University of Heidelberg
Faculty of Medical Informatics

Diploma's Thesis

Access, Handling and Visualization Tools for Multiple Data Types for Breast Cancer Decision Support

by

Christian Reichelt

Approved:

_____
Prof. Dr. Thomas Wetter, Thesis Adviser

_____
Angel Janevski, Thesis Adviser

Universität Heidelberg
Hochschule Heilbronn
Medizinische Informatik

Studiengang Medizinische Informatik
Masterstudiengang Informationsmanagement in der Medizin

..... …….....
(Name, Vorname) (Matrikelnummer)

Thema der Diplom-/Masterarbeit: .....

.....

.....

Ich erkläre hiermit an Eides Statt, dass ich die vorliegende Arbeit selbstständig und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe; die aus fremden Quellen (einschließlich elektronischer Quellen) direkt oder indirekt übernommenen Gedanken sind als solche kenntlich gemacht.
Bei der Auswahl und Auswertung des Materials sowie bei der Herstellung des Manuskripts habe ich Unterstützungsleistung von folgenden Personen erhalten:

.....

.....

.....

.....
Die Arbeit wurde bisher weder im Inland noch im Ausland in gleicher oder ähnlicher Form einer anderen Prüfungsbehörde vorgelegt und ist auch noch nicht veröffentlicht.

....
(Ort, Datum) (Unterschrift)

Abstract

Breast cancer is the most commonly diagnosed cancer among U.S women, besides skin cancer. More than 1 in 4 cancers among women are breast cancer. And though death rates have been decreasing since 1990, about 40,170 women in the U.S. were expected to die in 2009 from breast cancer.

The progress of molecular profiling, in the last decade has revolutionized the understanding of cancer, but also introduced more complexity with new data such as gene expression, copy number variation, mutations and DNA methylation. These new data open up the possibility of differential diagnosis, much more precise prognosis as well as prediction of therapy response than any of the diagnostic tools that are available in the current practice. Additionally, epidemiological databases store clinically relevant information on hundreds of thousands of patients. However, with the abundance of all this information, clinicians will need new tools to access and visualize such data and use the information gained to treat new patients. The general problem will be to access, filter and analyze the data and then visualize them in a clinical context. This data ranges from clinico-pathological information, to molecular profiles from high-throughput genomic measurements and imaging data. Furthermore, data from patient populations is aggregated on epidemiological level and can be found under numerous clinical studies.

The goal of this masters' thesis is to develop tools that enable handling, combining and visualizing multiple molecular and epidemiological data types in the context of a clinical application for breast cancer and include them in a demonstrator that will showcase the approach.

This platform will cover several classes of tools: 1) Tools for visualizing data from different molecular modalities 2) Visualizing epidemiological data, 3) Case based reasoning that will compare the data of a current patient with a collection of data from previous cases or analyze the specific data of a patient with the aid of combining available multiple data types. 4) Filtering and organizing tools for clinical trials, which present the clinical trials relevant to the patient under treatment clearly and in a more user-intuitive manner. Additionally, we will explore ways to generalize these tools to datasets from ovarian cancer and possibly other diseases.

Acknowledgments

I want to begin by thanking my advisor at the University of Heidelberg Prof. Dr. rer. nat. Thomas Wetter. This thesis would not have been possible unless he initiated the contact to Philips Research.

I would like to thank PhD Charles Lagor who was my first counterpart at Philips and helped to manage my stay in the United States.

A very special thank goes to my advisor Angel Janevski and PhD Sid Kamalakaran for their guidance and support. They were always ready to help and to support my work with an incredible patience. They helped to make this thesis to what it is now.

Finally I would like to thank my colleagues at Philips Research for their help and understanding during my work on the thesis and all other people who made this six month in the USA unforgettable.

Table of Contents

List of Figures

List of Tables

Chapter One: Introduction

In time of rising data transfer rates, increasing storage capabilities and generation of new data from sensors, computers, cameras, phones, etc. more and more data is produced and needs to be handled. This trend is not limited to one specific field, rather it ranges from the personal data such as picture and music collections, up to the huge amounts of data generated by enterprises.

Companies like the retail giant Wal-Mart handle more than one million customer transactions every hour, feeding databases estimated at more than 2.5 petabytes. The social-networking website Facebook is home to 40 billion photos its members uploaded. Furthermore, research facilities like CERN, Europe's particle-physics laboratory, during their experiments at their Large Hadron Collider, generates 40 terabytes of data every second.

This data offers great new opportunities, but it also leads to new challenges in making this data useable.

1.1. Data Overload in Medicine

In the last few decades, there have been rapid strides in medicine with the invention of new technologies and a greater understanding of human biology. New diseases and challenges in managing existing diseases have driven a continuous development of new diagnostic and therapy methods taking advantage of developments in all branches of science and technology. However, the use of advanced technologies is also changing clinical practice. The paper-based files used to document patient medical data have given way to digitally recorded and centralized electronic health record (eHR), easily accessible from everywhere. We have progressed from imaging data developed from single-layered X-ray images to digital computerized tomography (CT) and magnetic resonance imaging (MRT) that create three-dimensional views of the human body. More effective methods and higher resolutions are making these images increasingly detailed.

New developments in the clinic have also been driven by advances in high-resolution molecular profiling. The human genome was first sequenced in 2001, new high-throughput technologies were developed to profile DNA, RNA and epigenomic profiles of samples, all which created a new opportunity to understand the human organism in a completely new way. In the last decades these developments have lead towards development of new molecular diagnostic tools, which have already advanced our understanding of cancer and other diseases.

However, even a basic output of sequencing of a personal genome already contains ca. three billion base-pairs of the human DNA. And this data is extended by additional technologies like the molecular profiling, mentioned before. This adds new data types like gene expression, copy number variation, mutations and DNA methylation to the already available data used in the medicine.

All of these technologies are creating huge amounts of new and complex data that carries great potential. They provide possibilities of differential diagnosis, accurate prognosis as well as prediction of therapy responses than any of the diagnostic tools that were available before. But they also create new challenges like the accessing and handling of this data that need to be solved

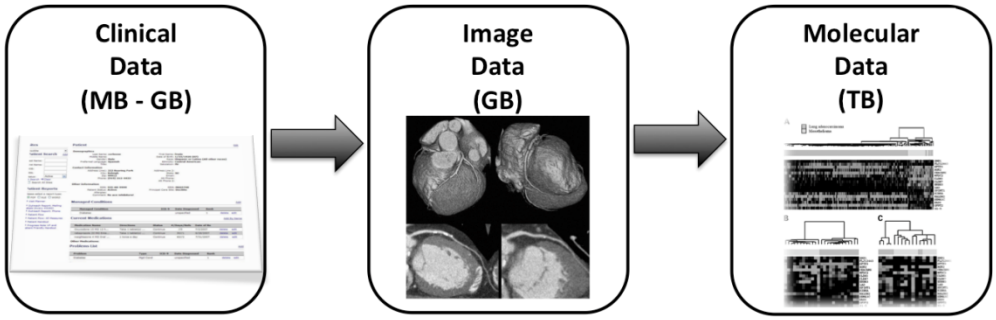


Figure 1: Growing sizes of medical data.

1.2. Data Visualization as a Solution

„A picture is worth a thousand words."

