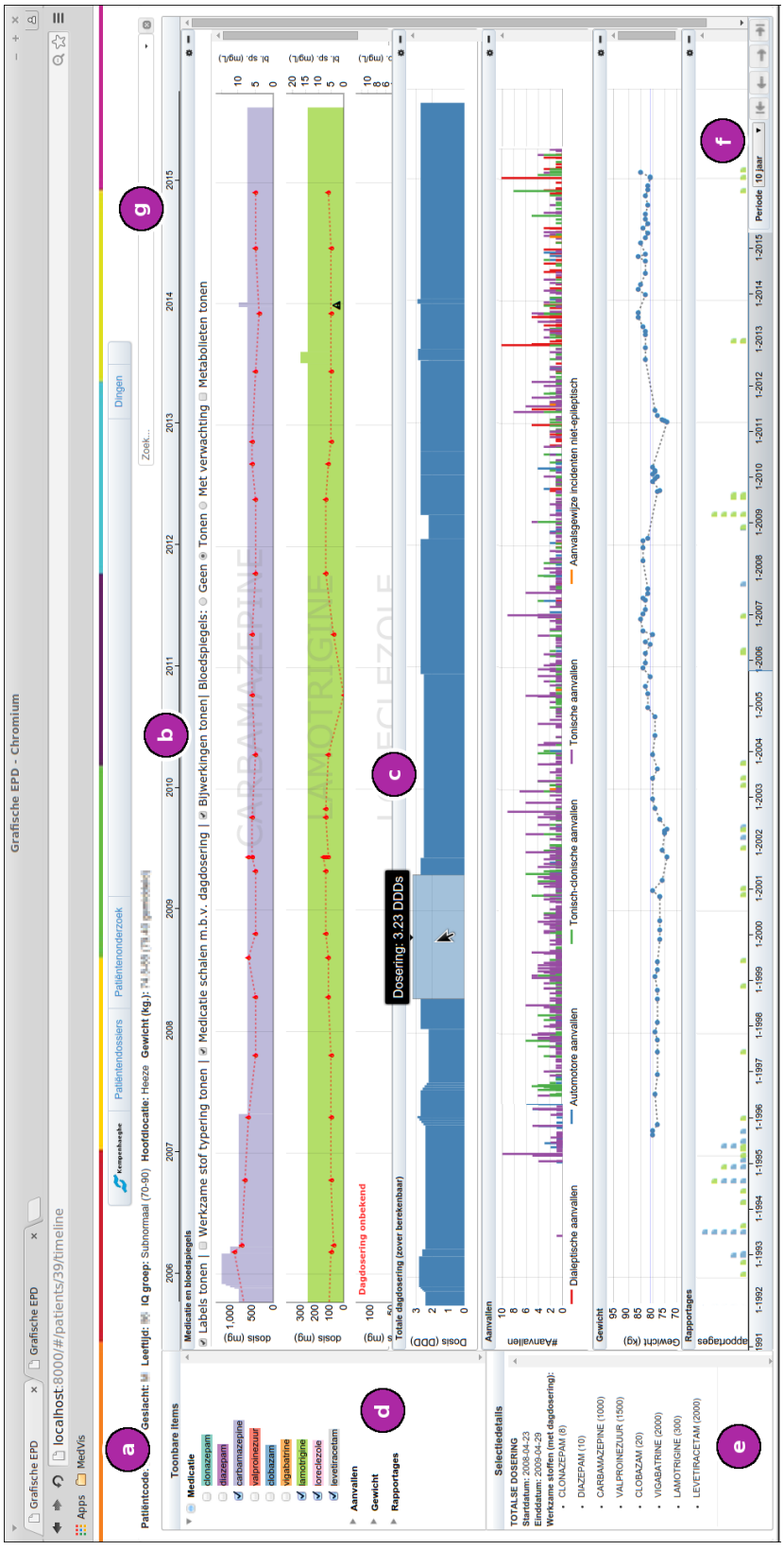


Figure 5.1: Overview of the Graphical Electronic Health Record (GEHR). (a) Basic patient information bar, like code and gender. (b) The timeline. (c) A portlet (window) on the timeline. (d) The ontology selector. (e) The details panel for a selected item. (f) Time interval selector. (g) Keyword based search bar.

5.4 Portlets and details panel

One of the distinguishing factors for EHR systems is the number of variables that fit on the screen (Section 2.4). Like the VISITORS system and the HCI by Microsoft we adopt the usage of smaller windows (portlets) for the charts, but we design them more flexible.

A portlet is a resizable and movable window, meaning that the vertical order can be changed by dragging the portlet to a new location. A portlet has the minimal height of one chart, but it can be resized vertically to show more. There are two buttons on the top right of a portlet. The first hides / shows the options bar and the second one collapses the portlet to its title bar. These options feature flexibility to the user to make optimal use of the vertical screen space.

Following consultation with the domain expert, we have added separate portlets for:

- the medication over time, including blood compounds and adverse effects;
- the sum of the total dosage over time;
- the seizure distribution over time;
- the weights measurements over time; and
- the reports filed over time.

All portlets are detailed below.

5.4.1 Medication

One of the most important features our system has to support is obviously the visualization of medication over time. The domain expert noted that a doctor is not primarily concerned about the actual medicine prescribed, but instead with the active substance within the medicines. Because of the structure of our data (Section 3.2) we have to merge all prescriptions for a particular substance over time before visualization, as shown in Fig. 5.4.

Merging prescriptions is normally done by translating a dose using the *Defined Daily Dose (DDD)* associated with an active substance, which allows for comparing and summing of dosages with different units. The DDD is a measure that is determined by the World Health Organization and indicates the average dose, per day, for a regular adult. Using the DDD we can normalize a dose for comparison and the addition of multiple dosages. Since doctors in their daily work are used to working with dosages in milligrams, we show the dosage in milligrams and the merging of the prescriptions for the same medicine is done in milligrams.

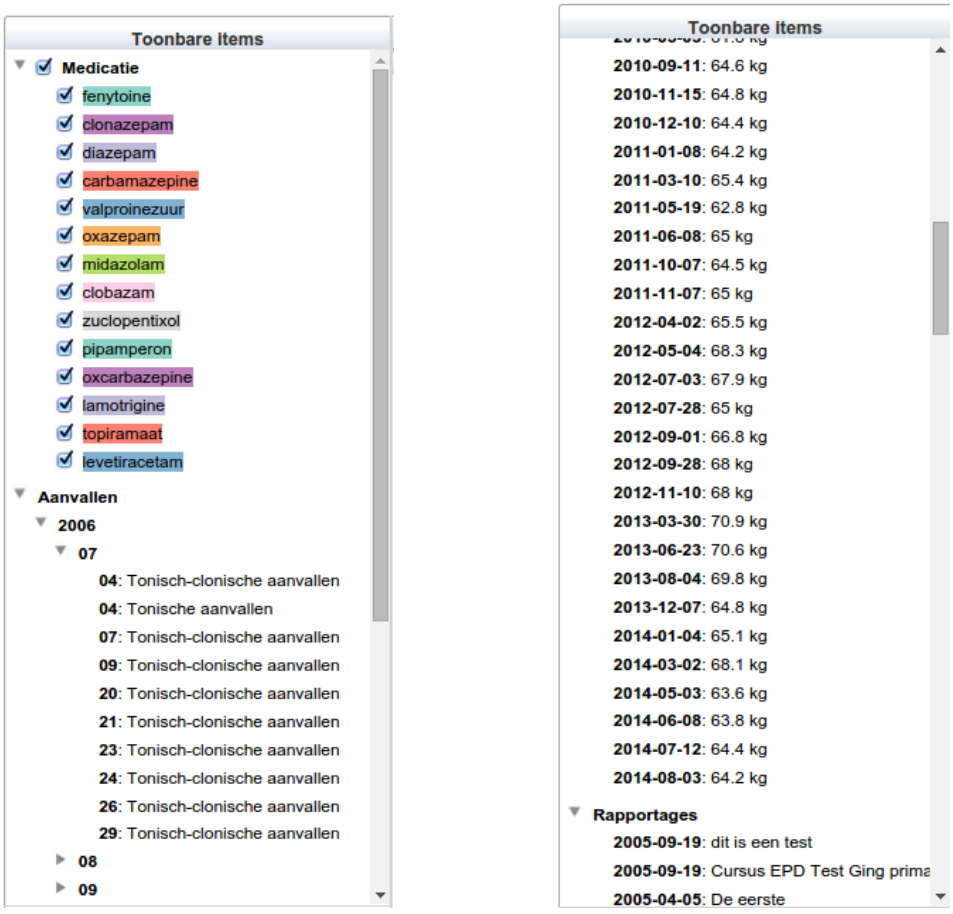
In Fig. 5.5a an example of a medication portlet is shown. The axis on the left shows the dosage in milligrams. The color used for drawing the medication corresponds to the background color of the medication items in the ontology selector panel (5.3a) and are selected from a set of colors released by *ColorBrewer* [10].

With the vertical axis unit in milligrams, we can still put to use the DDD. We have invented an option for the medication portlet that allows the doctor to scale the vertical axis of the medication. Normally, the maximum value of the vertical axis is equal to the maximum value of the dose. This stretches the chart vertically as much as possible. Using the DDD we can rescale the vertical axes with a factor, such that the maximum vertical stretch of a chart for a substance, in comparison with charts for the other substances, reflects the relative load (a measure for the impact on the patient's body).

(a)

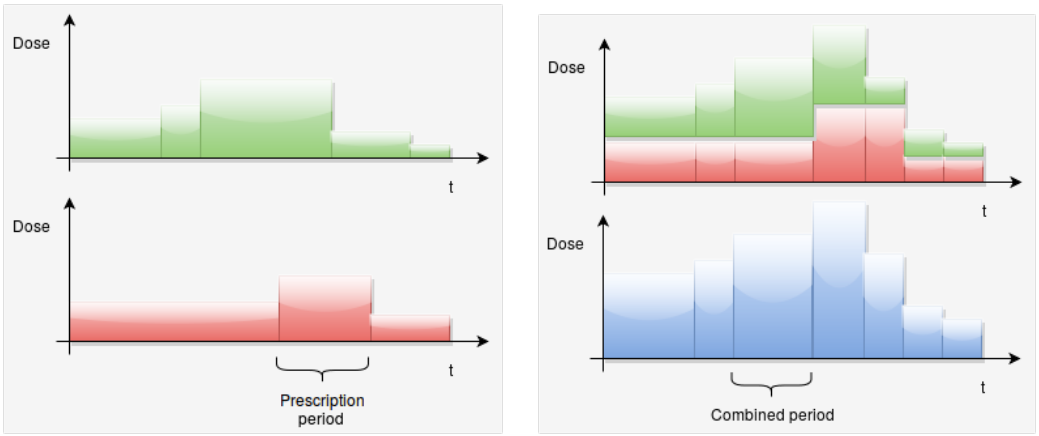
(b)

Figure 5.2: Timeline navigation. (a) Draggable timeline bar. (b) Period selector.



(a) **(b)**

Figure 5.3: The ontology based selector panel. (a) Medication toggles and selectable items for seizures, sorted on year, month and day. (b) Selectable items for weights and reports.



(a) **(b)**

Figure 5.4: Merging of prescriptions. (a) Two prescriptions. (b) The prescriptions merged into one. The original prescription periods lead to new artificial prescription periods.

To calculate the stretch, we take the maximum value of the dose for each substance (after merging all prescriptions for that substance) and divide that by the DDD, obtaining a number representing the maximum load of the substance on the patient. Then we take the maximum of all those load numbers and use that to normalize all load numbers. The resulting factors are the scaling factor for vertical axes of the charts, for which the units remains in milligrams. This scaling changes the maximum value of the vertical axes according to the scaling factor.

Fig. 5.5b illustrates the option of scaling using the DDD. The charts that show a relatively low dosage (load in terms of DDD) will appear more compressed vertically than those with a relatively higher dosage. Suppose that the maximum dosage over all substances is 3 DDD, then the chart for another substance with maximum dose of 2 will have the vertical axis scaled by a factor of 2/3. That way a doctor gets a quick feel for relative impact of a prescription on the total dosage. If the DDD is unknown for a substance, its maximum load is assumed to be 1 in the scaling process and a warning is shown.

In the figure the dashed lines indicate the level of a dosage of 2 DDD for that substance. A prescription of a substance for a patient with a dosage above the 2 DDD level should be not too common, meaning that a doctor proceeds with caution if that level is reached.

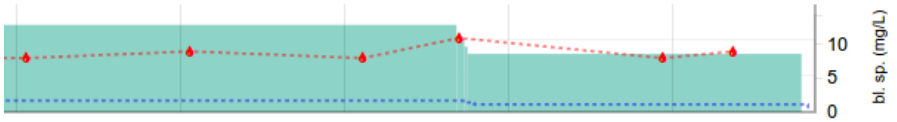
A second use of the DDD is to sum all loads to calculate to total load on the patient for all substances together. This is calculated and plotted over time for the chart in the dosage portlet, shown in Fig. 5.5c. Only substances with known DDD are included in the load calculation. It enables the doctor to reason whether the load is acceptable. Since the patient can now easily see his / her own history and dosages of medicines, he is able to ask more direct questions to the doctor about the proposed treatment procedures. This offers a basis for a richer form of clinical interaction.

Because the blood compounds measurements and the reported adverse effects are registered in association with a medicine and in turn with a substance, these are drawn over the medication chart (see Fig. 5.6a). Little blood drips show the measurements of the blood compound for the substance, a red line connects these and indicates the variation over time. A doctor may turn on the option to show the linear extrapolation of the first measured value of the blood compound in combination with the dosage of the

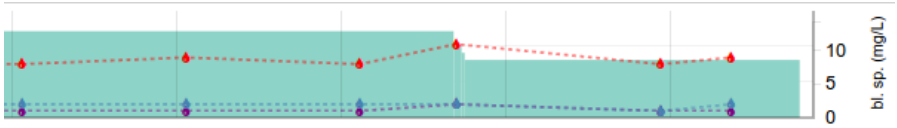
Figure 5.5: Portlets for showing medication over time. (a) The medication of three substances over time. (b) The same as (a) with DDD scaling on. (c) The total dose in DDD terms of the patient over time.

substance, for a quick indication. This does not mean in practice that the extrapolated value should always be close to the real value, because of the underlying complexity of the human body, it serves only as a quick indication.

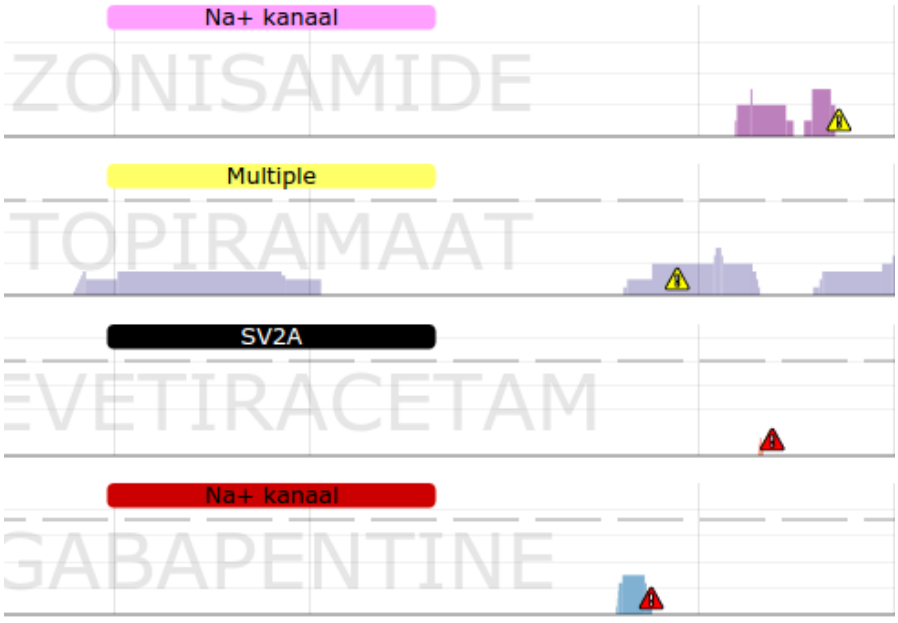
Besides the blood compound level for the substance itself, the values of related substances might have been measured for the study. These are refereed to as *metabolites* and are shown in fig. 5.6b. There are two remaining options in the medication portlet, see Fig. 5.6c: showing of adverse effects and substance type. An adverse effect is attributed to a substance and, hence, it is shown drawn over the substance at the time of occurrence. Substances have effects on different part of the body, and if necessary, a label can be shown indicating the body part it has an effect on.



(a) Blood compound value for a substance.



(b) Values of the metabolites.



(c) Adverse effects are indicated by triangles with an exclamation mark inside. A yellow filled triangle represents a non-serious adverse effect and a red filled triangle represents a serious adverse effect. The type of body parts the substance has its effect on are shown by a label with a colored background (for all color codes see Appendix B). For instance, "ZONISAMIDE" has an effect on the "Na+ kanaal".

Figure 5.6: Chart options for the medication portlet.

5.4.2 Seizures, weights and reports

The seizures portlet shows the seizures distribution over time (Fig. 5.7a). The stacked bars depict the number of seizures over time, the color of a bar indicates the type of seizure, see the legend below. The smallest distinguishable time period on the timeline is a day, meaning that the data is assembled on that basis. Hence, the minimum width of a bar spans a day, see Fig. 5.7b.

Suppose now that the user wants an overview and selects a long interval of time, say 10 years, then the bars become more like spikes, due to the time scaling. The longer the time interval gets, the less space there is remaining for the bars in a given time period. As a result, if the remaining screen space gets down to a small number of pixels horizontally, there might not be enough room left for all the bars in a given time period. In response, we have implemented *temporal binning*, which means that the data that becomes invisible for a given time period, due to lack of screen space, gets binned into buckets representing longer time intervals that are actually visible.

The bucket size is kept to a minimum to preserve as much of the temporal distribution of the seizures as possible. If the timeline gets zoomed, the time interval that is behind two pixels is computed and used as the bin size. The data in each bin is aggregated and visualized as stacked bars of width two, see Fig. 5.8.

Consequently, the user is able to see the correct seizure counts, as indicated by the stacked bars, for a minimal loss in accuracy regarding the temporal distribution of the seizures. The drawback of this approach is that the reciprocal relationship between the bin size and the sum total of the values in the bin, causes the stacked bars to stretch higher. This may cause a rescaling of the vertical axis, which makes perceiving the smaller bars harder. For our data, however, the method was feasible in producing acceptable results.