This phrase illustrates the advantages of a common method of information processing that humans have used time and time again - „information visualization". Maps from ancient Egypt, the geometry diagrams of Euclid and the statistical diagrams of Playfair are only a few examples in which information is presented in a visual way to support its understanding by the human mind. The idea behind this technique may have remained the same, but with the development of new technologies, the art of visualization has been developing continuously. They are no longer static pictures of a single data set. Today visualizations can be prepared automatically at time of use, can be made dynamic and interactive, and can be integrated into a larger process of decision making.

Questions that are too difficult to be answered purely mentally can be analyzed and solved with the help of visualizations. The use of rapid access to large amounts of data, and the fact that they analyze and process data faster and with accuracy higher than humans could ever do, make them to a technology with a great potential in handling the data overload of our time.

Visualizations follow a basic processing of data consisting of four parts, shown in Figure 2:

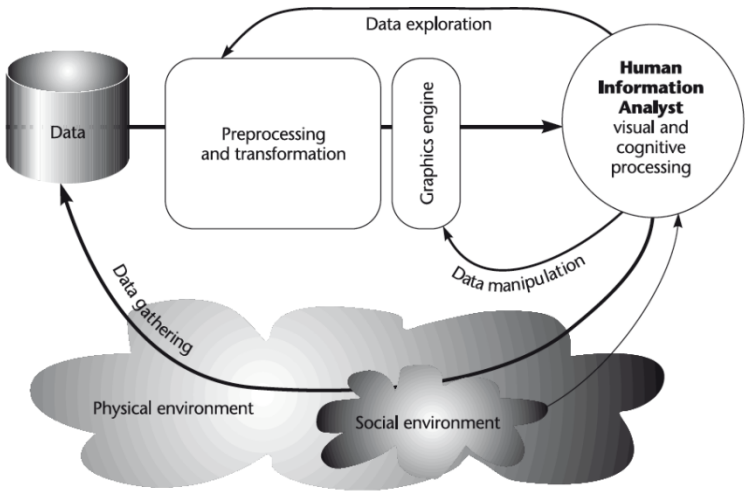


Figure 2: The process of data visualization in four basic parts.

- The data that includes the knowledge of the visualization
- A preprocessing and transformation part that transformers the data into a form the visualization can display
- The visualization itself that presents the processed data and represents the interface to the human
- The human that analyzes and processes the cognitive information.

1.3. Project Context

This thesis is the result of a collaboration between the University of Heidelberg, Germany and Philips Research North America. It originated as part of a research project of the department of Ultrasound, Photonics and Bioinformatics of Philips Research working in the area of clinical decision support. This team has focused on analysis and clinical application of molecular profiling and other data in the context of breast and ovarian cancer. Their experience with a variety of data types and the implication of their analysis in the clinic was the base from which the topic for this thesis emerged: *„Design and implementation of prototypes that use visualization technologies to access and handle multiple medical data types for breast cancer decision support.“*

In more detail, we used different visualization methods and technologies like maps and charts to create self-contained tools that can present a multiplicity of medical data types. Each tool also provides interfaces that make the communication and interaction between them possible. To show the benefits of those interactive tools, we included them in a demonstrator that showcases this approach.

The thesis includes a theoretical part as well as a practical part.

In chapter 2 we evaluate data overload in medicine based on the available literature on the subject. In addition, we interviewed clinicians from the University of Heidelberg to obtain better understanding of the needs and requirements for data visualization in oncology.

Chapter 3 describes the requirements for the project as well as the technologies and architecture we're using for the development.

Chapter 4 describes the actual development process in detail. It describes the motivation and specifications as well as the implementation and the issues and challenges for each tool.

In chapter 5 we evaluate our results by presenting the demonstrator to the same clinicians we interviewed earlier to obtain feedback on the implemented prototype. As a follow up evaluation step we describe two new tools, which were build on the components developed in the course of the thesis.

In chapter 6 and chapter 7 we discuss the advantages and disadvantages we identified in this thesis and end with a conclusion.

Chapter Two: Methods

Data overload has been discussed in industry as well as in medicine. There are many publications written about the problem of data/information overload and equally as many publications proposing solutions to handle this overload. Here we present an overview of the literature we used to understand data overload in medicine and about the idea of visualizations as a solution of this problem. Furthermore, we describe interviews with physicians, which provided information about the requirements and the needs they have and to validate the ideas we obtained from the literature. The result of this was a number of use-cases for several data types, which served as templates for the later implementation.

2.1. Literature Survey

The focus of this thesis covers two specific fields, data overload in medicine, especially in breast cancer treatment, and data visualization as one way to handle this overload. We begin by reviewing the extensive literature that has been written about data overload in clinical field. This is needed to evaluate the approach of data overload in medicine and also to find medical datatypes that afflicted by this overload. We then turn to the literature and research that has been written about visualization technologies. The goal of reviewing the literature is define the scope of work: the relevant medical datatypes, the ideas, and the concepts of visualization.

The first question we have to be clear about is: „Do we have medical data types that consists of huge amounts of data that need to be supported by computer systems?“ As a starting point the book „Biomedical Informatics for Cancer Research“ from M.F. Ochs et al.[1] reviews the work from a number of researchers who have produced open source software addressing the need for data management, integration, analysis, and visualization to aid cancer research. It provides a good overview of the use of data in cancer research. It highlights the advent in genomics and clinical trials as main producer of clinical data in the last decades. Molecular data is defined as one of the most important data source in the future which will revolutionize the treatment of cancer patients [2; 3]. Other projects already tried to use this new type of data to create decision support systems like the Caleydo framework [4].

Another type of data that use for decision support is underestimated is the epidemiological data. This data is stored in large epidemiological databases like the Surveillance Epidemiology and End Results (SEER) database [5] which currently contains over 800.000 records. Only a handful of tools have used this data, which actually has high potential in supporting physicians with statistical information. One of these tools is Adjuvant! Online [6], which utilized the SEER database to help physicians and patients to select therapy and understand the risks and benefits of getting additional therapies after surgery.

A typical data source that is used in the clinical work all the time is the clinical literature in form of books and articles. But the amount of new publications is increasing very rapidly. Databases

like the PubMed database [7] contain references to more than 16 million articles in some 4800 biomedical journals [8]. To access this information effectively, new clinical tools are necessary.
The four data types mentioned in the literature before have similar characteristics: they are stored in large databases, several data sources are available for every data type, each of the data types could benefit from visualizations and in this way avoid the increasing of the data overload in the clinic. For that reason we decided to focus our efforts in finding visual solutions for these types (overview in Table 1).

Data Type	Clinical Need	Use Example
Epidemiological Data	Population based Studies, Population Statistics, Decision making tools	SEER Database is used to drive „Adjuvant! Online“, widely used tool in Breast Cancer Oncology
Molecular Data	Molecular Profiling	CGI - a 97-gene measure of histological tumor grade
Clinical Trials Information	Trials as basic mechanism of discovery in clinic	Therapy Decisions, Enroll patients in trials
Clinical Literature	Dissemination of Knowledge	Cancer Biology, Drug Information, Study Reports, etc.

Table 1: Data types and their use in medicine.

Before creating data visualizations, we need to understand the types of visualizations in existence and the information they can present. David McCandless provides a good first overview about available visualizations in his book „The Visual Miscellaneum: A Colorful Guide to the World's Most Consequential Trivia“[9], in which he uses graphs, charts and illustrations to visualize relationships and compelling data. A deeper insight is given by Chun-houh Chen’s „Handbook of Data Visualization“[10] and Julie Steeles’ „Beautiful Visualization: Looking at Data through the Eyes of Experts“[11], that provides additional types of visualizations. They supplement their work with historical information and facts about what is needed to make a good visualization as well as the differences between them.
During this search other projects were found, which are discovering the possibilities of visualizations to present medical data, [12-14]. All of them use different visualization types in different medical fields to present medical data.

2.2. Design Use-Cases

Based on the information about the data types and the visualization techniques collected from the literature, we created use-cases for a number of clinical scenarios. In this chapter the different use-cases are presented and explained in detail.

2.2.1. Epidemiological Data

The epidemiological data is used as input to two different tool types. In the first tool statistical information about the occurrence of cancer in the population are shown in form of pie charts. These charts display for example the distribution of breast cancer patients over different age groups or stages. The data of the current patient is automatically highlighted in the particular chart.

The second tool uses the data to show survival curves of selected population groups. In this way the effects of different therapies, the advancing ages or other attributes can be compared.

To create an intuitive interface the pie charts of the first tool are used as selection panel for the survival tool. By selecting different slices of the charts a query can be generated which can be used to specify the population group used to create the curves. The design of this use-case is shown in Figure 3.

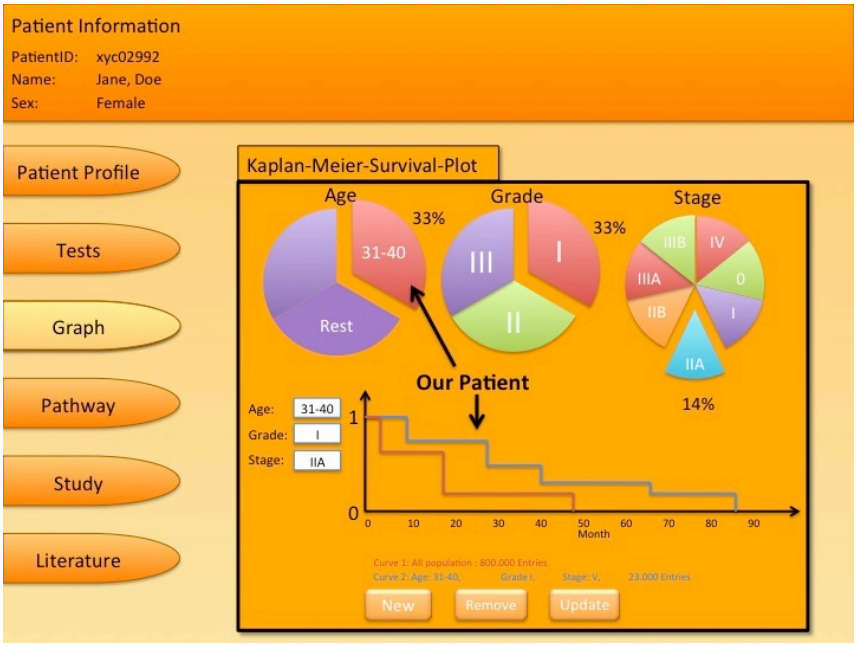


Figure 3: Use-case for the epidemiological data.

2.2.2. Molecular Data

The molecular data is used to present biological pathways, which are relevant to a patient. A pathway can be selected and is shown in its original version on the left side of the screen. This pathway now can be analyzed, which means the patients molecular data is compared with the original pathway data and presented on the right side of the screen. This modified pathway will highlight any differences from the original pathway by changing the color of missing or additional molecules. If a molecule is missing the effects are shown at the connections to other molecules. In this case the lines that connect the molecules are drawn thinner or thicker and in red or green depending whether they are inhibiting or activating. Additionally there is a perturbation score that shows the significance of the pathway deregulation. The design of this use-case is shown in Figure 4.

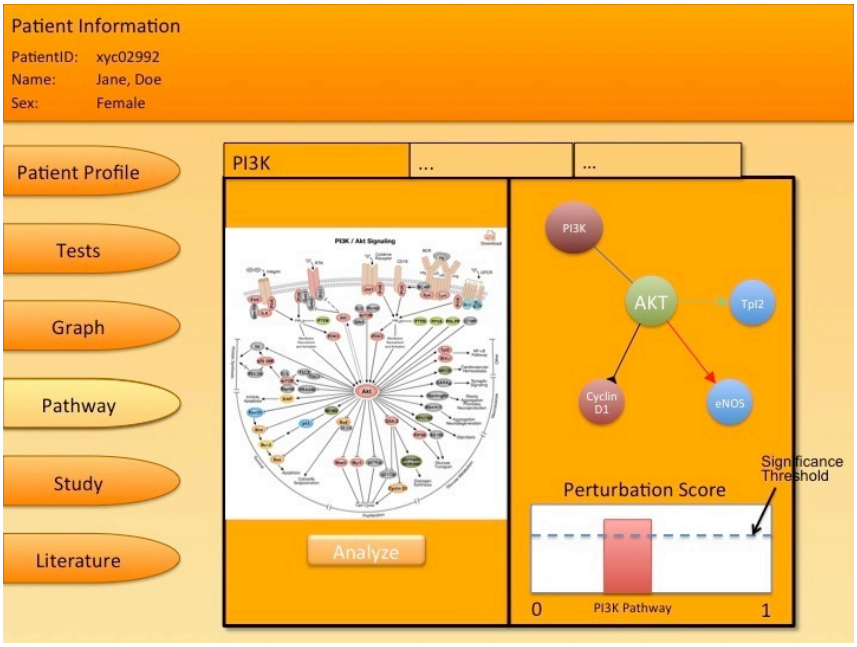


Figure 4: Use-case for the molecular data.

2.2.3. Clinical Trials

The data from the clinical trials is used in a search tool that positions the locations of the particular trial on a geographical map. It's possible to filter the results by specifying the state and country in which the trial should be and by choosing the status open or closed. For each result the tool creates a marker on the map which includes the information of the trial like title, description and so on. The map itself contains control elements to move around and to zoom in and out. The design of this use-case is shown in Figure 5.

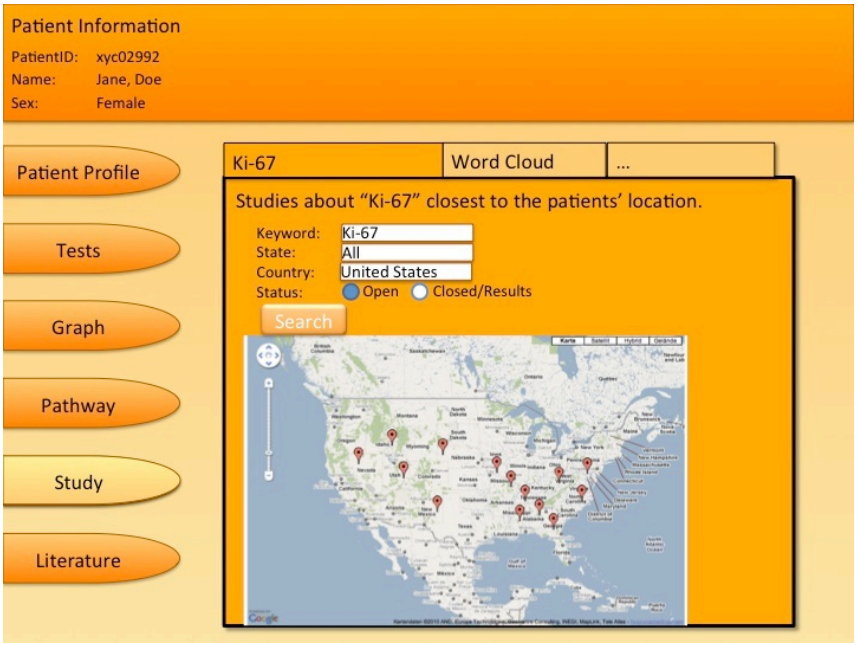


Figure 5: Use-case for the clinical trial data.