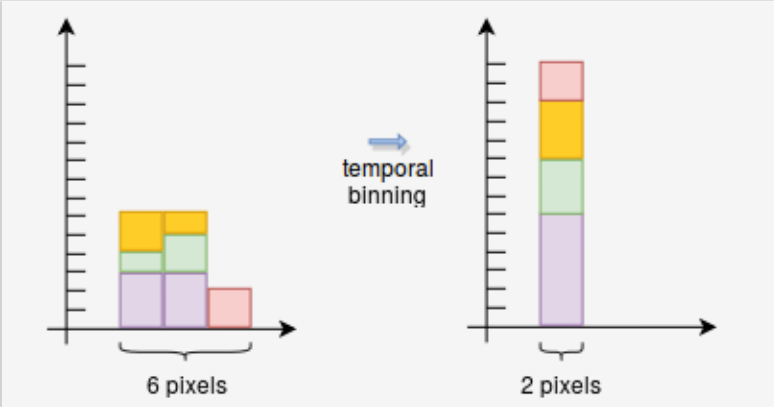
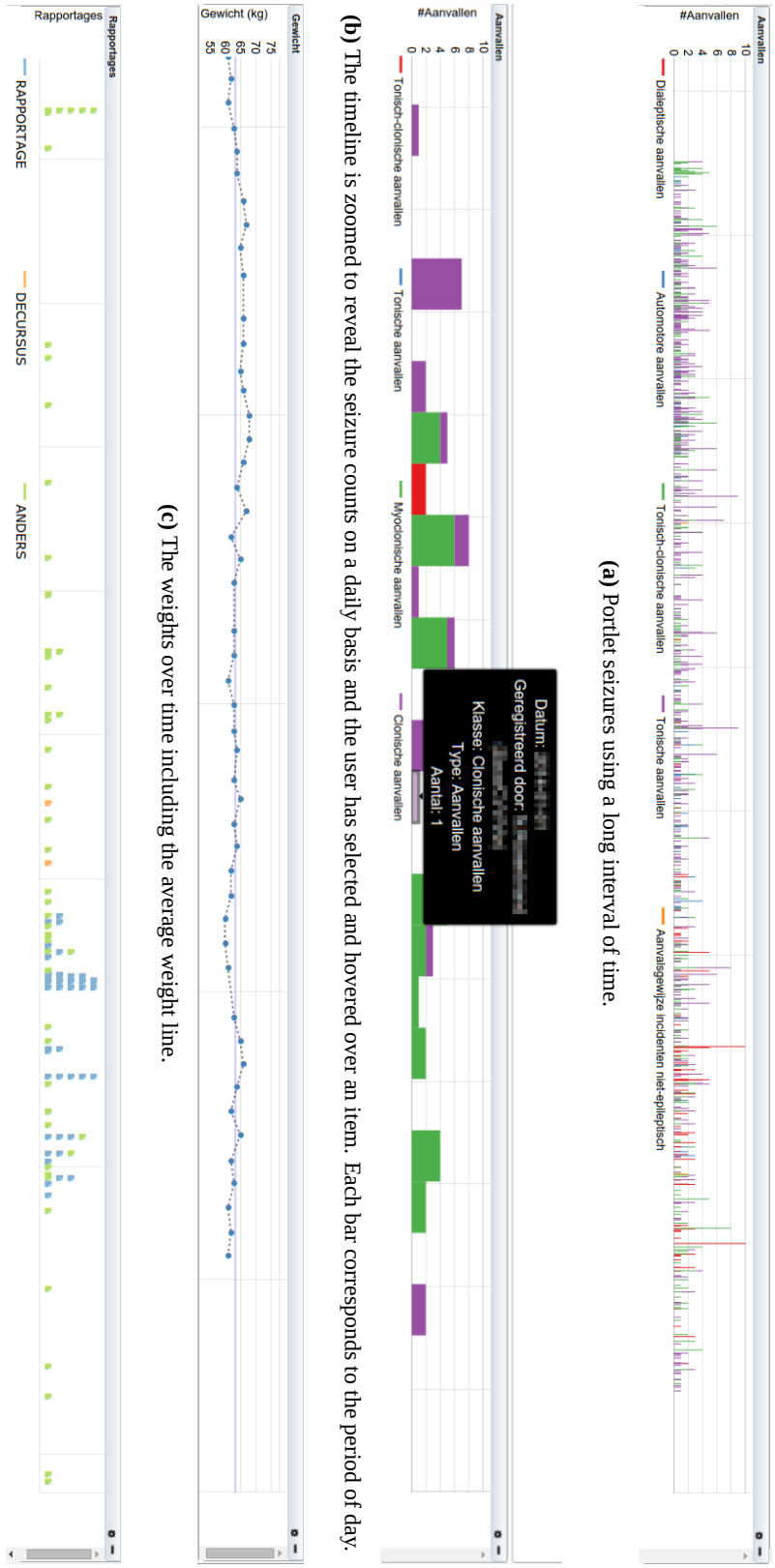


Figure 5.8: Temporal binning: On the left there is data that requires a minimum screen space of six pixels wide to be shown correctly. Under the assumption that the available screen space is only two pixels wide, the data is binned. The data is aggregated and stacked accordingly.

The next portlet, shown in Fig. 5.7a, is for the weight of the patient over time. A line plot shows the measured values and (optionally) a horizontal line shows the average weight of the patient. Finally, there is a portlet for the reports. A report has a type of which two important ones are the "RAPPORTAGE" and "DECURSUS". The charts show a little icon for each report, which is color coded depending on its type.



(a) Portlet seizures using a long interval of time.

(b) The timeline is zoomed to reveal the seizure counts on a daily basis and the user has selected and hovered over an item. Each bar corresponds to the period of day.

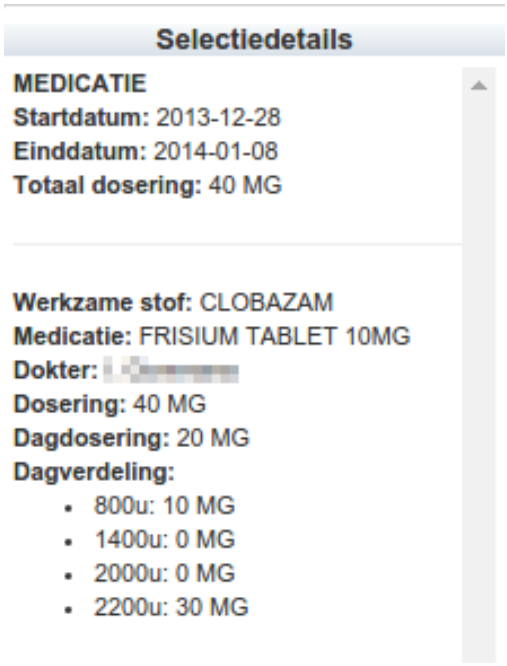
(c) The weights over time including the average weight line.

(d) Reports over time.

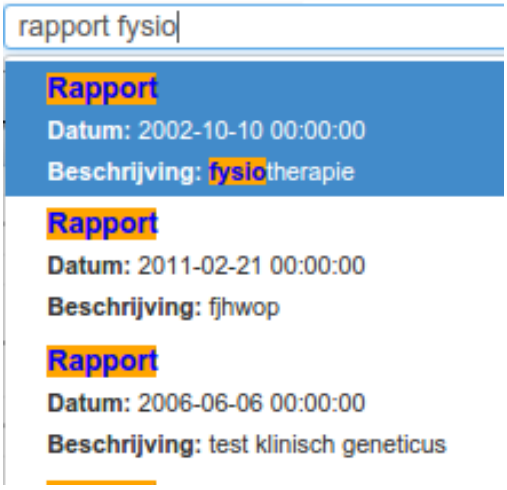
Figure 5.7: Portlets for seizures (a) and (b), weights (c) and reports (d).

5.4.3 Searching and details

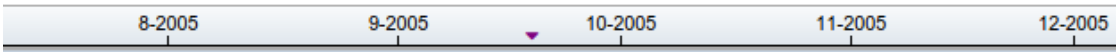
In general the system reacts to mouse hovering and clicking on visual items in the charts. Hovering over an item normally shows the value, like the weight in kg, or the type of seizure and who reported it (e.g., Fig. 5.7b). For more details a user can click on an item. The items details are shown in a designated panel, shown in Fig. 5.9a.



(a) The details for a medication period, showing the time interval, total dose, substance and the distribution of the dose on a daily basis.



(b) Searching on the keywords 'rapport' and 'fysio'. The keywords are highlighted in the search result.



(c)

Figure 5.9: GEHR components. (a) Details panel showing details of a prescription. (b) Search bar filled with keywords to search for reports on physiotherapy. The keywords are highlighted in the results, which are shown below the search bar. (c) Timeline bar with time indicator (pink) for a selected item.

If a medication item is selected, the details panel shows the full details of the prescription period. In case such a prescription period is artificial, meaning it was calculated by combining multiple prescriptions (Fig. 5.4b), all details of the merged prescriptions are shown for that period. In other words, the information of all medicines involved is shown in the details panel.

Doctors currently working at Kempenhaghe expressed their need for a search mechanism. Instead of scrolling their way through events in tabular forms, they wanted a "Google-like" search mechanism to quickly find events from the past. We have therefore added a search bar on the top of the screen for the

input of keywords. The systems searches through all events, performing a full-text search, and filters the events containing the keywords. The matches are shown below the search bar with the keywords highlighted. For an example see Fig. 5.9b.

If the user selects a match from the results (or selects an item in the ontology based selector for that matter), the associated events are panned into view automatically and highlighted. This activates the details panel for that item and a small indication is shown in the timeline bar (Fig. 5.9c).

5.5 Summary

The GEHR shows the EHR of a patient in a graphical way, providing a doctor and / or a patient with a quick overview of the medication history including blood compounds measurements and adverse effects. A doctor may rescale the dosage axes to distinguish substances of which a relatively low dosage is prescribed from substances for which a relatively high dosage is prescribed. In addition the seizure distribution, weights and reports over time are available in the overview. The seizure distribution is rendered as a stacked bar chart for the GEHR prototype, but it is advised to consider the usage of temporal abstraction in further developments of the system.

The user is enabled to navigate through time using a time interval selector and draggable timeline. Furthermore, the system provides flexibility in the usage of screen space with resizeable and moveable portlets that each contain a specific kind of the chart, showing one of the variables over time just mentioned. Portlets have an options panel to customize the contained charts. The items rendered in a chart respond to mouse interactions by showing more details in the form of a pop up or by updating the details panel.

A doctor can quickly filter or select items using the ontology based selector or using the keywords based search mechanism. The graphical overview of the EHR forms a new basis for improved clinical interaction.

The main methods used are found in literature, but the movable, collapsible and scalable portlets, in combination with the ontology selector, enable better usage of vertical screen space than charts on fixed positions. The temporal abstraction used for the seizures respects the temporal distribution of the events as much as possible, by keeping the width of the bars to a minimum, without losing information. Finally, we choose to combine related variables into one chart, based upon ontology.

Chapter 6

Patient Research and Exploration Tool

Clinical researchers at Kempenhaeghe commonly gather patient information for their studies by means of manual retrieval of cohorts of patients from the database in spreadsheet format. We aim for improvement of that situation by means of a system that automatically retrieves this data, offers visual methods for exploration and makes it more easy to share studies among other researchers.

A study normally starts with selection or specification of a cohort of patients that is examined regarding a certain variable, for instance, the effect of a substance over time on the seizure rates. As discussed in Chapter 4, the MPS EHR systems found in literature usually offer charts showing aggregates, temporal distributions of variables, or a comparison widget for multiple event sequences (featuring sentinel events and / or temporal abstractions) of a cohort of patients. The specification of a cohort of patients is generally done via a faceted search mechanism, which implies that if a user wants to compare cohorts with other cohorts of patients, the user has to save each cohort separately.

Although it may be beneficial for Kempenhaeghe to investigate a cohort of patients by comparing events sequences, there is a more compelling first step to be made, namely the specification of a cohort of patients by visual means, instead of working with data in spreadsheet format. We could adopt the faceted search approach, but aim for more flexibility so cohort specification can be partly shared between multiple cohorts and be altered on the fly.

The tool we have developed is dubbed the *Patient Research and Exploration Tool (PRET)* and is shown in Fig. 6.1. A researcher specifies a cohort of patients using a series of filters that are connected graphically in a graph. The result of a series of filters is a group definition that can be inspected visually using the visualization panel.

On the top left a toolbar is shown offering standard functions, like saving the study and zooming the graph drawing canvas in the center. The left toolbar contains the graph drawing modes and tools. There is an element selection mode (arrow pictogram), canvas panning mode (hand pictogram) and a green and red arrow (edge) mode. Furthermore, there are the grouping / ungrouping buttons for selected filters. Thirteen data filters are available and two special filters for computing union and intersections of cohorts of patients (the pictograms of circles with green fillings). Finally there are the patient group and notes pictograms.

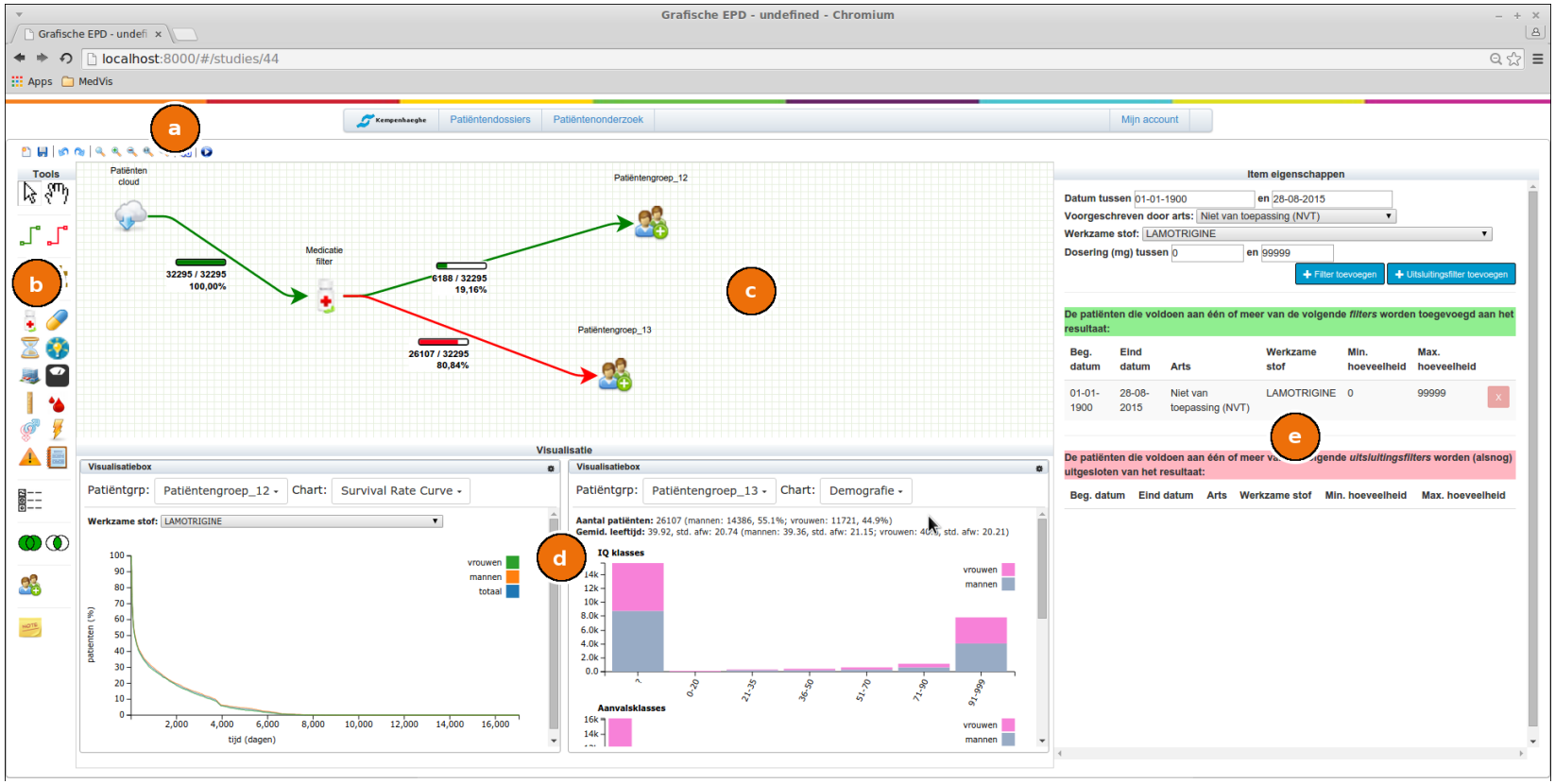


Figure 6.1: Overview of the Patient Research and Exploration Tool (PRET). (a) Main toolbar with new / save buttons, zoom functionality buttons, label mode selector button and run button (b) Graph drawing elements (c) Graph drawing space. (d) Visualization panels. (e) Filter properties panel.

Filters and patient groups can be added as nodes of the graph. If a node is clicked, its properties are shown and editable in the properties window on the right. Lastly, a patient group node can be selected in the visualization panels on the bottom for visual inspection and comparison.

6.1 Cohort specification using a DAG

Inspired by the Outflow system (see Fig. 4.11), we employ a graph drawing tool as visual metaphor for defining sequences of filtering steps, the filtering takes place on various aspects of the patient data and in stages, one aspect after the next, stepwisely reducing the cohort of the patients. If a sequence of filters could fold back on itself we could specify an indefinite filter process. Therefore, we restrict the graph from containing cycles. Also, a sequence of filters must have a clear direction, hence we use a directed graph. A graph with the former two properties is called a *Directed Acyclic Graph (DAG)*.

For an example specification using the tool see Fig. 6.2. The starting point of the graph in the PRET is a node called the *patient cloud*, of which there is always one present on the graph drawing space. It represents the cohort of all patients of which data is stored in the database at Kempenhaeghe.