2.2.4. Literature

Medical literature serves as input for an interactive search tool. The tool offers several filter options to specify the search such as keywords, publication language, etc. The literature search can be accessed from any other tool and would start the search automatically if a start value is transmitted.

Furthermore the results of the literature search can be used as input for a summarizing tool that presents the keywords in form of a Word Cloud. In this visualization type the keywords are shown in different font sizes depending on their occurrence in the text. The higher the occurrence the larger the font size. The Word Cloud is also interactive so that the user can click the shown words to start automatically a new search. The designs for the two use-cases are shown in Figure 6 and 7.

Figure 6: Use-case for the literature search.

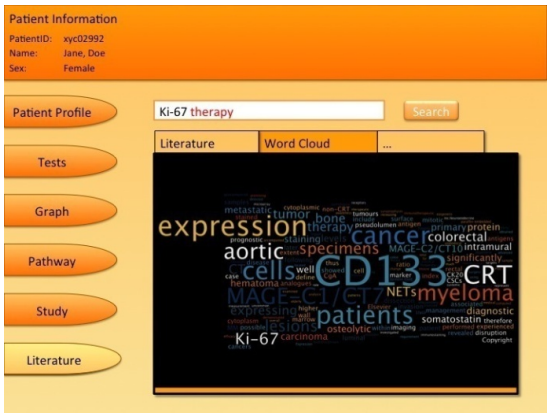


Figure 7: Use-case for the word cloud.

2.3. Interviews with Clinicians

The cooperation between clinicians and software developers is essential in the development process for useful and intuitive applications. Without including the end-user into the development process it's almost impossible to create software that fits their requirements and needs. Providing solutions that meet the user requirements is the key to deliver a useful product that will be accepted. For that reason, we contacted oncologists who were interested in the problems our project addressed. Three oncologists of the University Hospital of Heidelberg kindly agreed to talk to us.
These are:

- Dr. Clemens Stockklausner, physician at the Pediatric Clinic of the University Hospital of Heidelberg, working in the field of children oncology.
- Dr. Jörg Heil, assistant physician at the Women Hospital of the University Hospital of Heidelberg, working in the fields of mamma carcinoma, diagnostic, surgery and system therapy.
- Dr. Florian Schütz, senior physician at the Women Hospital Salem that is in cooperation with the University Hospital of Heidelberg. He is the vice-chairman of the hospital and reasonable for the whole field of the gynecology and perinatology working in the medical fields of the gynecological oncology and perinatal medicine.

The interviews were divided into three parts, a general part, a prototype presentation and a discussion.
The general part contained questions about the data types their using in their daily workflow, tools their already using and wishes they would have for new software.
After these general questions we presented three PowerPoint scenarios of possible workflows to them. These scenarios were built from the use-cases described in chapter 2.2 and show possible processes of the tools and the transitions between them. They served us to explain the physicians our ideas and to get feedback what is good or what they would change. The complete scenarios are presented in Appendix A.
Finally, we asked them about their impressions and what they think about our concepts in general. (A complete flow of the interview is given in Figure 8.)
All physicians agreed with the statements of very fast growing data in medicine and with very high potential in software that supports them with a better use of this data. Currently the only software that finds use in the work of the physicians, is the decision support tool Adjuvant! Online. They agree with the data types discussed in the literature, but they were divided in their wishes for the presented tools. They see the use of some of the tools not only in decision support for themselves, but more in helping their patients to understand what choices they have and what the consequences are. Another important field the tools could be helpful is in scientific research. Access to published literature for example doesn't find the biggest use in patient care, because

therapies have to follow strict guidelines, but in answering scientific questions or supporting the physician with the newest studies helping him to stay up to date.
In the following section we summarize the physicians opinions and suggestions to each of the tools in the scenarios.

Visual Query Builder in combination with Survival Curve Manager

- Intuitive, fast and easy to use
- Useful for physicians with less experiences, given that experienced physicians are intimately familiar with these statistics.
- Particular interesting if statistics incorporate different therapy choices.

Biological Pathway Visualizer

- Big potential and will be very important in future.
- Very future-oriented. Will definitely come but may take another several years for such data to be used routinely in the clinic.

Geographical Trial Finder

- May not be very useful for physicians in Germany, because usually physicians typically refer to their own studies or the studies of their colleagues.
- Bigger use for patients who are interested in what and where studies are available.

Literature Search and Word Cloud

- Usually, literature searches aren‘t used in patient care. Exceptions are very uncommon cases.
- High potential in scientific research
- Good filtering options, but would need additional options, e.g. Patient age, gender, etc.
- Text Summarizing tools like the Word Cloud are nice gadgets but may not be practical for use in patient care. Possibly use as an quick overview in a research application.

In the last part of the interviews we asked the physicians about their general impressions and feelings of these tools.
They like the way of visualizing large amounts of data and see high potential in these types of tools. They also appreciated the automatic functions, the interaction between the tools and the intuitive user interface seems to be helpful in making their work easier.
Transcripts of the complete interviews are in Appendix B.

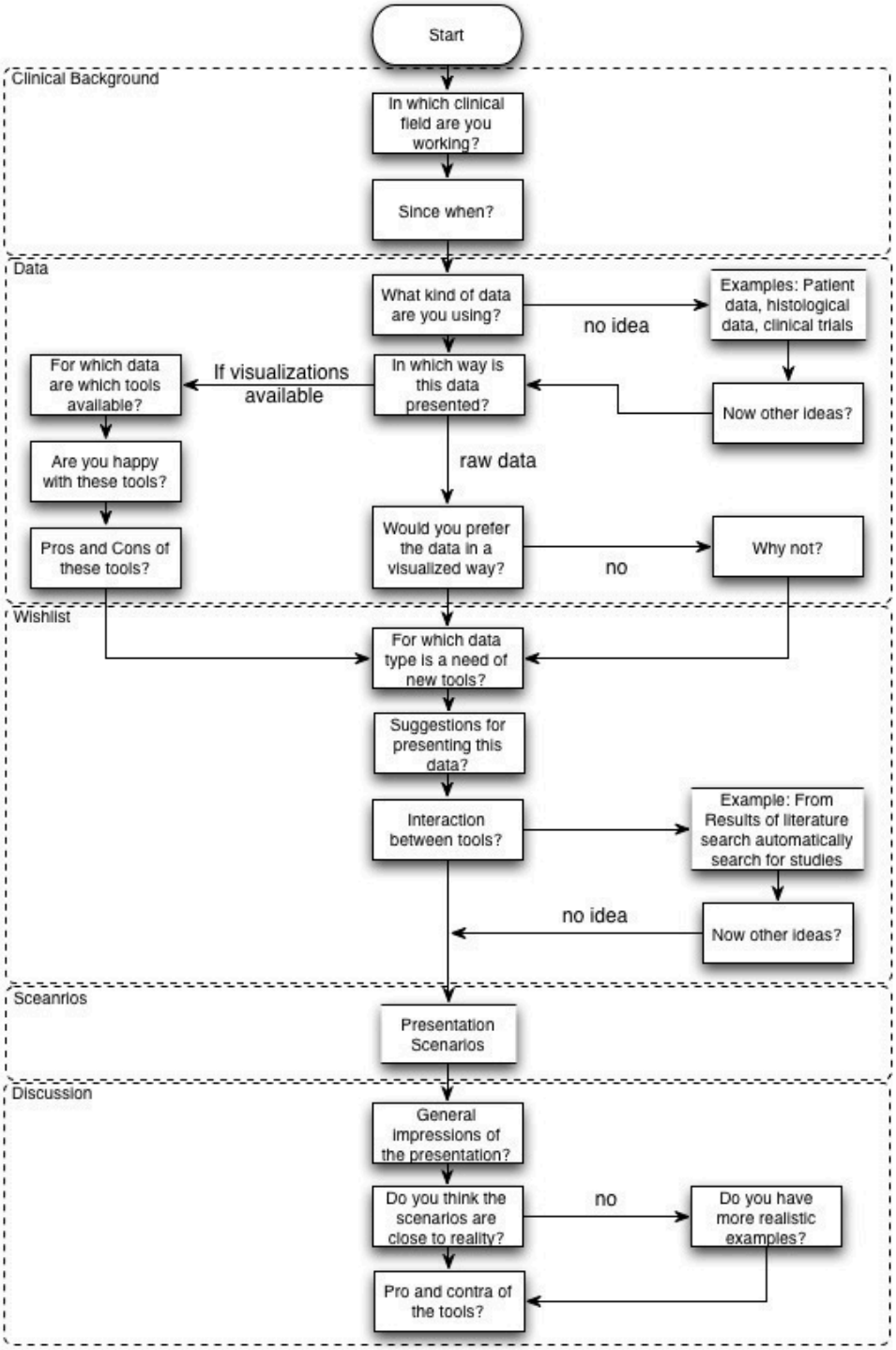


Figure 8: The flow of the interview.

Chapter Three: Technical Details

3.1. Requirements

The requirements for this project were built out of the literature research and the interviews. They combine the wishes of the physicians as well as the technological features we want to include. The key requirements for our prototype were:

- Accessible from everywhere
- Intuitive User Interface
- Consistent User Interface. UI elements should be ordered in the same way in every tool. This guarantees a high usability.
- Easy integration into other applications
- Modular expandable, easily changeable. It should be simple to integrate tools into the platform as well as it should be possible to change or remove tools that aren't needed.
- Performance, very large datasets are used.
- Minimal system restrictions, software must be fully working on any available systems.

3.2. Technologies

The selection of the right technology for the task is one of the critical points in software engineering. The complete project depends on the technology that is used. The technology defines what is realizable and what isn't.

After discussing the requirements of Chapter 3.1 we decided to develop our tools in form of a web platform. Web Platforms have several advantages to typical Desktop Applications. They are accessible from every system with internet access, running out of standard web browsers which available on every computer. Typically, web-based applications have almost no system restrictions.

Therefore, we needed a web technology that will provide the functionalities suitable based on the requirements, to receive data from databases and other web services and for the creation of visualizations. In the next sections we're introducing four modern web technologies, compare them and elaborate on the rationale for our choice.

3.2.1. HTML5

The World Wide Web's markup language has ever been HTML (HyperText Markup Language). It was primarily designed to describe scientific documents, but over the years the language has continued to develop and learned to describe a lot of new types of documents. The main area that has not been adequately explored is the area around the so called „Web Applications". HTML5 [15], the newest major revision, attempts to rectify this by developing new specifications. In particular, HTML5 adds many new features like <video>, <audio> and <canvas> elements which are designed to improve the inclusion and handling of multimedia and graphic content as well as other new elements such as <section>, <article>, and <header> that improve the semantic richness of documents. At the moment HTML5 is still under development, what means that still not all functions are fully available and not all bugs are fixed. But the progress is very fast and also appearing problems are fixed very quickly.

In this project we're interested in the <canvas> element, since it provides the possibility for dynamic, scriptable rendering of 2D Shapes and bitmap images. The canvas element is a drawable region defined in HTML code with a height and width. It can be placed at any point in a HTML document. JavaScript provides functions to get access to this canvas element and also provides a full set of drawing functions similar to other common 2D APIs, which allows generating dynamical graphics. Canvas only supports one primitive shape - rectangles. All other shapes must be created by combining one or more paths. There is a collection of path drawing functions which make it possible to compose very complex shapes, but the effort to create such a shape is very high.

Figure 9 shows a simple graphic which is created by using paths. This example also shows that the complete coding is done in JavaScript and is only be drawn into the canvas element.

Code:

```
<html>
 <head>
  <script type="application/javascript">
   function draw() {
     var canvas = document.getElementById("canvas");
     if (canvas.getContext) {
       var ctx = canvas.getContext("2d");
       ctx.beginPath();
       ctx.arc(75,75,50,0,Math.PI*2,true); // Outer circle
       ctx.moveTo(110,75);
       ctx.arc(75,75,35,0,Math.PI,false);   // Mouth (clockwise)
       ctx.moveTo(65,65);
       ctx.arc(60,65,5,0,Math.PI*2,true);  // Left eye
       ctx.moveTo(95,65);
       ctx.arc(90,65,5,0,Math.PI*2,true);  // Right eye
       ctx.stroke();
     }
  }
  </script>
 </head>
 <body onload="draw();">
   <canvas id="canvas" width="150" height="150"></canvas>
 </body>
</html>
```

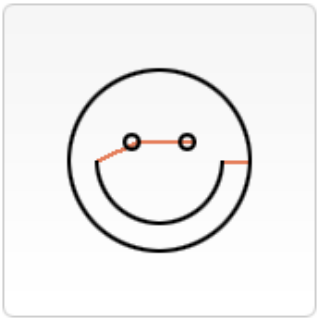


Figure 9: Simple Graphic generated with paths.

Advantages

- Minimal learning curve since most people already know the basics of HTML
- Large pool of developers
- Supported natively by the web browser (Currently not all functions are supported in every browser)
- Does not require any installation of third party software

Disadvantages

- No predefined widgets like panels or dialogue boxes.
- Javascript underlies the Same Origin Policy (SOP). This policy permits scripts running on pages originating from the same site, but prevents access to methods and properties across pages on different sites. Makes the access to external data very complicated

3.2.2. Protovis

Protovis [16] is a free and open-source project of the Stanford Visualization Group. It's a very powerful graphical toolkit that uses JavaScript and SVG to build web-native visualizations. Part of what makes it special is that, because it's written in JavaScript, don't need any plugin to run in the browser. Protovis is especially developed for designing visualizations. In this way, it uses some of the simplicities and low-level controls of graphical systems by dealing directly with graphical elements but specifies marks declaratively as encodings of data. Figure 10. shows this way of declarative programming and how easy it is to create an attractive chart. In the paper „Protovis: A Graphical Toolkit for Visualization", Michael Bostock and Jeffrey Heer [17] write in more detail about the motivation and idea behind Protovis and the benefits of Protovis compared to other graphical toolkits.