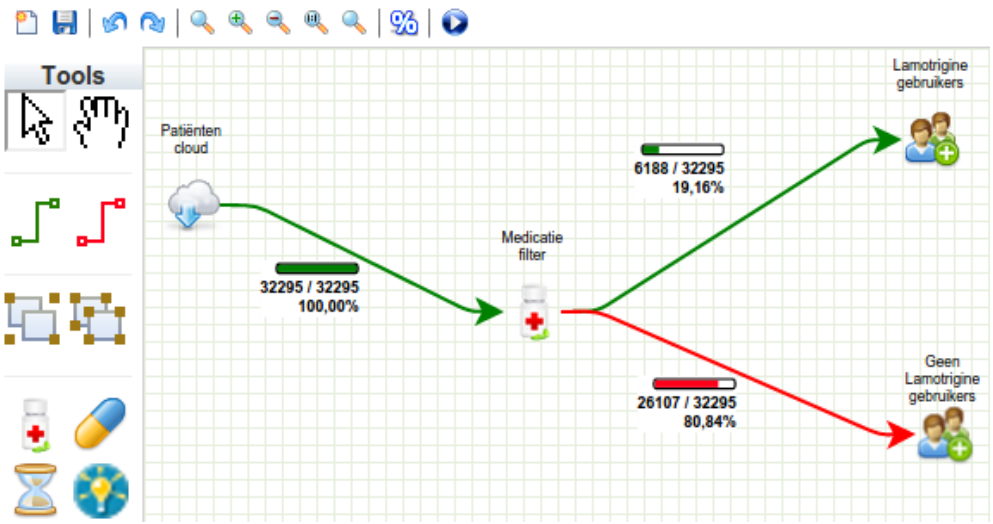


Figure 6.2: Example specification using the PRET. On the top left of the tools panel, there are interaction modes for selection, moving, connection arrows and grouping nodes in the DAG. Just below those are the filter nodes. In the graph drawing space the patient cloud is on the top left, connected via a green arrow to the medication filter. The cohort is filtered on medication and the resulting cohorts are defined as "Use Lamotrigine" and "No Lamotrigine".

A researcher connects, starting from the patient cloud, a series of filter nodes using green and red arrows. Each arrow represents a subset of the original set of patients, before the filter was applied. A green arrow represents the patients passing through the filter and a red arrow represents all patients in the cohort that are filtered out. Either resulting cohort can subsequently be filtered by a new filter node. If a cohort is sufficiently specified it can be connected to a *patient group* node. This node basically assigns a label and color to the cohort for reference. It is possible to continue connecting arrows, starting from a patient group, to more filters. A cohort of patients then simply passes through the patient group node to the next filter. Additionally, there are two "filters" for computing the union or intersection (unconditionally) of

multiple cohorts.

To save space, the researcher may select a set of nodes and define a node group using the buttons below the green and red arrow connector buttons in the tools panel. A group can be collapsed and expanded at will, maintaining the arrow connections. A collapsed group has the size of a regular filter, but typically represents a more complex filter.

When the researcher is done specifying he/she presses the "play" button in the main toolbar. This causes the system to compute the new cohorts of patients and update the labels on the arrow. The labels show the number of patients directed along arrows and the percentage of the patients, with respect to either the patient cloud or the previous node. The user can toggle the percentage mode in the main toolbar.

Let us now consider the difference between specify a cohort using the DAG and using the faceted search mechanism. As stated before in Section 4.4, the filter mechanism behind a faceted search translates the conditions on each facet (of the data) into a query. The intersection of the resulting sets of all these queries defines the final result set.

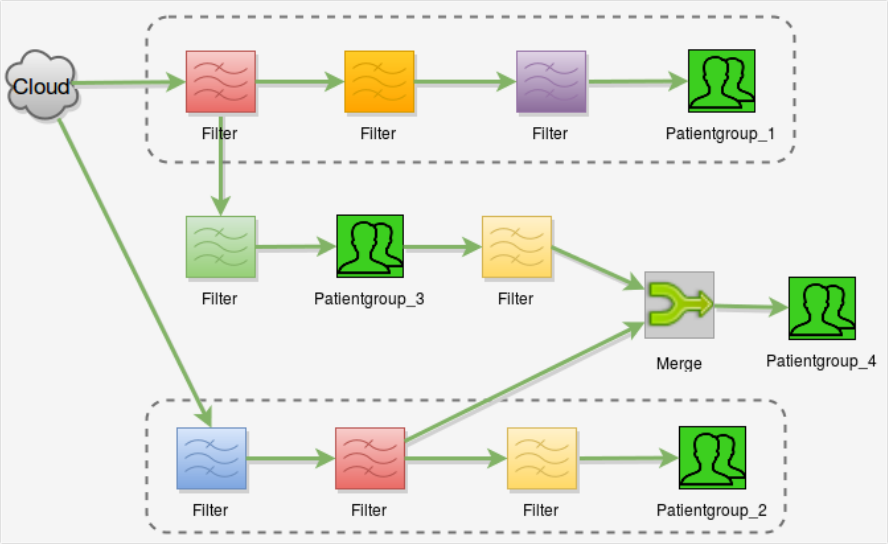


Figure 6.3: Specification using DAG vs faceted search. The filter sequences inside dashed rectangles can be specified using a faceted search, while the filter sequences derived from the dashed sequence cannot. As can be observed from the picture, the comparison options for cohorts of patients (patient groups) is more flexible with a DAG specification.

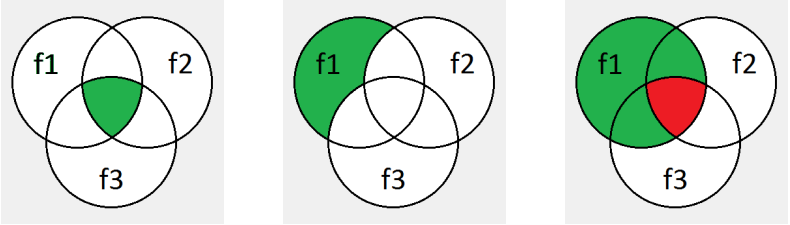
In contrast to the DAG, the faceted search is a linear sequence of filters, see Fig.6.3. Existing EHR systems feature the comparison of cohorts of patients that are defined individually with such a linear sequence. If the researcher wants to split a cohort of patients into alternative cohorts simultaneously, merge them together or inspect these separately, specification using individual faceted search is more involved than using the more flexible DAG.

Moreover, using the DAG a researcher can specify a cohort indirectly by disallowing specified subgroups with more expressiveness than using a faceted search. This difference in expressiveness is illustrated in Fig 6.4. The faceted search results in a group of patients that is computed by taking the intersection of all

subgroups that are defined by conditions on a facet per subgroup, see Fig. 6.4a.

Now consider two arbitrary sets A and B. The set operation $A \setminus B$ returns all items in set A that are not in set B. This operation can be rewritten as $\{x \mid x \in A \wedge x \notin B\}$. Therefore, by negating the condition for a facet in a faceted search, subgroups can be excluded, see Fig. 6.4b. What about excluding subgroups defined by multiple facets? The faceted search may provide setting multiple conditions on a facet, but all these facets remain independent, and therefore, this is not supported.

Using the DAG, any specified subgroup can be used as a filter on another one. Setting conditions on multiple facets simultaneously in such group specification is straightforward. Hence, the DAG offers more expressiveness, see Fig. 6.4c. For instance, a researcher wants to specify a cohort that excludes women over age 55, because a certain medicine is prescribed with caution to women that are fertile and he wants to avoid statistical bias. This requires conditions on the gender facet and the age facet simultaneously, which cannot be specified directly using a faceted search, without excluding men aged under 56. The solution using the DAG is shown in Fig. 6.5.



(a) Conditions on facets f1, f2, and f3 yield three subgroups of which the intersection is taken as the result, indicated by a green color.

(b) Conditions on facet f1 and negated conditions on facet f2 and f3 yield three subgroups of which the intersection is taken as the result, indicated by a green color.

(c) Filtering f1 using a set yielded by the intersection of the groups yielded by conditions on facet f2 and f3 ($f1 \setminus (f2 \cap f3)$), indicated in red, resulting in the set indicated in green.

Figure 6.4: Expressiveness of the faceted search and the DAG. (a) the basic filter logic for a faceted search. (b) filtering by negating conditions in faceted search. (c) filtering using the intersection of multiple facets, only possible using the DAG.

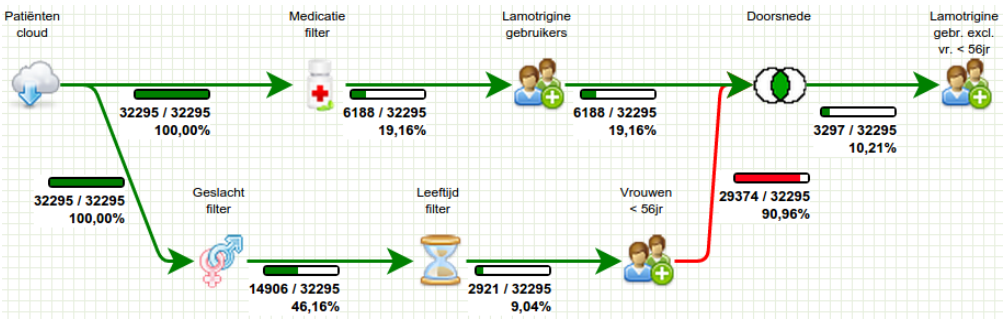
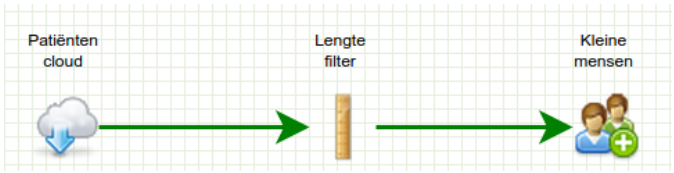


Figure 6.5: Filtering using a specified subgroup and the intersection filter. The age was determined on the first usage of "Lamotrigine". The percentages are with respect to the number of patients in the patient cloud. The negation of a patient group is with respect to the patient cloud (universe).

6.2 Filters

For the filter design we faced the question of filter complexity. On the one hand, a filter can be designed to include all possible aspects of the data, making the filter a complex filter leading to a more compact but less informative graph. On the other hand, a filter can be made a simple as possible, but then more filters are needed in a graph to specify cohorts. We have chosen the approach of keeping the filters as simple as possible, but we added to that the notion of multiple subfilters for that filter type. The rationale behind this is not only to support more experienced users, but also expressiveness.

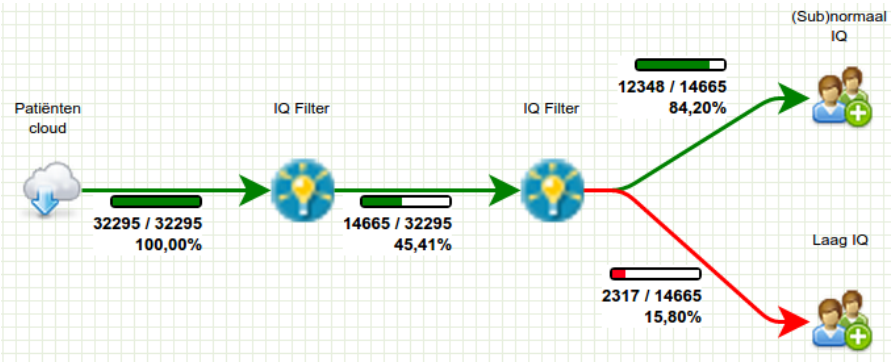


(a) Graph specification for filtering patients on height.

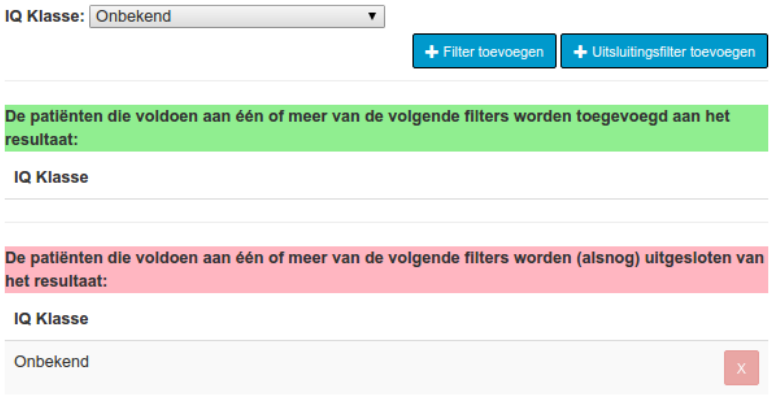
(b) Filtering properties for height filter. All height measurements values must be between 0 - 145cm during the time period starting on 01-jan-1900 and ending on 01-sept-2015.

Figure 6.6: Basic PRET filter. (a) Graph specification for filtering patients on height. (b) The properties of the height filter.

In the basic filter setup, the user sets basic temporal conditions followed by conditions on the data attributes of the entity, see Fig. 6.6. Another example is shown in Fig. 6.7. In this case, the patients are filtered on IQ class using inclusion and exclusion subfilters in combination with green and red arrows. The green arrow directs the patients for which the conditions in the filter apply and the red arrow the rest. Note that the exclusion filters work with respect to the green arrows and not the red arrows. Some more involved filters are shown in Fig. 6.8 and Fig. 6.9.



(a) Graph specification for filtering patients on IQ class. It involves two IQ class filters. The first filters out all patients with an unknown IQ class. The second filters patients with a normal or high IQ class. By using the relative percentage labels on the arrows, we see that 45.41% of the patients have a known IQ class, Furthermore, within that group, 84.20% has a normal or high IQ class.

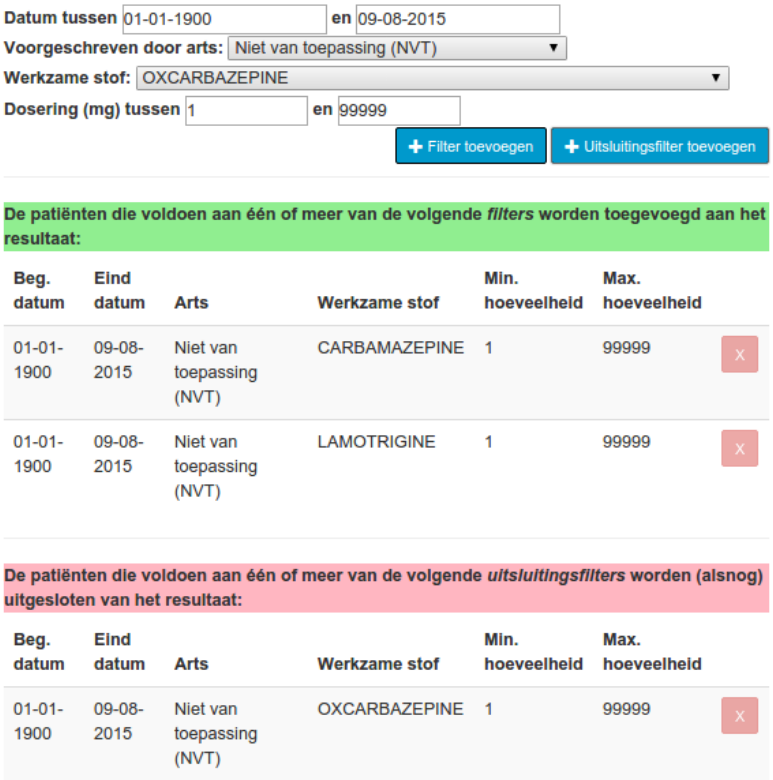


(b) Filtering out patients with an unknown IQ class. Since there are no inclusion subfilters specified, by default all patients are included. The exclusion subfilter, however, does filter *out* all patients with an unknown IQ class.

(c) Selecting all patients with a (sub)normal IQ class.

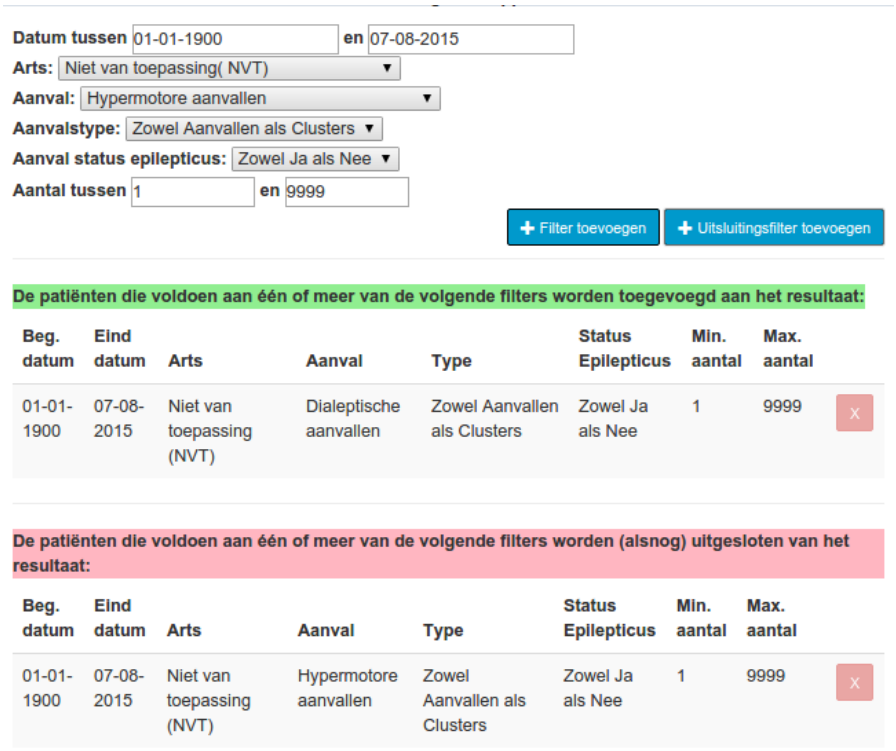
Figure 6.7: PRET specification for filtering patients on IQ class. (a) Graph specification for filtering patients on IQ class. (b) The properties of the first IQ filter. (c) The properties of the second IQ filter.

(a) Selecting all people with an age between 10 and 30 at the moment of their first prescription of "FYTINEZUUR". That first prescription date has to be between 01-01-1900 and 09-08-2015.



(b) Selecting all patient that have been prescribed either "CARBAMAZEPINE" or "LAMOTRIGINIE", but not "OXCARBAZEPINE" between 01-01-1900 and 09-08-2015, with dosage between 1 and 99999mg.

Figure 6.8: Properties of the age filter (a) and the substance filter (b).



(a) Selecting patients that have had seizures of class "Dialeptische aanvallen". We explicitly exclude those patients that have had seizures of class "Hypermotore aanvallen" from the results.

(b) Selecting patients that have had a serious headache ("hoofdpijn") as an adverse effect of the substance "FYTINEZUUR".

Figure 6.9: Properties of the seizures filter (a) and the adverse effects filter (b).

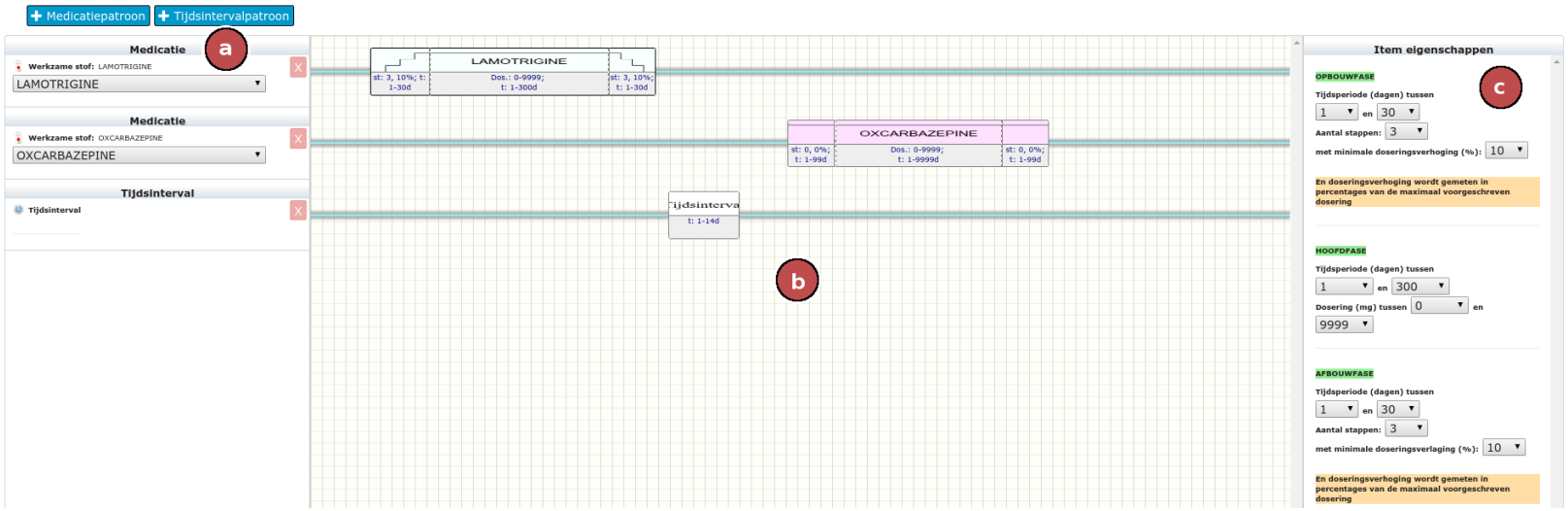
For the prototype, thirteen filters have been defined that roughly correspond to the basic characteristics of a patient and the related entities outlined in the problem statement covered in Section 3.2.2. In particular, we have the following filters.

- **The age filter:** All patients that had an age between a given age range at the moment of their first prescription of a certain substance or the first seizure of a certain type. That moment has to lie in a given time interval.
- **The gender filter:** All patients that are either male or female.
- **The main treatment location filter:** All patients that have a specific main treatment location.
- **The IQ class filter:** All patients that have a specific IQ class (range).
- **The weight filter:** All patients with their weight over time always between a given range and within a specific time interval. This was considered sufficient for now by the domain expert, but there are more possibilities, like filtering all patients that have been losing weight recently.
- **The height filter:** All patients with their height over time always between a given range and within a specific time interval.
- **The blood compounds filter:** All patients having their blood compounds measurements values between a given range, for a specific substance, during a given time interval. Optionally, the associated doctor can be specified.
- **The adverse effects filter:** All patients having an adverse effects count in a given range that were registered between a given time interval. Optionally, the doctor that reported the seizures is explicitly set, as well as the associated medicine and/or active substance. Finally, keyword matching in the associated description of the event can be included.
- **The seizures filter:** All patients having the seizure count in a given range for a given time interval. Optionally, the seizure class, type and status epilepticus can be set.
- **The active substance filter:** All patients prescribed a given substance during a given time interval and with the dosage between certain bounds.
- **The medication count filter:** All patients that have been prescribed a medication count in a given range during a particular time interval. Optionally, only anti-epileptic medicines are included in the count.
- **The reports filter:** All patients that have an associated report count in a given range during a particular time interval. Optionally, only reports issued by a given doctor, having a specific type, or containing certain keywords are included in the count.
- **The temporal pattern filter:** All patients that have a specified temporal pattern in their medication sequences. All sequences have to overlap the given overall time interval.

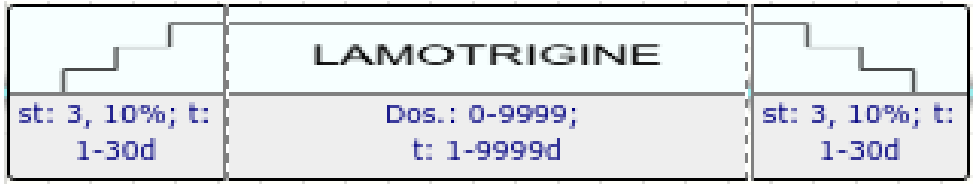
Temporal filtering

With the filters described above, a time window can be specified in which certain conditions on specific variables must hold individually, but what about temporal patterns within the order of events that involve multiple variables? To filter patients with conditions on these patterns, the temporal filter was designed.

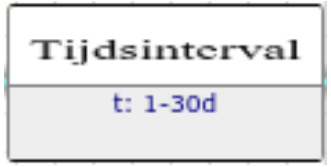
The temporal pattern filter works as basic filter, but the properties of subfilters are set using a dedicated temporal pattern editor, shown in Fig. 6.10d. A pattern consists of a number of element sequences that are specified in juxtaposition. Adding and removing a sequence is done via the panel on the left. Each sequence is specified on a line and for a single element type, where an element represents one ore more events. Elements can be moved or resized horizontally and clicked to set its properties in the panel on the right. Within an element, its properties are displayed textually and / or graphically.



(d) The temporal pattern editor, showing two medication elements and one time period element. (a) Panel for adding and removing element sequences to the pattern. (b) Two element sequences for medication events and one element sequence for time interval events. (c) Element properties panel. In this case the properties for the top left element are shown.



(e) The medication elements has three parts: the titration phase (left), maintenance phase (center) and down titration phase (right). For both the titration phase and down titration phase, the time period is restricted to 1-30 days and the increase / decrease in substance should be in three steps with a minimum of 10 percent of the maximum dose prescribed each. The maintenance phase must have a dose between 0 and 9999mg and its time period is restricted to 1-9999 days.

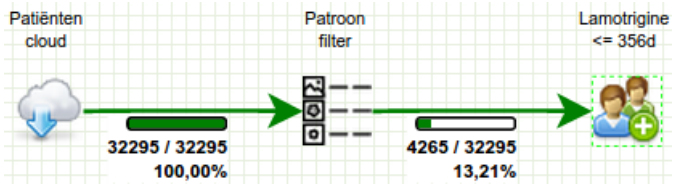


(f) The time period is restricted to 1-30 days.

Figure 6.10: Pret temporal pattern editor. (d) Overview (e) A medication element. (f) A time period element.

There is no ordinary time axis on which to relate the various element to each other, but an implicit *ordinal* time axis: only the *order* from left to right matters. Therefore, the left side of an element represents the start of an event and the right of an element represent the end of it. Consequently, two elements in a sequence define a pattern in which the event represented by the second (most right) element does not start before the event of the first element has finished. For the prototype we have implemented elements for two types of events: medication (prescription of a substance) and time intervals.

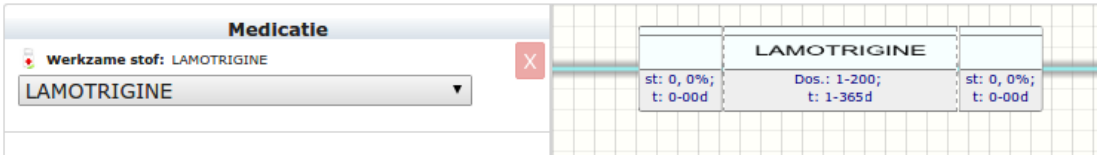
For a medication element, shown in Fig. 6.10e, some details on the distribution over time of the substance can be set. There is the *titration phase* in which the doctor tries out the substance on a patient with increasing dosage, the *maintenance phase* of a substance in which effective dosages are prescribed and the *down titration phase* for stepwise reduction of the dosage. The researcher can set a fixed number of steps for the titration phase, a minimal percentage increase per step (with respect to the maximum value in the temporal distribution of the substance) and time period. The same properties can be set for the down titration phase and the maintenance phase, except for the maintenance phase there is no percentage threshold, but an absolute range (in mg) for the dose. Finally, for the time element a period of time needs to be specified, see Fig. 6.10f.



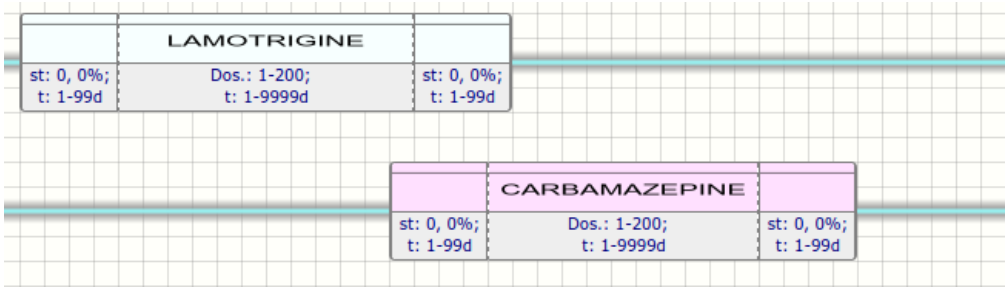
(a) Specification of a group using the temporal pattern editor filter node.

(b) The addition of a new pattern to the filter, which has a large time window that spans the complete history of the data on all patients.

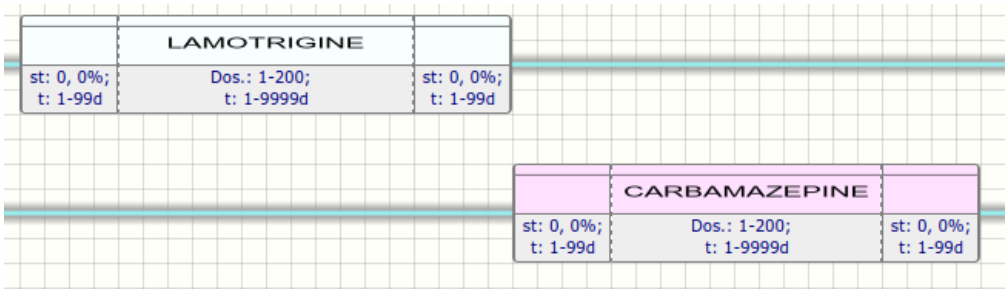
Figure 6.11: Specification of a temporal pattern filter in the DAG.



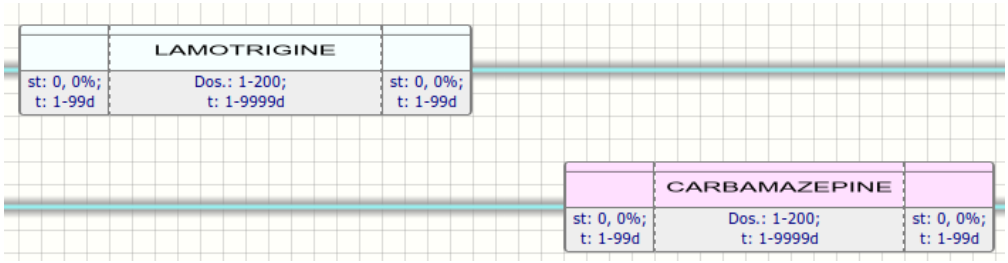
(a) A sequence of one element that specifies a maintenance period of at most one year in which "Lamotrigine" has been prescribed. The dose has to be between 1 and 200mg. Both the titration phase and down titration phase are disabled, by setting a minimum and maximum of 0 days for the time period.



(b) The prescription of "Carbamazepine" starts 0 or more days before the prescription of "Lamotrigine" ends.



(c) The prescription of "Carbamazepine" starts on the same day that the prescription of "Lamotrigine" ends.



(d) The prescription of "Carbamazepine" starts 0 or more days after the prescription of "Lamotrigine" ends.

Figure 6.12: Examples of temporal patterns.

As an example, suppose that a doctor wants to filter al patients that have been prescribed "Lamotrigine" for at most a year. First he uses the DAG to specify a patient group using the temporal pattern editor node and adds a new pattern to the filter, as shown in Fig. 6.11. Then, the added pattern itself is specified using the pattern editor, see Fig. 6.12a. There are 4265 of the 32295 EHRs exhibiting this pattern (Fig. 6.11a).

Analogously, elements of multiple sequences together yield a more complex pattern via the relative po-

sitions of their start and end points. Examples of patterns yielded by two types of medication elements are shown in Fig. 6.12b-Fig. 6.12d. With the time period element we can force time constraints between begin and end points of prescriptions, see Fig. 6.13.

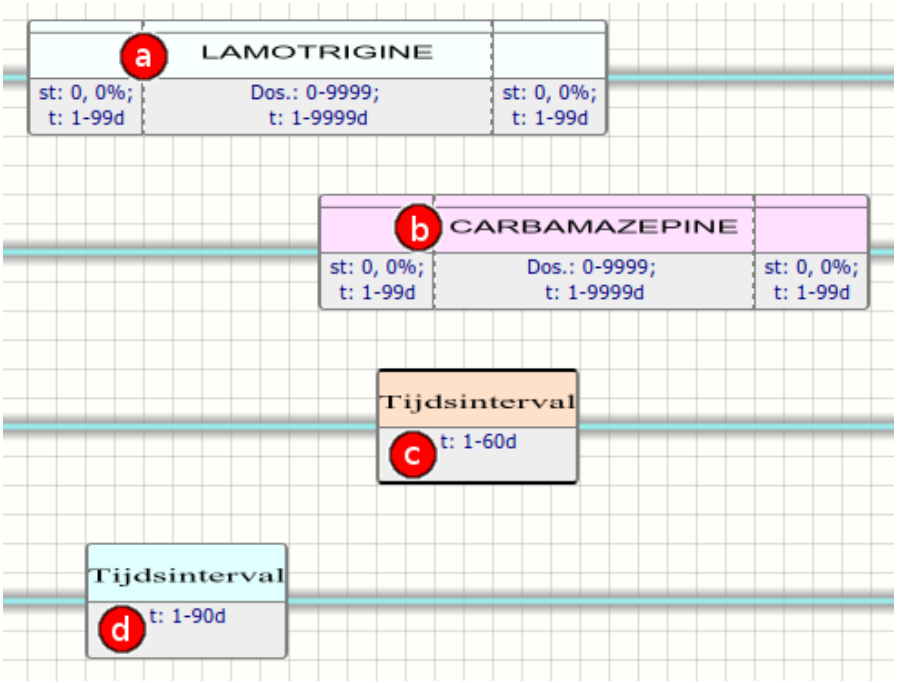


Figure 6.13: Pattern with two time intervals. Time period element (d) is positioned between the start point of prescription (a) and prescription (b) and enforces the condition that the two starting points of the events are a minimum of one day and a maximum of 90 days apart. Similarly, time period element (c) forces an interval of 1 to 60 days between the starting point of prescription (b) and the end point of prescription (a).

In Fig. 6.14 another example of a pattern is sketched. It involves three sequences with elements of type medication and one sequence with elements of the time period type. The start and end points of the elements are projected onto the ordinal time scale, which yields a series of periods. Every two successive points define a period between them. Some interesting periods are described in the figure.

The start of the first prescription of "substance 2" commences before the start of the first prescription of "substance 3" and the end of the prescription of "substance 1" coincides *exactly* with the end of the second prescription of "substance 3"; Since the PRET pattern editor uses a day as the unit for the ordinal time axis, these former prescriptions have to end on the same day. The time period elements explicitly create time periods with conditions on the minimal and maximum number of days. Furthermore, the width of an element in itself has no influence on the series of periods, only the relative order does, but it may be convenient or even necessary to stretch the width of an element while specifying the order of elements.

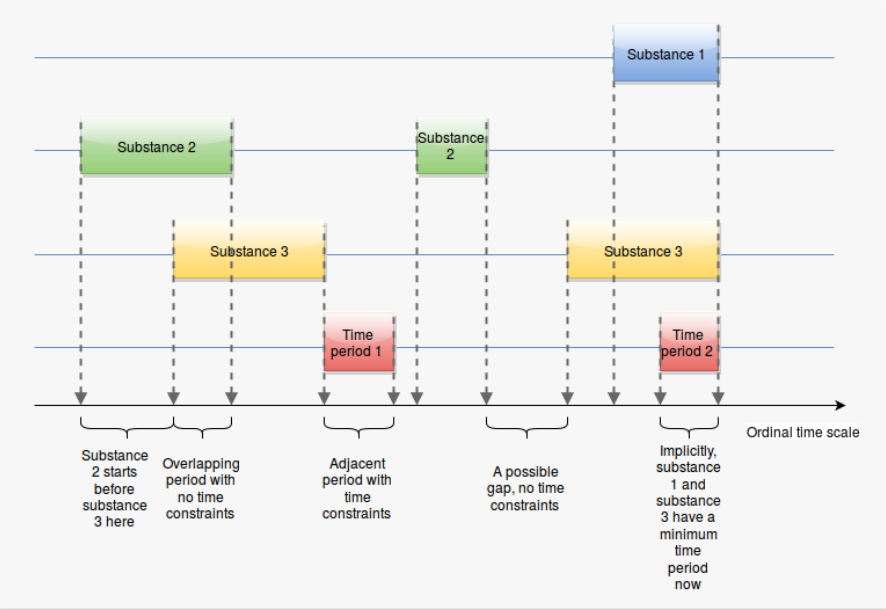


Figure 6.14: The ordinal time scale of the pattern editor.

For an additional use case see Section 8.1.

The PRET filters patients for a temporal pattern filter nodes as follows. First, all patients that have all element types in their EHR, as specified in the filter, are selected. For each selected patient, all elements within the EHR that overlap the overall temporal window of the filter node are selected and projected on an ordinal time scale, in the same way as shown in Fig. 6.14. During the projection all relevant event data associated with the elements are saved along with the start points and end points of the periods on the ordinal time scale for later reference.

Given the projection of the elements in an EHR, the second step is to determine whether there is a subpattern in this projection that matches the pattern of the projection of the filter node. For this purpose, the pattern yielded by the EHR is overlaid with the pattern yielded by the filter node. Starting with both initial periods coinciding, the pattern yielded by the filter node is shifted to the right, in increments of one period, in search for a complete match with a subpattern of the pattern yielded by the EHR.

A match is determined in steps. In the first step it is checked whether the correct events occur the EHR at the start points and end points of all periods in the pattern yielded by the filter node, like the start or end of a prescription. If this is the case, the second step is to check the constraints specified by the time interval elements in the filter node, using the data stored in the projection of the EHR. Finally, in the third step, more details are checked, like dosages and time spans. In case of a complete match, the patient is selected.

6.2.1 Expressiveness

There are two aspects to the expressiveness of the pattern editor in relation with the DAG that we like to discuss: the indeterminate overlap or absence of elements and the temporal aspects of divergence.