Code:

```
var vis = new pv.Panel().width(150).height(140)
    .add(pv.Bar)
      .data([1, 1.2, 1.7, 1.5, .7, .2])
      .bottom(0).width(20)
      .height(function(d) d * 80)
      .left(function() this.index * 25)
    .root.render();
```

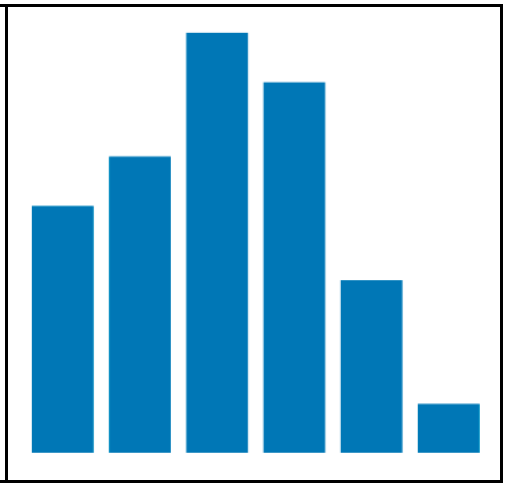


Figure 10: Bar Chart generated with Protovis.

Advantages

- Fast in building small charts
- No additional plugins required
- Can handle datasets with more than 10.000 data points

Disadvantages

- Code gets very complex for bigger charts

3.2.3. Processing

Processing [18] is an open-source programming language and environment especially for creating images, animations and interactions. The developers wanted a means to „sketch“ ideas in code. Originally built as a domain-specific extension to Java for artists and designers, Processing evolved into a full-blown design and prototyping tool used for large-scale installation work, motion graphics and complex data visualization. Processing itself is based on Java but because its kept fairly simple it supports only few libraries of the Java API. Figure 11 shows an applet with a static pie chart that uses a set of data points, stored in an array.

Code:

```
size(200, 200);
background(100);
smooth();
noStroke();
float diameter = min(width, height) * 0.75;
int[] angs = [30, 10, 45, 35, 60, 38, 75, 67];
float lastAng = 0;
for (int i = 0; i < angs.length; i++)[]
  fill(angs[i] * 3.0);
  arc(width/2, height/2, diameter, diameter, lastAng,
lastAng        + radians(angs[i]));
  lastAng += radians(angs[i]);
}
```

Figure 11: Pie Chart generated with Processing.

Advantages

- Many additional libraries available
- Easy to generate

Disadvantages

- Java required
- Applets are slow especially for bigger amounts of data points
- Very strict restrictions for external data access

3.2.4. InfoVis

The JavaScript InfoVis toolkit [19] is, like the name says, completely based on JavaScript and provides tools for creating interactive data visualizations for the web. Because of its JavaScript programming It is easy to integrate into other web applications. With release 2.0 the number of new visualization types has doubled. There are simple visualizations like Bar Charts and Column Charts up to very complex visualizations like Tree Maps and Graphs. It also includes very special visualizations like the Sunburst, shown in Figure 12. or the Icicle. As input for the graphs the toolkit uses only JSON (JavaScript Object Notation) Objects [20]. JSON is a lightweight data-interchange format that is easy to read and write for humans. It is similar to XML but reduces the necessary chars to a minimum. JSON objects only consist of simple, unordered name/value pairs.

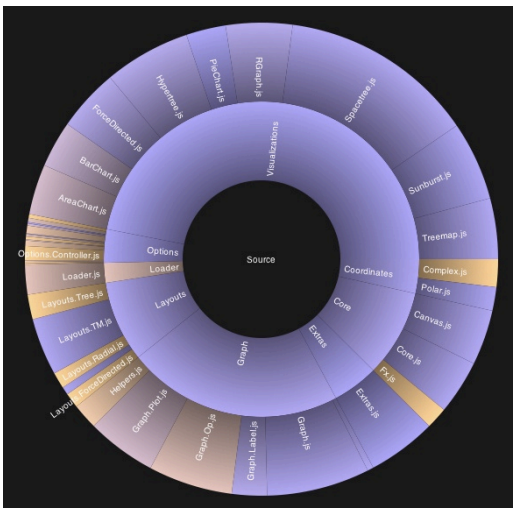


Figure 12: Sunburst generated with InfoVis.

Advantages

- Many complex visualizations
- Fast loading
- Fast in building
- Easy to integrate in other web applications because of JavaScript

Disadvantages

- Only JSON as Input allowed

3.2.5. Google Web Toolkit

The Google Web Toolkit (GWT) [21] is a development toolkit for developing and optimizing complex browser-based applications. It provides the developer with a huge collection of development tools, programming utilities and widgets that make the creation of Rich Internet Applications (RIA) totally different to other frameworks. The biggest difference to other web technologies is the use of Java to realize the browser-side code instead of JavaScript. The main problem of Rich Internet Applications is the ever increasing size and complexity that is very difficult to manage. Java was designed to make managing those large applications easier. While bringing all of Java's benefits to Rich Internet Applications there are still methods to integrate and communicate with JavaScript code. Means you can use your old code as well as third-party libraries that are written in JavaScript.

At the core of GWT is a Java-To-JavaScript compiler that produces optimized code capable of running on all modern browsers, like Internet Explorer, Firefox, Mozilla, Safari and Opera. Beyond this compiler GWT includes a large library of widgets and panels that makes it easy to create full web application that looks more like desktop applications. This library includes simple elements like textfields, buttons, drop down menus and other form fields. In addition, it includes more complex widgets including menu bars, dialog boxes, tab panels and others. Figure 13. shows a demo mail application completely created with GWT including panels, trees, lists and buttons.

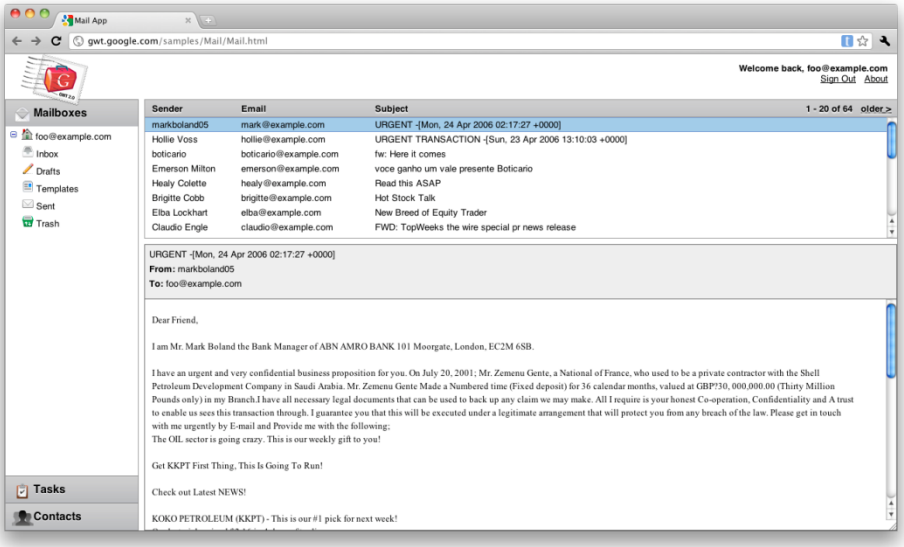


Figure 13: Demo Mail Application created with GWT.