Indeterminate overlap and absence

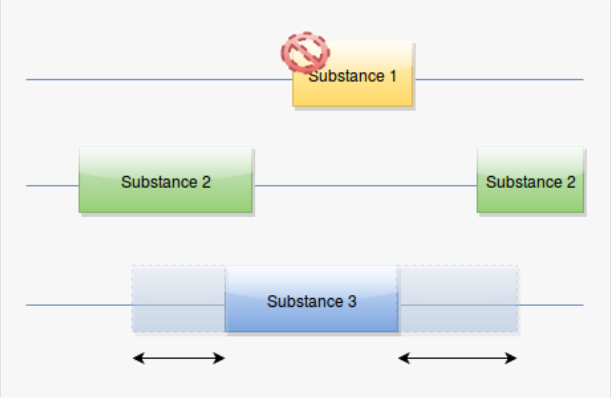


Figure 6.15: Expressing absence and indeterminate overlap of elements. A red glyph on top of the yellow colored element indicates the absence of this event in the pattern. The blue shades around the blue colored element indicate an indeterminate interval, which means that interval is "flexible". This implicitly defines multiple patterns at once. In this case, the prescription of "Substance 3" can either end before the start or after the start of the prescription of "Substance 2".

Systems like EventFlow (Fig. 4.16) enable the user to specify explicitly the absence of events and other systems, like planning systems using Gantt charts, provide a means of specifying indeterminate event intervals, as shown in Fig. 6.15.

Our domain expert stated that those additional means of specifying patterns are too complicated for this first prototype, hence they are left out. Consequently, a clinical researcher using the PRET pattern editor has no means to specify the absence of an events and has to specify alternative patterns explicitly. For high complexity patterns with many alternatives, this leads to a combinatorial explosion of possibilities. Currently, the complexity of the patterns is estimated to be low, but in the future, as this complexity increases, the shorthand of specifying indeterminate intervals should be reconsidered.

Temporal aspect of divergence

We have seen no common methods for specifying alternatives in a temporal pattern. An informal expression like "either event a or event b has to overlap with event c", is not readily translated. Even if there is a hypothetical mechanism to specify alternative subpatterns, what if that subpattern diverges into alternative patterns itself? The more divergence, the more complex handling the mechanism becomes.

In the PRET editor we do have the DAG to support alternative traces on a high level (at the cost of specifying all patterns for each trace explicitly), but we have to be cautious here with the temporal aspects in the DAG. For each node, the temporal window can be specified, but two consecutive nodes can operate on two completely unrelated temporal windows. There is no linear progression in time for a trace through the DAG. Partly, this can be compensated for by specifying adjacent temporal windows. An open question is if there is an intuitive way to preserve this linear progression of time in a trace and what improvements it could offer.

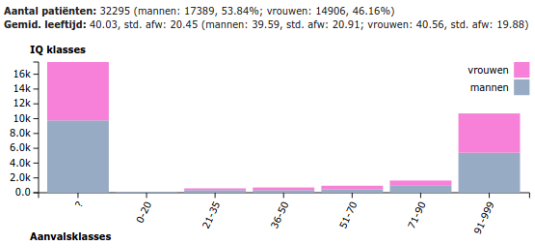
6.3 Cohort visualization

Once a researcher has specified a patient group, it can be investigated via the visualization panel, where two visualizations can be rendered simultaneously for direct comparison. The PRET prototype supports visualizations for the:

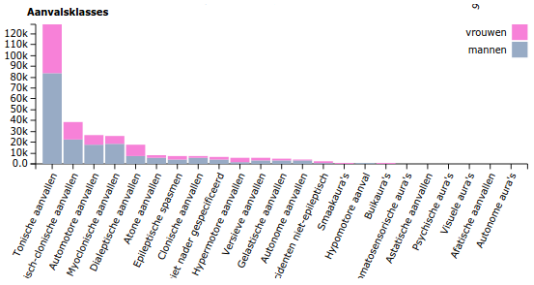
1. **Demography:** basic characteristics of a group;
2. **Medicine usage:** the usage of the top 10 medicines over time;
3. **Anti epileptic medicine count:** the anti epileptic medicine usage count over time;
4. **Seizure distribution:** the seizure count over time, during the prescription of a given substance;
5. **Survival rate curve:** the percentage of patients using a substance over time.

An example of all visualizations is found in Fig. 6.16. The seizure distribution and survival rate curve are actually used in clinical research, see for instance the work of *Boutsma* [4].

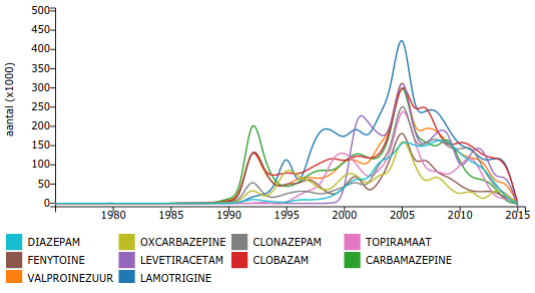
Fig. 6.16a shows that for the majority of all patients treated at Kempenhaeghe, the IQ class is either unknown or normal. In the seizure distribution, shown in Fig. 6.16b, by far the most common is the tonic seizure class. According to Fig. 6.16c, Fig. 6.16d and Fig. 6.16g, there is a huge peak in the number of medicine prescriptions around 2005 and a smaller one around 1992. Note that the line plots show a sudden downward curve towards 2015. This is because the charts were rendered with a snapshot of the data up to 2014 and the data of 2015 was not available. In Fig. 6.16e and Fig. 6.16f we observe a significant increase in the number of seizures after two years of "Lamotrigine" usage and generally more seizures have been reported for males than for females. Finally, Fig. 6.16h shows that the percentage of patients still using "Lamotrigine" after 6000 days approaches zero. Note that the curves for the males and females are almost identical.



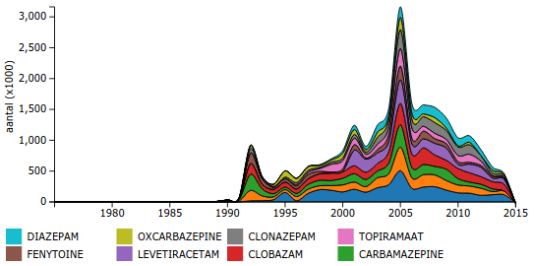
(a) Demography: patient count (male / female), average age, and IQ class distribution (stacked for male / females).



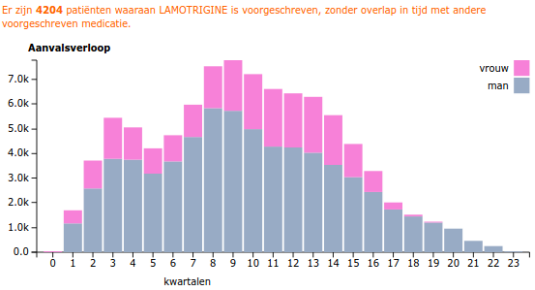
(b) Demography: seizure types distribution (stacked for male / females).



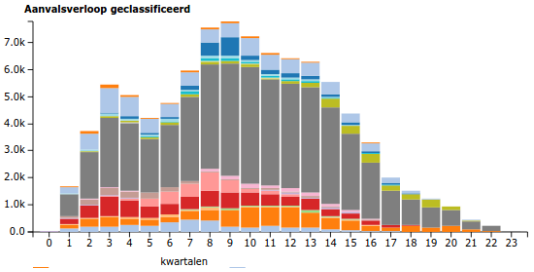
(c) Plots of the top 10 medicine prescriptions over time.



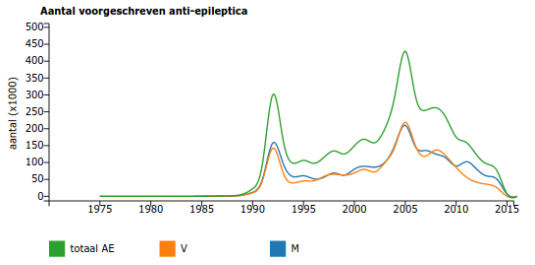
(d) Stacked plot of the top 10 medicine prescriptions over time.



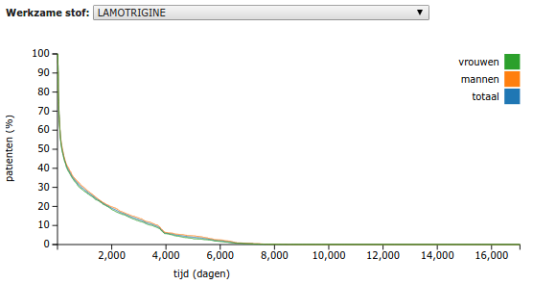
(e) Seizure distribution over time for a given substance (stacked for male / females).



(f) Seizure distribution over time for a given substance (stacked using seizure class).



(g) The anti epileptic medicine prescription count over time.



(h) The survival rate curves for a given substance (plotted for total, males and females).

Figure 6.16: PRET visualizations (a)-(h).

6.4 Statistical aspects

If a cohort is filtered, one needs to keep in mind that the numbers reflect what is known in the data *so far*. The registration of adverse effects, for instance, started much later than that of medication, introducing a bias. Researchers need to keep in mind these type of biases, before drawing conclusions.

The PRET has currently no support for checking the statistical prerequisites when comparing two cohorts. It is up to the researcher to select a cohort properly. For this prototype we offer the demography chart, but this is just a first step. The proper use of statistics in clinical research has been addressed by *Feinstein* [8].

Feinstein explains that the common statistical tabulations in medical literature do no necessarily represent a disease correctly and that the use of standard statistical methods is not sufficient for proper scientific clinical experimental research [8, p. 17-27]. A clinical experiment is about studying the sequence

$$\text{initial state} \xrightarrow{\text{manoeuver}} \text{subsequent state}.$$

The maneuver can be chosen by nature, the investigator, or by the person who acts as the experimental preparator.

Most of the experiments cannot be adequately planned with the concepts described in current statistical writings about experimental design (more recent work may have improved this situation, but we still need to take note of the fundamental aspects). Statistical strategems can often be used successfully if the scientific architecture has already been well designed. For the types of clinical experiments described above, many times the statistical models are not pertinent or too superficial for the fundamental demands of scientific rigor.

The statistical tactics of "design" depend upon the basic assumptions that

1. the research is being performed as an experiment,
2. the experimental material and its responses can be reproducibly identified,
3. "random samples" can be readily obtained, and
4. "random allocations" will provide satisfactory solution to problems in "control".

For scientific design in clinical investigation, these assumptions are often erroneous or naive. This might suggest why the analytic focus strategy of random sampling is absent for all systems in Table 4.1.

Feinstein [8, p. 25-26]) forwards five basis prerequisites to scientific research:

1. Did the researcher choose the right people to question or experiment on?
2. Did the researcher choose a statistical unit that made his problem solvable?
3. Did the researcher use a control group and choose and use it properly?
4. Are the groups truly comparable?
5. Did the researcher guard against a probable bias in the people he was testing?

Underlying these questions, is the problem of bias. A patient ends up in a cohort via a sequence of choices and circumstances. These include choices based upon subjective reasoning by the patient, as to how to proceed with treatment, and varying diagnosing capabilities of a doctor. If a researcher selects a sample

of the population, these underlying factors can introduce a *sampling bias*. This occurs when members of some segments of the population are not sufficiently represented in the sample.

For example, suppose you want estimate the number of times people favor driving a car instead of riding a bicycle, when it is not necessary to take the car. Therefore you create a poll you ask people to fill in. If the majority of your sample population consists of people who are a member of the bicycle club, you create a bias by favoring people who already strongly like riding a bicycle over people who dislike riding a bicycle. Such a sample is not a good representation of the general population and generalizing statistical results based upon such samples leads to errors.

This previous example shows the need to reduce bias as much as possible. Therefore, when selecting a cohort of patients we need insight in the sequence of events and circumstances that lead to each patient becoming a member of a cohort. Hence, we need to use this so called "soft data" in conjunction with the the "hard data" that usually only involves concrete numbers (of measurements).

The PRET supports some filters that include keyword based filtering of patient events, but if in the future soft data is properly structured and recorded, new filters in that regard can be added within the same paradigm. Possible sources of bias can then be investigated by alternating the tweaking of those filters with the visual inspection of the cohorts. Additionally, segments of the population can be explicitly excluded or merged using the DAG.

According to *Feinstein* [8, p. 197-213]), there are two ways of collecting research data on patients. In the first form, the data is first collected by people that are not performing a specific investigation and recorded in routine records. Then, the data is extracted from these records by a researcher for an investigation. This form of collection data is called *retrolective*. Alternative, there is the *prolective* form, in which plans are made before actually collecting any data on patients.

Regardless of the way in which the data was collected, there are two directions of populational pursuit, forward or backward. In forward research we can, for instance, create two groups of people. One group is given a medicine and the other group is not. Then the development rate of a certain condition is monitored. In backwards research, we first assemble a group of people that has the condition and one group that has not. Then we take note of the proportionate frequency with with the groups had previously used the medicine.

The term *cohort* describes a group that is pursued in the forward or *prospective* direction. Conversely, a group that is pursued in the backword or *retrospective* direction is called a *trohoc*, a term coined by Feinstein. Depending on the principal maneuver that is studied, the cohort may be divided for comparison of the effects of the maneuver. The main problem with trohoc research is, given some observed effects, to trace back the causes.

To trace back the causes, a trohoc may be divided into segments with distinguishing properties. In general it is impossible to proof the absence of (all) bias, because the full sequences of hidden events that lead to the observed effects for the group are not known. The best a researcher can do is to apply as much as domain knowledge as possible to compensate for as much bias as possible. Also, the calculation of (rates of) probabilities are easily erroneous due to the underlying bias.

It is an interesting question if we can extent the DAG in such a way that the temporal progression is maintained. This could then possibly support effective forward reasoning or backward reasoning. Also, auxiliary features can be more effective if the PRET is sensitive to the direction in which the reasoning takes place. For instance, the PRET may inform the researcher about the impact that several (automatic-

ally) identified segments of a trohoc have on the statical results.

By extension, the visualizations of statistical data must be accompanied with the data the statistics are derived from. For the PRET this means that only showing the end result of a statistical calculation is not enough. It should be as clear as possible on how the calculation led to the end result that is shown. The user should be enabled to explore and adjust the underlying segments of a cohort / trohoc interactively, in combination with the DAG. If statistics are computed based upon other statistics, the researcher should have the ability to fully inspect these underlying statistic factors as well. The improvement of the current system using all of these insights, requires more research in cooperation with the clinical researchers at Kempenhaeghe.

6.5 Extensions

Currently, the time windows of two sequential nodes are independent. An interesting question is whether temporal progression can be maintained in traces through the DAG. Additionally, the readability of the DAG might be improved, such that a user can quickly interpret the DAG on a deeper level, without inspecting the properties of the nodes. The property panels of the nodes can be improved by employing more graphical methods for setting the conditions on the filter that provide more realtime insight in the incoming / outgoing data of the filter. The work on *Scented Widgets* by *Willett et al.* [33] is a good starting point.

Furthermore, the pattern editor should be extended with the remaining temporal entities, like seizures, adverse effects, blood compounds and reports. In this way, the temporal components between those entities can be fully utilized in the filtering process. Also, new visualizations can be added that show statistical properties and indicate possible sources of statistical bias. If visualizations are added that show the involved doctors, protocol violations can be reported on a per doctor basis.

Next to coming up with new filters, it would be beneficial to add import and export functionally to the PRET, preferably in spreadsheet format. Perhaps the patient cloud is then replaced with an import source and the visualization panel offers ways to investigate the patients within a cohort using graphical means (e.g., event sequence comparison), but also using a table. Clicking on a patient the GEHR is triggered for that patient.

To the pattern editor one could add some artificial intelligence that predicts common patterns while specifying a pattern. This could be based on, for instance, a *Bayesian network*.

6.6 Bridging the gap with the GEHR

A direct way to bridge the gap between the GEHR and the PRET is to add the capability of clicking on an item in the GEHR and compare that value with a cohort of patients, which was defined earlier in the PRET. For all variables in the EHR, one could either reuse an existing visualization of the PRET or design a dedicated one. Single values of a variable can be put into perspective by direct comparison with the distribution of that variable over a group of patients. We have added an example of this feature to the prototype of the GEHR, which is shown in Fig. 6.17.

For temporal variables, the sequence can be compared with the sequences of a group of patients, either directly or with an abstract summarized version of the sequences of the group. Furthermore, one could add a button to the GEHR for generating a DAG in the PRET based upon the active EHR. All static variables could be translated into individual filters and all temporal variables could be translated into subpatterns

of a temporal filter. Starting from this, a clinical researcher can start tweaking the filters to find similar patients, for further investigation.

6.7 Generalization

The PRET method of using a DAG with filter nodes is not domain dependent. It is a filter methodology, like the faceted search, which can be adapted to other domains by redesigning the filters and visualizations in accordance with the domain ontology and entities.

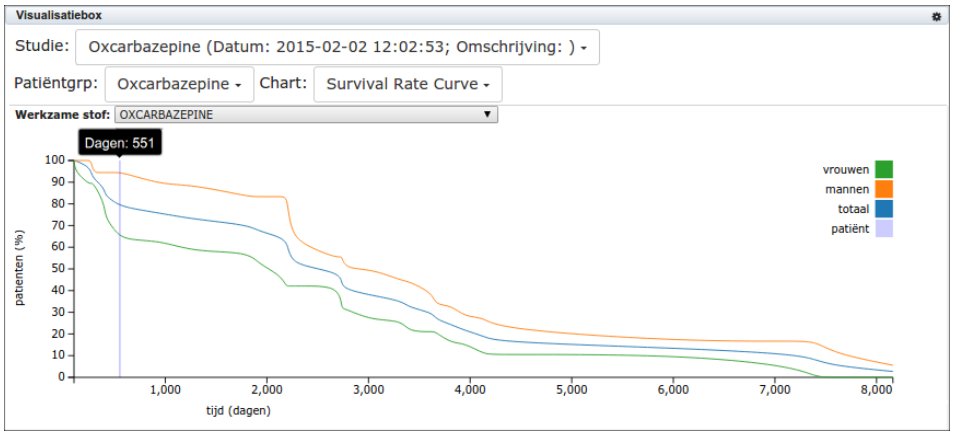


Figure 6.17: Example feature for combining the GEHR with the PRET. The retention time (first usage) of "CARBAMAZEPINE" is compared, in a pop up chart, with a cohort consisting of all "CARBAMAZEPINE" users, defined earlier in the PRET. It shows the percentage of patients that uses the substance over time (days). Up to roughly 500 days there is a significant drop in percentages of male patients that are using the substance. Similarly, there is a significant drop for female patients after 2000 days. The purple vertical line indicates that the patient has been using the substance for 551 days.

6.8 Summary

We have designed the PRET that aids researchers and doctors at Kempenhaeghe in their studies of cohorts of patients. It enables automated retrieval of patient information of the database and interactive exploration of it. Moreover, researchers can share their studies and findings more easily.