But there are also additional libraries that aren't included in the standard package, like the Visualization API which is of special interest in this project. This API provides methods and

interfaces to communicate between data sources and visualizations. GWT itself includes only a basic set of charts like pie charts, bar charts and column charts. More advanced visualizations like graphical networks or tree maps are currently not included. But with the advantages of the tight JavaScript communication it's possible to integrate third-party libraries to use every chart needed. More information and a good starting-point for working with GWT is the book „GWT in Action" from Robert Hanson and Adam Tacy [22]

Advantages

- Java-to-JavaScript Compiler
- Simple programming of complex Rich Internet Applications
- Large library of panels and widgets

Disadvantages

- Debugging the compiled JavaScript code is practically impossible

3.2.6. Technology Discussion

The key differences between the technologies discussed in the previous sections, is the purpose they are developed for. Protovis, Processing and InfoVis are languages purposed for creation of graphics, such as charts and images. In this field the possibilities cover a wide range, from a simple 2D image to a complete interactive 3D animation. But for the purpose of building complete web applications this toolkits aren't the best choice. In first attempts we used these technologies to create prototypes for visualizing survival curves (Chapter 2.2.1) to discover which technology covers best our requirements. During these developments it turned out that building graphs is very simple but integrating other components such as buttons and windows isn't provided. Every component must be created by drawing single elements. This cost much time and lowers the performance.

This is where HTML5 and GWT provide a better solution. Both are made for creating complete, interactive websites. In HTML5 the most of the structure must be coded manually, but like GWT, it supports a full set of predefined panels and widgets, which makes implementing faster and easier. Also both support the generation of visualizations. HTML5 with the canvas element, where the individual graphics must be drawn manually with basic shapes and paths, and GWT with its basic set of visualizations that can be integrated. One must take note that HTML5 as GWT aren't made for visualizing data. Especially, large amounts of data and complex visualizations, both technologies are reaching their limits. This is where Protovis, Processing and InfoVis have their advantage. A summarizing of all pros and cons is shown in Table 2.

Technology	Advantages	Disadvantages
HTML5	• Common web language • Large pool of developers • Natively supported by most web browsers • No third party software required	• No predefined widgets • Javascript underlies the SOP
Protovis	• Fast development of small charts • No additional plugins required • Can handle large datasets	• Complex code for big charts
Processing	• Many additional libraries available • Easy to generate	• Java required • Slow • Restrictions for external data access
InfoVis	• Many complex visualizations • Fast loading • Fast in building • Easy integration possibilities	• Only JSON as input
GWT	• Java-to-JavaScript Compiler • Simple programming of complex RIAs • Large library of panels and widgets	• Difficult debugging after compiling

Table 2: Summary of pros and cons of the technologies.

Both technology groups have their own advantages and disadvantages. So our choice was to mix the technologies to get the best sides of both. The biggest advantage of GWT compared to HTML5 is that the predefined components make creation of RIAs very easy and fast. Furthermore, GWT provides a much better manageability of large applications. Therefore, we decided to use GWT as main programming language that is reasonable for the UI and the communication between the client and the backend. Additionally, we extended these UI elements by using ExtGWT [23], a Java library that modifies the available components and add even more. Since the visualization capabilities of GWT and ExtGWT did not meet our requirements, we decided to integrate an additional toolkit to to provide the visualization component. We choose InfoVis because of the possible interactions, fast graphics and the simple way of generating them, but this is only a personal preference of the author. In the implementation we show that it is quite easy to use the GWT visualizations as well as the InfoVis graphics, but it is also straightforward to integrate any other visualization technology such as Protovis or Processing.

Technology Decision

GWT and ExtGWT

- Core- Technology
- Client-Side
  GWT specific components such as buttons, windows, lists and panels will be used to generate an intuitive user interface that will contain the different visualization types.
  Simple charts like pie charts and columns charts will be also implemented by using GWT.
- Server / Backend
  The backend will also be implemented in GWT specific Java code but during the compiling it's translated into Java servlets. It will include the interfaces between the application and the data sources as well as the application logic which will transform the plain data into a format the tools can visualize.
- Client-Server Communication
  The communication between client and backend will be done by GWT RPC (Remote Procedure Calls), which makes it easy to pass Java objects back and forth over HTTP.

InfoVis

- More complex visualizations
- InfoVis is offering many new interactive visualization types such as SpaceTrees, ForceDirected Graphs, TreeMaps, Pie and Column Charts. For the first implementation of the tools the SpaceTrees will be used to visualize the biological pathways.

3.3. System Design

In this Chapter the general architecture of the web application as well as the generic design of the tools are explained. Both structures need special attention because they are responsible for the changeability and expandability of the tools.

3.3.1. Application Architecture

The general architecture follows the approach of the MVC (Model-View-Controller) Pattern. MVC is a modern type of architecture that separates the architecture into three layers.

The View renders the model into a form the user can interact with. This can be an interface element or a simple output. It is possible that multiple views exist for a single model. The Model manages and processes the data given from databases and web services. It responds to queries

about its state and responds to instructions to change state. The Controller is the communication interface between the view and the model. It accepts input from the user and instructs the model to perform actions based on this input. On the other side the controller takes the information from the model and sends them to the view.

This type of architecture supports the web application with the functionality of easily interchangeable and expandable tools. The data is processed in the model layer and every tool can get access to this information by sending a request to the controller. Figure 14 shows this architecture but with a special addition, every tool has its own controller in addition to the application controller. GWT requires a controller for every view to handle the incoming events. This makes the application even more flexible because if we remove a tool, we don't need to change anything on the controller or the backend. If a fired event reaches a tool, the tool handles the event and if there is no tool that could handle the event nothing happens. That is a unique feature since in other languages an exception occurs if a request doesn't reach the target.

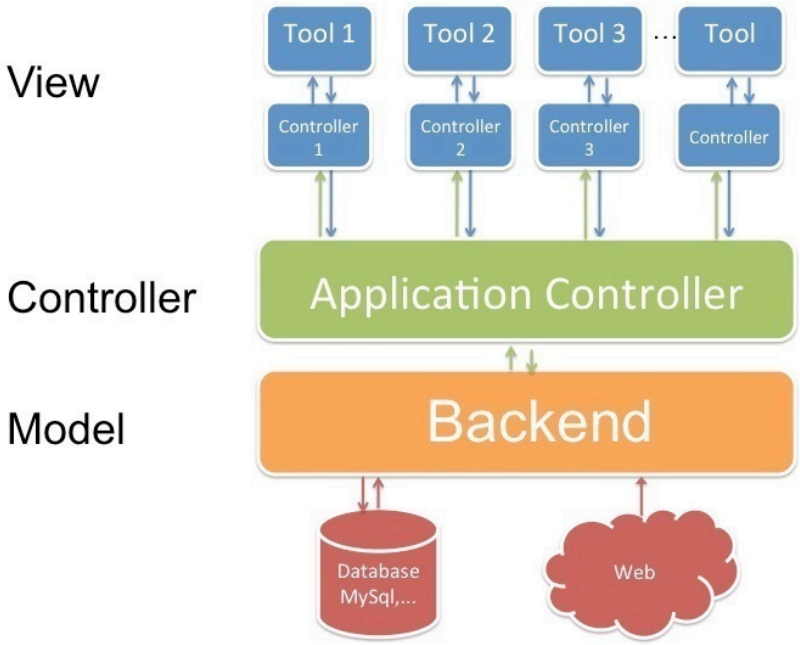


Figure 14: General architecture of the web application.

3.3.2. Generic Tool Design

The exact implementations of the tools can differ from each other in some detail. However, the principal design which all tools are following is shown in Figure 15. Every tool accepts as input the data of the actual patient and the particular data from databases, the web or local files. The patient data is needed for automatic selections, searches and similar things. This makes the work more efficient for the physician. The tools consists of a set of UI elements like panels, buttons, containers, etc. which supports the physician with different possibilities to interact with the tools. The output is defined by the tool and the visualization type. The data can be visualized in elements such as pie charts, maps, tables, etc.