The PRET is designed for flexible cohort specification using a DAG, as opposed to the faceted search method. Using simple filter nodes, cohorts of patients are specified one step at a time. In this setting, also temporal patterns are first class citizens in the DAG as another filter node. Each node can be split into alternative cohorts and two cohorts can be merged together using union and intersection operations, allowing for more complex cohort specification than possible with a faceted search.

Intermediary cohorts of patients can be compared visually with their demography and explored using charts on medication and seizures. This also enables the user to explore the data more rapidly, by tweaking a filter and updating the charts immediately thereafter. The temporal pattern filter can be easily extended to include more entities in the patterns. It is also straightforward to apply the method of a DAG with filter nodes in other domains, if the filters and visualizations are redesigned in accordance with the domain ontology and entities. Finally, we have considered the possibilities of bridging the GEHR with the PRET and implemented one of these that shows how the clinical interaction can benefit from the bridging.

Chapter 7

Implementation

In this chapter the implementation is briefly discussed. First an overview is given and then the tiers of the software architecture are presented.

7.1 Overview

As mentioned in Section 3.3 on the research objectives, the application is in the form of a web application. The GEHR and the PRET are conceptually two subsystems that are only connected via the database, but for the implementation we used a shared code base. Especially for the bridging example of the GEHR and the PRET this was convenient. For an overview of the system, see Fig. 7.1. The system is a three-tier application, consisting of data tier, processing tier and presentation tier.

7.2 The data tier

For the data tier of the application we rely on the *PostgreSQL* [27] database server. The data Kempenhaeghe has provided has been augmented and imported in the database, leaving the original important links intact, such that the system is easier to integrate if it gets out of prototype stage. The used database scheme is found in Appendix A.

7.3 The processing tier

This tier is backed by the *Laravel PHP Framework* [21]. It is responsible for sessions management and performs most of the computations on the data, before sending it to the presentation tier. The models and controllers that are coded reflect the entities in the data. Routes are coupled to the controllers using the RESTful resource naming convention, which prescribes a standard for URL formats. The HTTP requests are passed down via the Nginx web server. The other way up, the PHP code returns data using *Web Standards Model* [30] and the *Extensible Markup Language (XML)* [29].

In support of the presentation tier, where a lot of the computing on visualizations goes on, there is much processing dedicated to preprocessing the data from the database in such a way that there is minimal computational overhead in the presentation tier. The timeline data for the GEHR, for instance, is delivered as one precomputed data structure that is shared by the portlets.

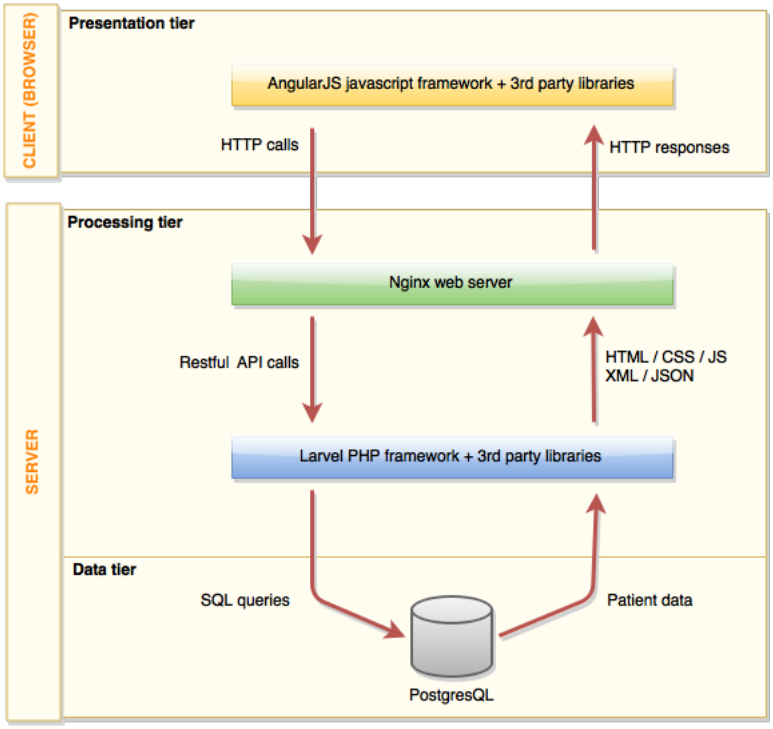


Figure 7.1: Overview of the system.

Before updating the DAG in the PRET the order in which the nodes have to be processed is efficiently calculated using an algorithm for *topological sorting*. To find the next node to process in case of a small number of nodes, however, a loop to search the next unprocessed node for which all parents nodes have already been processed suffices. We keep track of all intermediate results generated by the nodes. The results are stored in dynamically created tables with names related to the session and id's of the nodes in the graph. Every time an update is required, the temporary tables were reused for the new intermediate results.

The presentation tier performed well using this setup, but the query times indicate a steep increase in processing time if the number of patients goes up. This is mostly due to the structure of the data and the temporal nature of some queries, involving many conditions to be checked. For the prototype the response time was good enough: all statistics for the graphs shown in this work were computed within 30 seconds, running the web server and database server locally. Cohort visualization is done efficiently, because the data of a cohort is cached for each node in the graph. Still the number of patients increases and the filters become more complex, the need for more efficient support for (temporal) querying by the data tier will increase.

7.4 The presentation tier

The presentation tier is supported by the *AngularJS JavaScript MVW Framework* [9] and several third party JavaScript libraries. All the JavaScript files are compressed into one file using the *RequireJS JavaScript file and module loader* [6]. This means that the user downloads a single JavaScript file containing

the entire applications once. The advantage is an increase in response time for the application on the client side.

AngularJS provides a convenient framework on top of JavaScript to make front end programming more structured and reusable by means of a directive. A directive is a means to specify a new custom HTML tag that is (almost) as easily reused as classic HTML tags. The GEHR and PRET are hierarchically divided up into panels and finally charts or property windows. All of these are coded as (reusable) directives. When coding web applications for visualization purposes it is well worth investing effort into using the AngularJS framework for this return of investment on the reusability aspect.

For the visualisations and graph drawing panel we relied mainly on the *D3* [3] and the *mxGraph* [13] JavaScript libraries.

7.5 Code size

The main components of the code are listed, along with the lines of code, in three tables separately. The fifteen Laravel controllers are shown in Table 7.1, the 47 directives and eight AngularJS controllers are shown in Table 7.2 and Table 7.3. In total we have 15394 lines of code for the main components.

A client using the application has to download a single JavaScript file, containing the application of 1.8 MB. This is large, but once it is downloaded and cached, the application only transfers data occasionally with the server. This results in a highly responsive application. The size of the PostgreSQL database dump, including the database scheme, is 13.5 MB.

Laravel controller	LOC	Description
AuthController	80	Handles user authentication
Bloodcompound-sController	18	Transfers blood compound resources
DoctorsController	93	Transfers doctor resources
HomeController	22	Transfers main template and JavaScript file through the only Laravel view defined for the system
IQClassesController	16	Transfers IQ class resources
MedicinesController	76	Transfers medicine resources
PatientsController	606	Transfers patient resources, including the timeline data for the GEHR
PrescriptionsCon-troller	107	Transfers prescription resources
ReportsController	11	Transfers report resources
SeizureClassesCon-troller	16	Transfers seizure class resources
SeizuresController	107	Transfers seizure resources
StudiesController	92	Transfers study resources
StudyToolController	3236	Backend for the PRET that transfers cohort data and handles the updating of the graph
SubstancesControl-ler	24	Transfers substance resources
UsersController	126	Handles the users of the system
Total	4630	

Table 7.1: Code size of the Laravel controllers.

AngularJS directive	LOC	Description
timeline/portletDose	314	
timeline/portletSeizures	509	
timeline/timelineDragbar	138	
timeline/portletWeights	326	
timeline/timelineBrushbar	426	
timeline/timeline	29	
timeline/portletReports	400	
timeline/portletMedication	1003	
studyTool/studyToolPortletVisualization	245	
studyTool/studyToolPropertiesGenderFilter	63	
studyTool/studyToolPropertiesUnion	20	
studyTool/patternMedication	156	
studyTool/patternSeizureElement	291	
studyTool/patternEditorPropertiesMedicationElement	98	
studyTool/patternEditorPropertiesSeizureElement	63	
studyTool/patternSeizure	157	
studyTool/patternEditorSeizureSelector	67	
studyTool/patternEditor	196	
studyTool/patternEditorTimePeriodSelector	32	
studyTool/patternTimePeriod	149	
studyTool/patternMedicationElement	593	
studyTool/patternEditorPropertiesTimePeriodElement	58	
studyTool/patternEditorMedicationSelector	68	
studyTool/patternTimePeriodElement	287	
studyTool/patternEditorProperties	68	
studyTool/studyToolProperties	261	
studyTool/studyToolPropertiesWeightsFilter	112	
studyTool/studyToolPropertiesReportsFilter	167	
studyTool/studyToolVisualizations	32	
studyTool/studyToolPropertiesPatientList	78	
studyTool/studyToolPropertiesMedicationFilter	159	
studyTool/studyToolPropertiesAgeFilter	174	
studyTool/studyToolPropertiesPatternFilter	187	
studyTool/studyToolPropertiesBloodCompoundsFilter	160	
studyTool/studyToolPropertiesAdverseEffectsFilter	221	
studyTool/studyToolPropertiesSeizureFilter	187	
studyTool/studyToolPropertiesMainLocationFilter	101	
studyTool/studyToolPropertiesIntersection	20	
studyTool/charts/seizureRemission	395	
studyTool/charts/antiEpileptic	206	
studyTool/charts/medicineUsage	358	
studyTool/charts/survivalRateCurve	256	
studyTool/charts/demography	349	
studyTool/studyToolPropertiesEpilepticaFilter	128	
studyTool/studyToolPropertiesHeightFilter	112	
studyTool/studyToolPropertiesPatientGroup	50	
studyTool/studyToolPropertiesIQFilter	109	
Total	9578	

Table 7.2: Code size of the AngularJS directives.

AngularJS Controller	LOC	Description
patientTimelineCtrl	408	Core of the GEHR frontend
homeCtrl	6	Starting point of application
loginCtrl	43	Handles users logging in
applicationCtrl	15	Core of application
patientsCtrl.js	62	Patient search page
logoutCtrl	9	Handles users logging out
studyToolCtrl	547	Core of the PRET frontend
studiesCtrl	75	Studies search page
Total	1186	

Table 7.3: Code size of the AngularJS controllers.

Chapter 8

Evaluation

We have presented our work in the previous chapters, in this chapter we evaluate this. We start with some use cases, next review the research objectives, and finally perform a comparison with related work.

8.1 Use cases

The three use cases below are defined to capture the essence of the GEHR and the PRET.

GEHR

Consider a patient visiting the doctor complaining about a couple of seizures he has had in the past week. In response, the doctor uses the GEHR of the patient with the focus on the recent events, using the time period selector. The doctor notices a recent dose change for the prescribed substance, using the medication portlet. He also notices that some adverse effects have been reported by the patient since the change.

The doctor clicks on the adverse effects for more details and starts searching for the two most frequent ones using the search bar. A couple of matches show up and he immediately notices that the dates associated with the matches are close to each other. Then he navigates to the interesting time period and notices that a seizure, of a specific seizure class, is spiking during that time, after the same initial adverse effects. It seems to the doctor that recently reported events might be indications that the specific seizure is again about to spike. He knows that that particular seizure is often caused by a high value of one of the metabolites of the substance.

The doctor enables the showing of blood compounds with metabolites and sees that it is indeed the case that the value of one of the metabolites has been increasing steadily over recent times. Hence, he explains his findings to the patients using the GEHR. The patients sees the pattern in the series of events that lead to the seizures previously and recognizes a similar pattern building up in recent times. The patients now knows the pattern and will be more alert in the future. The doctor changes the dose and starts prescribing another substance too, with full understanding on the side of the patient.

PRET 1

Suppose that a clinical researcher is interested in exploring the seizure distributions of young adults (aged 20 to 39) and middle aged patients (aged 40 to 64) that have been prescribed "Lamotrigine" or "Oxcar-

bazepine". The age is measured at the moment the substance was used for the first time. To compare the seizure distribution we want to filter out patients that were using multiple substances simultaneously. The dose over time for Lamotrigine must be between 0 and 600 mg (under 2 DDD) and the dose over time for Oxcarbazepine must be between 0 and 2000 mg (under 2 DDD).

The doctor opens the PRET and adds a substance filter to filter out all patients using Lamotrigine and Oxcarbazepine with the dosages as required. Next, he places two age filters to filter out the two age groups. Finally, he defines two patient groups for visualization. The result is shown in Fig. 8.1. Next he specifies the filters, as shown in Fig. 8.2 and updates the graph. He selects the seizure distribution chart in the visualization panel and selects the substance under consideration. The PRET automatically excludes the data of patients that are using multiple substances simultaneously before rendering the chart. Finally, he compares the charts of the groups, shown in Fig. 8.3. A new research question comes to his mind about why there is such an increase in seizures for young adults (especially males) using Oxcarbazapine, after 15 quarters, which is not there for middle aged patients.

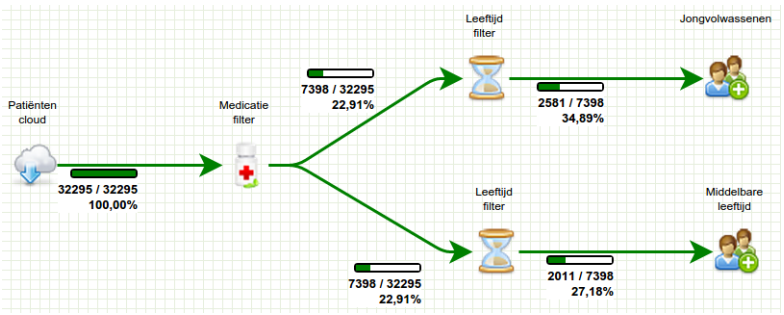
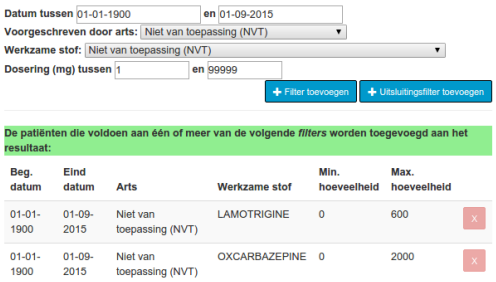


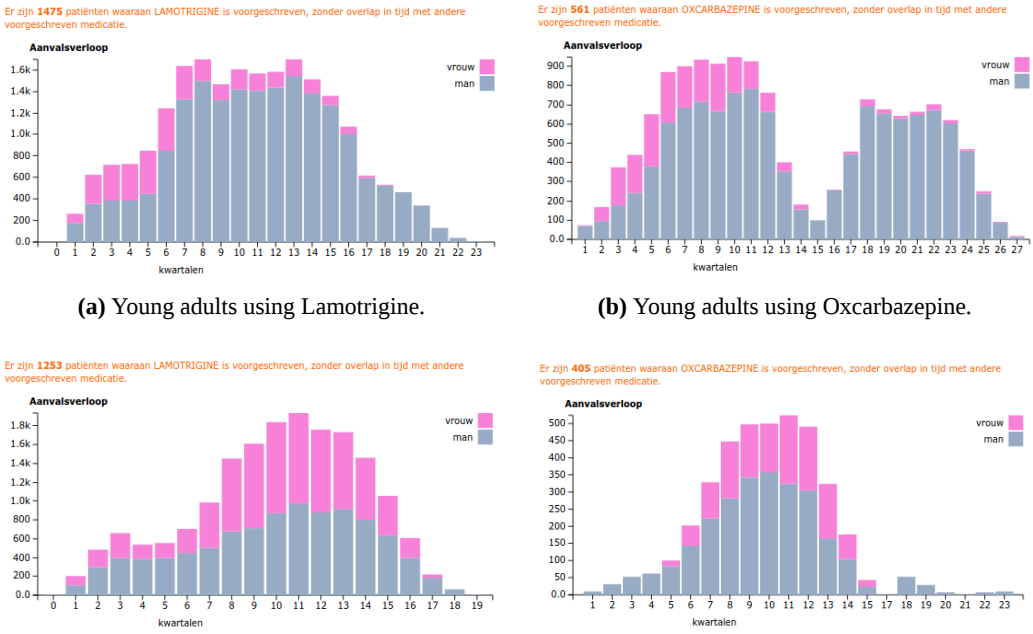
Figure 8.1: Graph specification.



(a) Selecting patients that have been prescribed "Lamotrigine" and "Oxcarbazapine". Note that the filter only includes patients for which an actual prescription is recorded, hence the lower bound of zero does not lead to inclusion of patients that have not been prescribed the substance.

(b) Selecting young adults for which the age is measured at the first time they had been prescribed "Lamotrigine" or "Oxcarbazapine".

Figure 8.2: Defining the filters. (a) Substance filter. (b) Age filter for young adults. Filtering out the middle aged patients is a similar case.



(a) Young adults using Lamotrigine.

(b) Young adults using Oxcarbazepine.

(c) Middle aged adults using Lamotrigine.

(d) Middle aged adults using Oxcarbazepine.

Figure 8.3: Seizure distributions for young patients (a)-(b) and middle aged patients (c)-(d).

PRET 2

Suppose that there was a protocol put into effect from the year 2000 onwards at the treatment location in 'HEEZE' that dictates that "Lamotrigine" can only be tried once on a patient. In addition "Oxcarbazapine" is supposed to be tried before "Lamotrigine" is tried. A researcher wants to find out the number of patients for which the protocol has been violated since the year 2000.

He sets up the DAG as shown in Fig. 8.4. Two temporal filters are added for checking the two conditions since the year 2000, see Fig. 8.5. The result of these filters are then merged together using a union filter. In total three patterns are required for the specification, one for the pattern filter node on top and two for the pattern filter node on the bottom. These are depicted in Fig. 8.6. This is because there are two cases of overlap that need to be included for the filter on the bottom.The final group consists of 2903 patients, for which one or both of the two conditions of the protocol have been violated since the year 2000. Finally, the researcher takes note of the demography of the resulting group, which is shown in Fig.8.7.

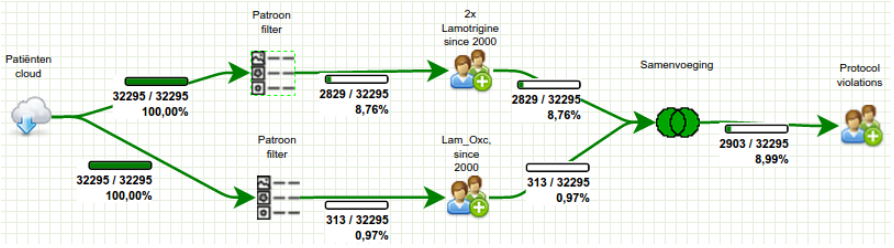
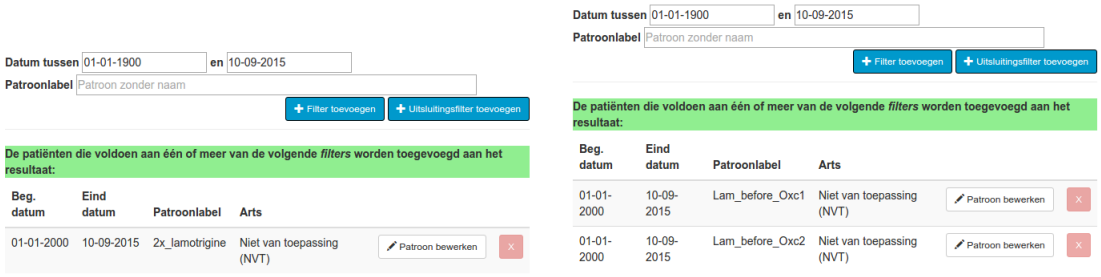
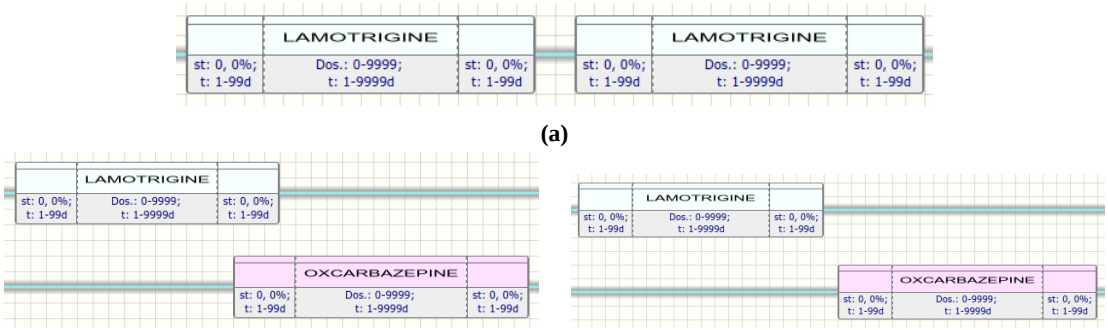


Figure 8.4: The DAG for PRET use case 2.



(a) **(b)**

Figure 8.5: Filter propeties for PRET use case 2. (a) Properties for the filter on top. (b) Properties for the filter on bottom. Two subpatterns are needed to cover the two cases overlap that are of interest.



(a)

(b) **(c)**

Figure 8.6: The patterns for PRET use case 2. (a) Pattern for the filter node on top. (b)-(c) Patterns for the filter node on the bottom, which filter two separate cases of overlap.

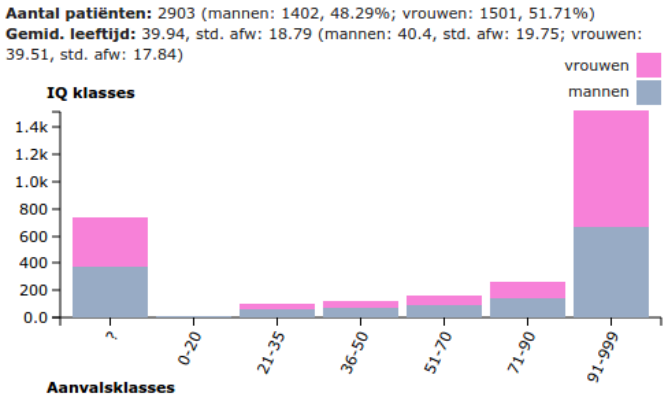


Figure 8.7: Demography for the resulting group of PRET use case 2.

8.2 Review of the research objectives

In this section we review the research objectives that are formulated in Section 3.3. For each objective we point out on what basis the objective has been fulfilled.

1. The development of an interactive visualization method for gaining a quick overview and insight into the EHR of any patient that:

   (a) shows graphical navigable display of medication over time,

   ✓*See the GEHR components in Fig. 5.1(b) and Fig. 5.1(c).*

   (b) including all (important) events, and,

   ✓*See the GEHR component in Fig. 5.1(c).*

   (c) offers the ability for searching on events using keywords.

   ✓*See the GEHR component in Fig. 5.1(g).*

2. The development of a method for comparing a patient (group) directly with another specified, possibly similar, patient group, such that:

   (a) the group definitions by *Paul van der Corput* are integrated,

   ✓*We have based our design around the perspective of a patient. The PRET user can specify freely the group (cohort) required, as shown in Fig. 6.1(a), Fig. 6.1(c) and Fig. 6.1(e). Some of the remaining definitions are slightly altered and put into a visualization, like for instance the retention time of a medicine in the form of a survival rate curve, see Fig. 6.16h.*

   (b) a way to investigate a single patient is included,

   ✓*The EHR of a single patient can be investigated using the GEHR (Fig. 5.1), which can act as a bridge between the GEHR and the PRET subsystems to compare a patient with a group of patients. For instance, the survival rate curve of a group can be used for direct comparison with a patient's retention time in the GEHR, as shown in Fig. 6.17.*

   (c) the statistical underpinning is (reasonably) considered, and,

   ✓*We have added the demography charts for the researchers, to assess the statistical similarities of groups, which are needed for good statistical practise, see Fig. 6.16a and Fig. 6.16b. We do think that future work is required to improve the (automated) support for this by the system, as described in Section 6.4.*

   (d) can we generalize the method, for use in other disciplines?

   ✓*Yes, the method of a DAG in combination with filter nodes and visualizations is domain independent, like the faceted search method. The filter nodes and visualizations must be designed in accordance with the domain ontology and entities.*

3. The augmentation of the previous method for temporal patterns for:

   (a) the specification of patient groups based on medication, and,

   ✓*For this we have developed the medication filter and the temporal pattern filter / editor. See Section 6.2 and Fig. 6.10.*

   (b) testing adherence to prescription protocols.

   ✓*This can be done using the PRET, using the temporal pattern filter, see Section 8.1.*

4. Contemplate and / or implement a bridge between the two methods above as an example of how techniques for SPS and MPS can be integrated

   ✓*This is covered in Section 6.6.*

8.3 User evaluation study

For a small user evaluation study two neurologists and one clinical researcher were gathered to test the system in its entirety. All participants were familiarized with the system before completing a series of tasks that involved utilizing most of the features the system has to offer.

8.3.1 Scope and setup

At the start, the test subjects were given a short overview of the purpose of the system, followed by a tutorial featuring the capabilities of the system. This familiarized the participants with the concepts and with the user interface of the system. Questions were answered in the meantime as well.
The specific features that were introduced during the tutorial for the GEHR are:

- The patient search box;
- The patient characteristics bar;
- The selection trees;
- The medication panel dosage charts, including blood compounds and adverse effects;
- All medication panel properties, except the option for prediction blood compound values;
- The seizures, weights and reports panels;
- The time period selector and drag bar;
- The selection details panel;
- The event search bar; and
- The pop-up chart for comparing a patient to a group was shown.

Next, the features introduced for the PRET are:

- The buttons for opening a new project, saving a project, running a project and zooming and the usage of mouse buttons in the graph editor;
- Selection modes and arrow connector modes;

- All filter nodes, with detailed examples of the gender filter, the substance filter, and the seizure filter;
- The pattern editor node was discussed with full specification of two medication subpatterns with a time period in between;
- The union node and the patient group node; and
- The demography chart and seizure distribution charts were shown.

It was made sure that the participants understood each step before progressing with the tutorial and that all aspects of it were registered.

8.3.2 Feedback and observations

After the tutorial the participants carried out tasks to answer a series of questions and provided feedback on the system. The used questionnaire is found in Appendix C. Apart from minor errors in the answers, like writing down a corresponding date when a value was asked, the questions were answered correctly.

GEHR

The feedback on the GEHR in general and its individual components are summarized in Table 8.1 and Table 8.2 respectively. Clearly, the feedback is positive and shows that systems are welcomed and valued by all participants. The time period selector was found to be the component of least usability, to which we add that the testing system experienced some latency when updating for a newly selected time period. This decreased usability mostly when the participants wanted to select a time period manually, as opposed to picking a predefined time period.

Participants were given the opportunity to share any additional feedback, which included:

- I was not waiting for such a system, but it is very welcome and handy.
- The font and the figures are a bit too small.
- I can easily search for reports on conducted research, but I also want to search the reports for the syndrome a patient was diagnosed with *(this data was not included in the dataset)*.
- The system is constructed logically.

During the session some observations were made. Participants really valued the idea of a single timeline on which all critical events were plotted simultaneously. Interestingly, the selection trees were used cunningly, whilst this possibility was not thoroughly demonstrated during the tutorial. This suggests that one of the main reasons for including the selection trees in the first place (Section 5.3), which is familiarity with scrolling through a list of events sorted on date, had paid of. Later on, when navigating the timeline, using a particularly *small* time span, the selection trees were effectively used in combination with the time period selector.

A comment on the reports chart was that there should be color coding based on all types of reports. We fully agree, but the reports chart was solely included to indicate a possibility in the prototype and therefore we limited support to only three types of reports: "RAPPORTAGE", "DECURSUS" and "ANDERS". We also want to add that only dummy data was provided for the reports and that if the number of reports for a patient is large, reusing the seizures chart for the reports is an option. The pop-up charts for comparing a patient with a group of patients seemed to impress the participants. Lastly, it was nice that some participants had also used the system afterwards for creating some figures for their own research and used

these as visualizations in a paper.

As was the case during the user study done by *Paul van der Corput*, most questions of the participants were simple questions about the user interface. For instance: where was that search bar again? It is reasonable to suggest that the initial confusion on UI details is quickly resolved after a little hands-on experience with the system. Although the users were capable of handling the time period selector after a short inspection, the user satisfaction remained only neutral. In line with the comments by *Paul van der Corput* on his time period selector, we also forced to question the intuitive quality of our time period selector. That is not to say that one should value less the concept of using a timeline in itself.

Statement	Strongly disagree	Dis-agree	Neut-ral	Agree	Strongly Agree
I was really waiting for such a system			1		2
The system was easy to use				2	1
The system enables me to quickly gain insight into an electronic health record of a patient				3	
It seems useful to me to continue using / researching with this system				2	1
I would like to use the system in the future				1	2

Table 8.1: Results user review of the GEHR in general.

Component	Not used	Very difficult	Diffi-cult	Neut-ral	Easy	Very easy
The time period selector				3		
The medication, blood compounds and adverse effects overview					2	1
The seizures overview					2	1
The weight and reports overview					2	1
The selection details panel					1	2
The keyword based search functionality				2		1

Table 8.2: Results user review on usage of the individual GEHR components.

PRET

All participants were capable of answering the questions using the PRET. Table 8.3 and Table 8.4 show the feedback on the system. Again the feedback is positive on the value of the system. The feedback on usability was more polarized, though still mostly positive. Consider first the additional feedback shared by the participants:

- I was (also) not really waiting for this system, but it is welcome and handy.
- This PRET was more difficult than the GEHR, but that is all right. Not all users need to use the PRET, because one can ask for help.

- It would be nice to expand the pattern filter later on.
- Once you get it, it works logically.
- It would be nice to filter on the syndrome a patient is diagnosed with.
- It would be nice to filter on seizure frequency per time interval. [This could be done in the form of an extension of the pattern filter].

After the sessions the participants were eager to express their strong interest in deploying the system during the upgrade phase of their workplace to a lab. They stated that this system should go beyond prototyping.

During the sessions it was observed that less questions about the UI were asked than for the first round with GEHR. Also, there was a was a distinct difference in the learning curve between participants with or without a background in mathematics and / or the workings of computer software. Yet, the basic concept of filtering in stages and working towards a resulting patient group that can be inspected seemed very intuitive.

In practice it may be beneficial to start with a small user base consisting of trained experts in clinical research that easily grasp the concepts the tool is build upon. Furthermore, features for exporting / importing patient groups are deemed necessary.

In conclusion we can say that the system was received very well for providing a valuable methodology. We take note that certainly more work is needed with clinical researchers to really battle test the system in the future, work on the obvious improvements and additions, and, keep an eye on the statistical implications and conditions.

Statement	Strongly disagree	Dis-agree	Neut-ral	Agree	Strongly Agree
I was really waiting for such a system			1		2
The system was easy to use			1	1	1
The system enables me to quickly gain insight into a patient group				2	1
It seems useful to me to continue using / researching with this system				1	2
I would like to use the system in the future					3

Table 8.3: Results user review of the PRET in general.

Component	Not used	Very difficult	Difficult	Neutral	Easy	Very easy
The chart drawing mechanism				1	1	1
The basic patient filters			1	1		1
The group operations	2					1
The pattern filter				1	1	1
The charts				1		2

Table 8.4: Results user review on usage of the individual PRET components.

8.4 Comparison with related work

We review the GEHR and the PRET according to the basic distinguishable features, user intent model and the analytic focus strategies, as described in Section 2.5 and Section 2.6.

The GEHR is capable of handling numerical and categorical data and shows five portlets having at least one variable. The medication portlet can show multiple medication variables, including blood compounds and adverse effects. Under the assumption that a couple of charts can be shown on an average screen, we estimate the number of variables it can show simultaneously around ten.

The PRET is also capable of handling numerical and categorical data and shows the statistics about the current flow of patients. In addition, there are two visualization panels that show one variable each. Therefore the systems supports up to three variables.

Extended comparisons using the user intent model and the analytic focus strategies are shown Fig. 8.8 and Table. 8.5. Using the user intent model, we see that the GEHR offers a decent number of features and that it is the only system for which an extra effort was made to avoid occlusion (with the flexible portlet setup). The GEHR, like many systems, incorporates a feature for temporal data binning, but keeps the abstractions to a minimum to preserve as much as the underlying distribution as possible. It is the only system that allows the user to compare a single patient directly with a group of patients, which can be specified dynamically (using the PRET).

A decent number of features is also supported by the PRET and if the mentioned extensions, mentioned in Section 6.5 are implemented, the PRET offers a large amount of features in comparison with other systems. Unlike most other systems, the PRET does not provide a means for navigating in time in a way that shows the values of (all) variable at a specific time or time period. It distinguishes itself by the more flexible way it allows the users to select a cohort of patients via a DAG with filter nodes. These filter nodes can, however, filter given an time window that is unrelated to the time window specified for other filter nodes. Cohort specification using a DAG in general offers more expressiveness than a faceted search mechanism, because subgroups can be used to define a group.

Temporal patterns can be used independently of other selection criteria and multiple alternative temporal patterns can be used simultaneously for specifying a group. In comparison with the faceted search, the advantage of using a DAG is that (intermediate) groups can be compared against each other. This leads to more efficient exploration, because when filters are tweaked, the changes propagate forward throughout the DAG, updating all related patients groups at once. The latter feature becomes more relevant if statistical aspects are integrated in the future and the researcher wants to investigate possible sources of bias. Also the checking of protocols can easily require multiple subspecifications that are more easily expressed in the DAG paradigm.

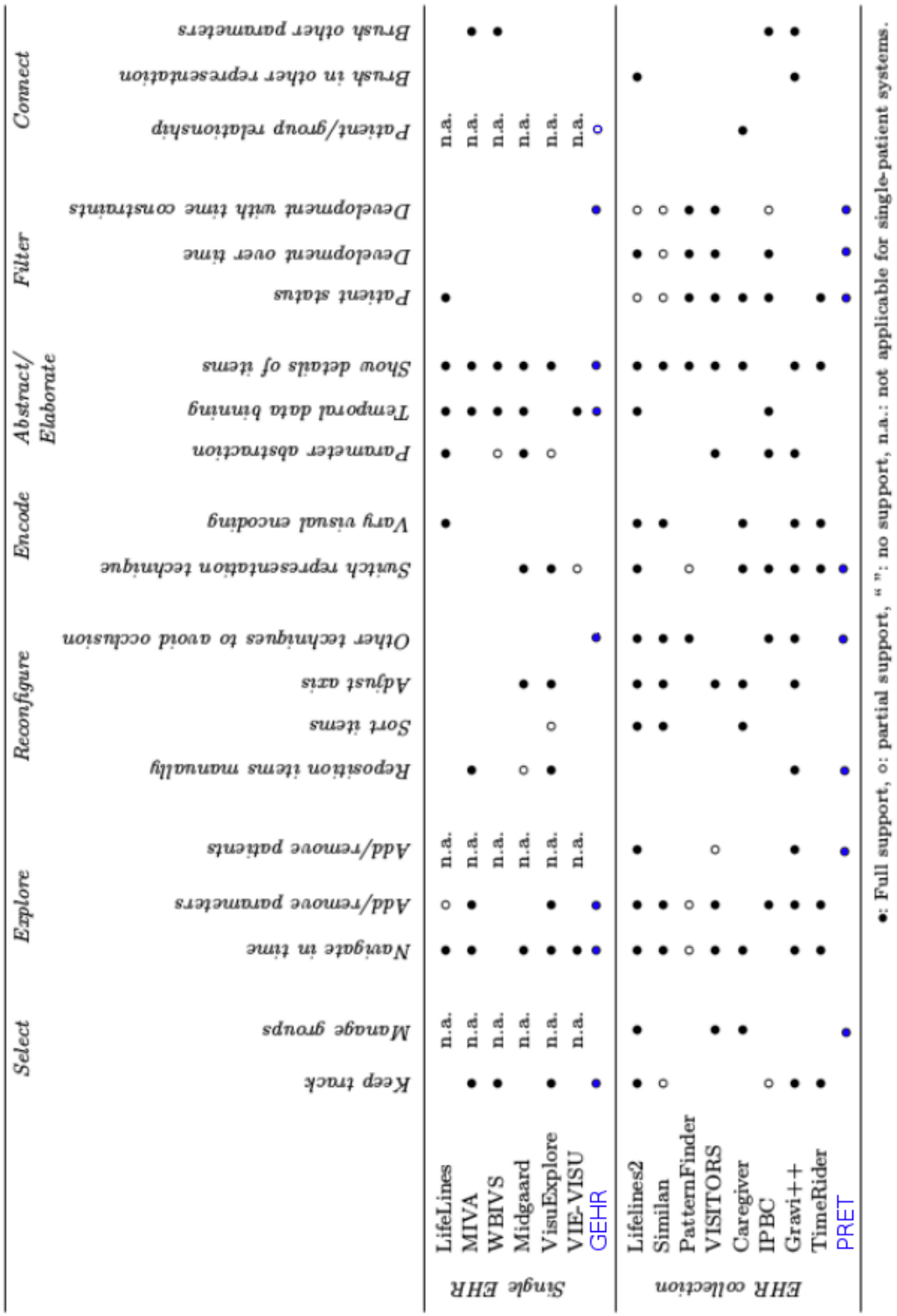


Figure 8.8: Extended comparison of the user intent model.

	3TV	LifeLines 2	LifeFlow	VISITORS	OutFlow	CAVA	EventFlow	PRET
Goal-driven record extraction	●	●	●	●	○	●		●
Goal-driven event extraction	●	●	●	●	●	●	●	●
Temporal windowing	●	●	●	●				●
Random sampling of records								
Temporal folding								
Grouping event categories (aggregation)				●			●	○
Selecting sentinel events in a stream	n.a.	●	●	●	○		○	
Converting multiple point events into a single interval event	n.a.		●	●			○	○
Converting multiple interval events into a longer interval event	●			○				
Identify hidden complex events	n.a.			●			○	○

Table 8.5: Estimation of support for analytic focus strategies by the MPS systems, based upon the descriptions of the systems. ●: Full support, ○: Partial or indirect support, " ": no support, n.a.: not applicable. The partly supported strategies are supported partly or indirectly using the filters.

Chapter 9

Conclusions

In this chapter we summarize the achievements and options for future work. Finally, we give some recommendations to the staff at Kempenhaeghe.

9.1 Achievements

We have presented our design of the prototype of a modernized version of the EHR system at Kempenhaeghe. Using interactive visualization techniques, the data of a single EHR can be investigated over time in one overview. The staff, who is used to working with an EHR system that mainly presents the data in tabular format, still has the ability to work in that fashion, but evaluated the new graphical format as highly advantageous. A main advantage is that the temporal aspect of the data is now easily interpretable, which allows for *visual comparison* of temporal variables. Additionally, a new search mechanism saves the staff a lot of time during the clinical interaction, because no more manual searching through the data is required.

Clinical researchers at Kempenhaeghe currently have to retrieve patient data from the database manually in spreadsheet format. All temporal aspects have to be reconstructed and / or calculated by hand. We improve this situation by offering a new system that features a graph drawing mechanism, where the nodes represents filters, to specify cohorts of patients interactively. Filters operate on a time window and filter out patients based upon variables in their EHR. A dedicated filter enables the filtering on event sequences as well, allowing the researcher to check for protocol compliance.

In comparison with the common faceted search method, our method offers more expressiveness, because a group of patients can be specified by including or excluding other specified subgroups. A change in a subspecification is propagated to all related groups, without the need to respecify all groups individually, and therefore, interactively comparing (intermediate) groups becomes more efficient.

9.2 Future work

For further development of the system, we propose the following points of interest:

- better integration of the substance types in the EHR overview;
- extend the research tool with more filters and event types for the pattern filter;

- improve the direct interpretability of the graph used for cohort specification and incorporate more graphical methods in the property panels for the filters;
- correctly integrate more statistics (see Section 6.4) in the visualizations of the PRET and add options for comparing multiple patient groups in a single chart, whereby the fixed color assigned to a group can be used to distinguish the group in the chart;
- automatically translate the EHR of a single patient to the research tool to start;
- integrate machine learning / probabilistic methods to support the selection specification of pattern by suggesting common patterns; and
- add more "soft" (subjective) data on patients to support (automated) methods for detecting bias in a cohort.

9.3 Recommendations

In order to fully support the future work, a better date structure is required for the database used at Kempenhaghe. For an example of such a system see *van der Linden et al.* [28]. Reconsider the way in which prescriptions are recorded. In that way, more efficient data retrieval might be possible when dealing with temporal questions. Also, make sure that a continuous prescription is stored as a single prescription with a series of prescription periods. Sometimes, a doctor starts a new prescription right after the previous one has ended, which degrades the quality of the data in the database. Finally, involve more clinical researchers in future development of the research tool.

Bibliography

[1] R. Bade, S. Schlechtweg, and S. Miksch. Connecting time-oriented data and information to a coherent interactive visualization. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '04, pages 105--112, New York, NY, USA, 2004. ACM. 7

[2] Dario Bonino, Fulvio Corno, and Laura Farinetti. Faset: A set theory model for faceted search. In *Proceedings of the 2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology - Volume 01*, WI-IAT '09, pages 474--481, Washington, DC, USA, 2009. IEEE Computer Society. 36

[3] M. Bostock. d3 data-driven documents. `http://d3js.org/`. Accessed: 2015-07-28. 77

[4] H. P. R. Boutsma. *New anti-epileptic drugs in pharmacoresistant epilepsy: retention time as outcome parameter*. PhD thesis, Maastricht University, 6 2009. 69

[5] J. Caban and D. Gotz. Visual analytics in healthcare – opportunities and research challenges. *Journal of the American Medical Informatics Association (JAMIA)*, 22(2), 2015. 3

[6] A. Chung. RequireJS JavaScript file and module loader. `http://requirejs.org/`. Accessed: 2015-07-28. 76

[7] P. N. A. van der Corput. Medicine Prescription Visualization. Master's thesis, Eindhoven University of Technology, 2013. 2, 13, 24

[8] A. R. Feinstein. *Clinical biostatistics*. The C.V. Mosby Company, USA, 1977. 71, 72

[9] Google. JavaScript mvw framework. `https://angularjs.org/`. Accessed: 2015-07-28. 76

[10] M.A. Harrower and C. A. Brewer. ColorBrewer.org: An online tool for selecting color schemes for maps. *The Cartographic Journal*, 40:27--37, 2003. 44

[11] K. Hinum, S. Miksch, W. Aigner, S. Ohmann, C. Popow, M. Pohl, and M. Rester. Gravi++: Interactive information visualization to explore highly structured temporal data. *j-jucs*, 11(11):1792--1805, nov 2005. 25, 31

[12] W. Horn, C. Popow, and L. Unterasinger. Support for fast comprehension of icu data: Visualization using metaphor graphics. *Artif. Intell. Med.*, 40(5):421--424, October 2001. 5

[13] jGraph. mxgraph javaScript graph visualization component. `https://www.jgraph.com/javascript-graph-visualization-library.html`. Accessed: 2015-07-28. 77

[14] J. Jin and P. Szekely. Interactive querying of temporal data using a comic strip metaphor. *Visual Analytics Science and Technology (VAST), 2010 IEEE Symposium on*, pages 163--170, October 2010. 35

[15] J. Keim, D. A. andKohlhammer, G. Ellis, and F. Mansmann, editors. *Mastering The Information Age - Solving Problems with Visual Analytics*. Eurographics, November 2010. 1

[16] D. Klimov, Y. Shahar, and M. Taieb-Maimon. Intelligent visualization and exploration of time-oriented data of multiple patients. *Artif. Intell. Med.*, 49(1):11--31, May 2010. 25, 28, 33, 42

[17] R. Kosara and S. Miksch. Visualization methods for data analysis and planning in medical applications. *International Journal of Medical Informatics*, 68(1-3):141--153, December 2002. 4, 5, 6, 7

[18] Microsoft. Health common user interface. `http://www.mscui.net/Components/`. Accessed: 2015-07-28. 19

[19] M. Monroe, R. Lan, J. Morales del Olmo, B. Shneiderman, C. Plaisant, and J. Millstein. The challenges of specifying intervals and absences in temporal queries: A graphical language approach. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '13, pages 2349--2358, New York, NY, USA, 2013. ACM. 33, 34

[20] The World Health Organization. The anatomical therapeutic chemical classification system with defined daily doses. `http://www.who.int/classifications/atcddd/en/`. Accessed: 2015-07-28. 15

[21] T. Otwell. Laravel php framework. `http://laravel.com/`. Accessed: 2015-07-28. 75

[22] C. Plaisant, B. Milash, A. Rose, S. Widoff, and B. Shneiderman. Lifelines: Visualizing personal histories. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '96, pages 221--227, New York, NY, USA, 1996. ACM. 19

[23] A. Rind, W. Aigner, S. Miksch, S. Wiltner, M. Pohl, F. Drexler, B. Neubauer, and N. Suchy. N.: Visually exploring multivariate trends in patient cohorts using animated scatter plots. In *In Ergonomics and Health Aspects, Proc. HCII 2011 (2011), LNCS 6779*, pages 139--148. Springer, 2011. 25, 32

[24] A. Rind, T.D. Wang, A. Wolfgang, S. Miksch, K. Wongsuphasawat, C. Plaisant, and B. Shneiderman. Interactive information visualization to explore and query electronic health records. *Found. Trends Hum.-Comput. Interact.*, 5(3):207--298, February 2013. 1, 3, 19, 21, 28, 33, 37, 38

[25] Y. Shahar, D. Goren-Bar, D. Boaz, and G. Tahan. Distributed, intelligent, interactive visualization and exploration of time-oriented clinical data and their abstractions. *Artif. Intell. Med.*, 38(2):115--135, October 2006. 19, 21

[26] B. Shneiderman and C. Plaisant. Sharpening analytic focus to cope with big data volume and variety. *IEEE Computer Graphics and Applications*, 35(3):10--14, 2015. 9

[27] M. Stonebraker. PostgreSQL open source database. `http://www.postgresql.org/`. Accessed: 2015-07-28. 75

[28] C.M.J. van der Linden, P.A.F. Jansen, R.J. van Marum, R.J.E. Grouls, T.C.G. Egberts, and E.H.M. Korsten. An electronic system to document reasons for medication discontinuation and to flag unwanted represcriptions in geriatric patients. *Drugs & Aging*, 29(12):957--962, 2012. 94

[29] World Wide Web Consortium (W3C). Extensible markup language (xml). http://www.w3.org/XML/. Accessed: 2015-07-28. 75

[30] World Wide Web Consortium (W3C). The web standards model. http://www.w3.org/wiki/The_web_standards_model_-_HTML_CSS_and_JavaScript. Accessed: 2015-07-28. 75

[31] T. D. Wang, C. Plaisant, A. J. Quinn, R. Stanchak, S. Murphy, and B. Shneiderman. Aligning temporal data by sentinel events: Discovering patterns in electronic health records. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '08, pages 457--466, New York, NY, USA, 2008. ACM. 9, 10

[32] T. D. Wang, C. Plaisant, B. Shneiderman, N. Spring, D. Roseman, G. Marchand, V.s Mukherjee, and M. S. Smith. Temporal summaries: Supporting temporal categorical searching, aggregation and comparison. *IEEE Trans. Vis. Comput. Graph.*, 15(6):1049--1056, 2009. 25, 26, 27

[33] Wesley Willett, Jeffrey Heer, and Maneesh Agrawala. Scented widgets: Improving navigation cues with embedded visualizations. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1129--1136, November 2007. 73

[34] K. Wongsuphasawat and D. H. Gotz. Outflow: Visualizing patient flow by symptoms and outcome. In *Proceedings of the IEEE VisWeek Workshop on Visual Analytics in Health Care*, pages 25--28, 2011. 25, 29

[35] K. Wongsuphasawat, J. Guerra G., Alexis, C. Plaisant, T. D. Wang, M. Taieb-Maimon, and B. Shneiderman. Lifeflow: Visualizing an overview of event sequences. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '11, pages 1747--1756, New York, NY, USA, 2011. ACM. 25, 27

[36] K. Wongsuphasawat and B. Shneiderman. Finding comparable temporal categorical records: A similarity measure with an interactive visualization. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology, IEEE VAST 2009, Atlantic City, New Jersey, USA, 11-16 October 2009, part of VisWeek 2009*, pages 27--34, 2009. 25

[37] J. S. Yi, Y. ah Kang, J. Stasko, and J. Jacko. Toward a deeper understanding of the role of interaction in information visualization. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1224--1231, November 2007. 8

[38] Z. Zhang, D. Gotz, and A. Perer. Iterative cohort analysis and exploration. *Information Visualization*, 2014. 25, 30

Appendix A

Detailed Entity Relationship Diagram

In Fig A.1 the tables for storing the data on the entities are depicted. In the database there were some auxiliary tables for storing user credentials the XML data for the PRET, but these are left out for clarity.

Figure A.1: The tables used for storing the entities in the database. Foreign key references are indicates by arrows.

Appendix B

Color codes for substances and body mechanisms

Figure B.1 shows the color codes the active substances and relates these to the different mechanisms of the bodies, on which the substance has an effect.

	INHIBITOIRE SYSTEEM						EXCITATOIRE SYSTEEM								
Anti-epilepticum	GABA systeem					K+ kanalen	Na+ kanalen		Ca2+ kanalen		Glutamaat		SV2A modulering	Carbo anhydrase remming	Multiple main mechanism
	Verlenging chloride kanaal opening	Toename frequentie chloride kanaal opening	Remming GABA transaminase	GABA reuptake remming	Anders	Kalium kanaal blokker	Snelle inactivering	Langzame inactivering	T-type	L-type	AMPA receptor antagonist	Anders			
acetazolamide														+++	
carbamazepine							+++								
clobazam		+++													
clonazepam		+++													
diazepam		+++													
eslicarbazepine							+++								
ethosuximide									+++						
felbamaat							+			+		++			X
fenobarbital	+++														
fenytoine							+++								
gabapentine										++ +					
lacosamide								+++							
lamotrigine							+++								
levetiracetam													+++		
oxcarbazepine							+++								
pregabaline										++ +					
perampanel											+++				
primidon	+++														
retigabine						+++									
rufinamide							+++								
sultiaam					+		+			+		+		+	X
stiripentol	+++		+++	+++											
tiagabine				+++											
valproinezuur			++				+		+			+			X
vigabatrine			+++												
zonisamide					+		+++		+					+	

Figure B.1: Color codes for the substances. The number of plus signs shows the relative impact of the substance on the mechanism. The main mechanism that is impacted is indicated with colored plus signs.

Appendix C

User Evaluation Questionnaire

The figures below show the four pages of the questionnaire.

User evaluation questionnaire

Part 1: Patient timeline

Please lookup the patient with code ***31255*** using the patient search bar before answering the qustions below.

Question 1. What is the minimum, maximum and average weight of the patient? When (approximately) was the maximum weight recorded?

Min. weight:	Max. weight:	Avg. Weight:	When max. weight:

Question 2. What was the dose distribution on day for the substance ***loreclezole*** initially, when the total dose was 75mg?

Question 3. Look at the medication overview and enable the showing of adverse effects and blood compounds. What is the last recorded adverse effect and who has reported the effect? What is the last known value for the blood compound, before the effect took place, for the related substance and who has reported the value?

Adverse effect:	Reported by:
Blood sample value:	Reported by:

Question 4. Which type of seizures have been reported between in 2009 between may the 5th and 9th and how many occurences are there per day? Make use the time selector with a small timespan.

Seizure type	May 5th	May 6th	May 7th	May 8th	May 9th

Question 5. When was the request for a wheel chair (keyword: 'rolstoel') reported? Make use of the searchbar on the top right of the screen.

Question 6. Select the pop-up chart for **carbamazepine** and select the ***Survival Rate Curve***. Pick the study called **'carbamazepine'** and group called **'carbamazepine'**. For how many days has **carbamazepine** been used by the patient.

Number of days:

Question 7. Please formulate a question for yourself. Does the system enable you to answer your question easily?

Figure C.1: Questionnaire page 1/4

Question 7. Please indicate to what extent you agree with the following statements.

Statement	Strongly disagree	Disagree	Neutral	Agree	Stronlgy Agree
I was really waiting for such a system	0	0	0	0	0
The system was easy to use	0	0	0	0	0
The systems enables me to quickly gain insight into an electronic health record of a patient	0	0	0	0	0
It seems useful to me to continue using / researching using this system	0	0	0	0	0
I would like to use the system in the future	0	0	0	0	0

Question 8: Please indicate your opinion on how easy it was to use the listed components of the sytem for selecting and / or intepreting the data.

Component	Not used	Very difficult	Difficult	Neutral	Easy	Very Easy
The time period selector	0	0	0	0	0	0
The medication, blood compounds and adverse effects overview	0	0	0	0	0	0
The seizures overview	0	0	0	0	0	0
The weight and reports overview	0	0	0	0	0	0
The selection details panel	0	0	0	0	0	0
The keyword based search functionality	0	0	0	0	0	0

Question 9: Please write down any additional feedback you would like to share.

Figure C.2: Questionnaire page 2/4

Part 2: Patient Research and Exploration Tool

Question 1. Select all patients that are threated in ***Heeze*** that have an ***IQ range*** that is either ***unknown*** or ***subnormal (70-90)***? How many male and female patients have an unknown IQ and how many have subnormal IQ? Make use of the ***threatment location selector, iq selector, the patient group and the demography chart***.

Unkown IQ males / females: /	Subnormal IQ males / females: /

Question 2. Open a new study. Split all patient on gender using a single ***gender filter***. Create two ***patient groups*** using a green and red arrow. Compare the ***medication top 10*** for males and females. What can be said about the usage of both ***Lamotrigine*** and ***Carbamazepine*** in both groups*?*

Males: Females:

Question 3. Select all patients that were in their ***80 (80-89)***, measured at the first prescription date of ***Lamotrigine***. Define a ***patient group*** and observe the ***Survival Rate Curve(s)*** for ***Lamotrigine.*** Draw a rough sketch of the curves.

Question 4. First filter all female patients and then add a **pattern filter**. Specify in the pattern filter that ***Oxcarbazepine*** was prescriped before ***Lamotrigine*** and that there is a time period of 1-30 days between the two. Define a ***patient group*** and chart the ***seizure distribution*** for ***Lamotrigine*** over time (using 'aanvalsverloop' chart). Please draw a rough sketch of the seizure distribution and state how many patients are involved.

Number of patients involved:

Question 5. If time alows for it, please formulate a question for yourself. Does the system enable you to answer your question easily?

Figure C.3: Questionnaire page 3/4

Question 6. Please indicate to what extent you agree with the following statements.

Statement	Strongly disagree	Disagree	Neutral	Agree	Stronlgy Agree
I was really waiting for such a system	0	0	0	0	0
The system was easy to use	0	0	0	0	0
The systems enables me to quickly gain insight into an electronic health record of a patient	0	0	0	0	0
It seems useful to me to continue using / researching using this system	0	0	0	0	0
I would like to use the system in the future	0	0	0	0	0

Question 7: Please indicate your opinion on how easy it was to use the listed components of the sytem for selecting and / or intepreting the data.

Component	Not used	Very difficult	Difficult	Neutral	Easy	Very Easy
The chart drawing mechanism	0	0	0	0	0	0
The basic patient filters	0	0	0	0	0	0
The group operations	0	0	0	0	0	0
The pattern filter	0	0	0	0	0	0
The charts	0	0	0	0	0	0

Question 8: Please write down any additional feedback you would like to share.

Thank you!

Figure C.4: Questionnaire page 4/4