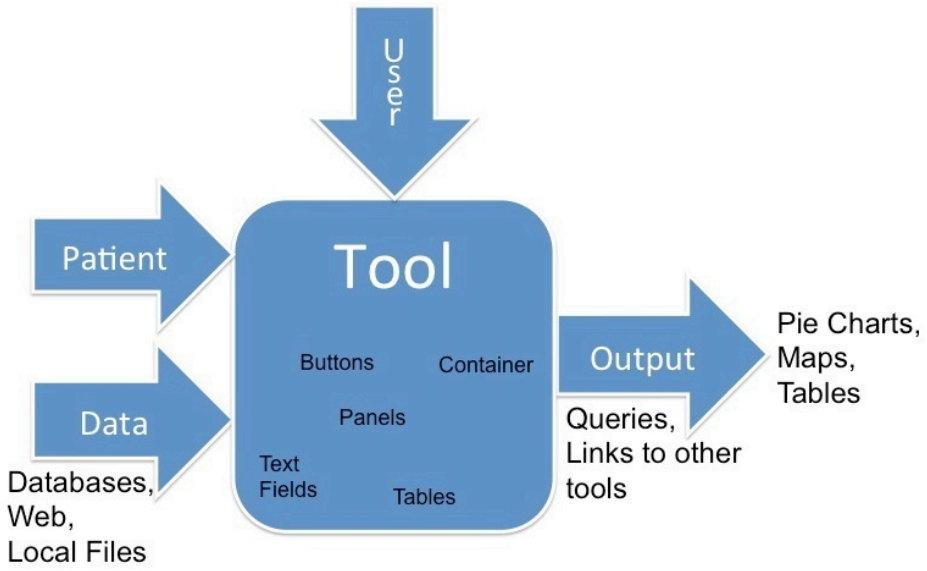


Figure 15: General architecture of the tools.

Chapter Four: Development

The development of the individual tools is the most time consuming step in the process of creating the application. The first and important step of a good implementation is to be clear about the purpose of the implementation as well as the characteristics and attributes. In this chapter the procedure of developing each of the tools is described in detail. Each tool description begins with a clinical scenario which was the motivation for the particular tool, followed by the exact specification of the attributes and methods. This specification defines in detail the input and output of the tool as well as every function that has to be integrated. The general implementation of the tool and special abilities of the tools are then provided followed by the issues and challenges that occurred during the development process are discussed.

Since the tools only reflect the visualization mechanisms of the application we describe in an additional section the structure of the backend which is the interface between the data source and the visualization. It describes the procedure of processing the data and making them available to the tools. Furthermore it explains some special implementation options that are given by the GWT.

4.1. Visual Query Builder

The Visual Query Builder (VQB) is a multifunctional tool. On clinical side it is used to visualize statistical occurrences of cancer within specific attributes, for example the patients' age or cancer grade. These statistics are visualized in form of pie charts that show the different values of the attribute in form of slices. The patients' value is automatically highlighted in the chart so that the physician directly gets the information about his patient.

On application side, the tool has an additional feature, makes it working as a Query Builder. The Query Builder allows the user to create a query by selecting different values in the pie charts or by activating or deactivating complete pie charts. These queries can be used by external tools to modify their data or display additional information.

4.1.1. Motivation / Clinical Scenario

The epidemiology of cancer is the study of cancer specific factors such as cancer incidences, mortality and other tumor characteristics to find the cause of cancer and to identify and develop improved treatments and therapies. It could be involved directly into therapy decision finding of physicians by presenting population based statistics for the current patient to them. For a new patient, this information about how frequent or rare the cancer is and the prognosis of the patients with a disease most similar to his/her own would be quite beneficial. For the doctor,

getting data on the types of treatment available for patients similar to his own maybe useful. Furthermore, the given population would reflect the survival outcome of the patient.

4.1.2. Specification

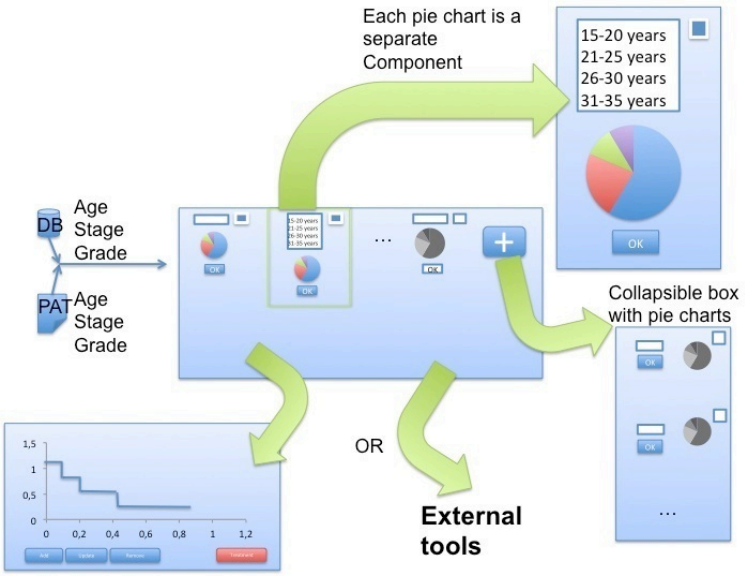


Figure 16: Overview of the Query Builder.

The Visual Query Builder (VQB) has two main functionalities. It is used as presentation tool that visualizes epidemiological data in form of charts and it is a query builder which uses the interface components to create queries that can be used by external tools. The specifications for both functionalities are explained in more detail in this chapter.

For the presentation part the VQB requires two data sources, population based data from a database and the patients' personal and medical data, for example age, gender, grade, etc. The patients' attributes are typically a subset of the attributes available in the database.

The interface which presents the epidemiological data is structured very simple and clear. It only consists of a set of chart components, whereas not more than three components are shown on the main screen. All other components are placed in a smaller version, in a collapsible box on the right border of the screen. They're ordered in a vertical, scrollable row with a pager on the top and bottom. In this way only a maximum of five components will be shown at a time the rest can be shown by switching the pages. The box can be opened and closed by a toggle button, which is also located on the right border.

Each of the chart components consists of the pie chart itself and a collapsible list to select values of the chart, for example to narrow the range of the age from 41-50 years to 45-47 years. This

selection box shows all available values for the attribute, so that the user can select one or more of them to modify the shown chart. A submit button that confirms the modifications is located at the bottom.

The initial screen shows three charts in the main panel and the remaining in the hidden box. This distribution is defined in the programming, but could be implemented as dynamical variable in a future version. The order of these charts is alphabetical, which can be changed during runtime through Drag'n Drop.

The creation of queries is the second ability of the VQB. It extents every chart component by a checkbox that enables or disables this attribute for the query building process. After a query is requested by an external tool, the Query Builder checks the status of all chart components and collects all enabled charts. The attribute name (age, gender,…) and the selected value(s) of these charts are mapped and stored in a query. This query is sent back to the requester and can be used for other visualizations. The possible procedure of generating a query is shown in Figure 17.

The user has three possibilities to interact with the Query Builder:

- Change the order of the charts by Drag'n Drop
- Un-/checking the checkbox reasonable for the state of a chart to enabled/disabled
- Select values in the according list of a chart to change the focus of this chart

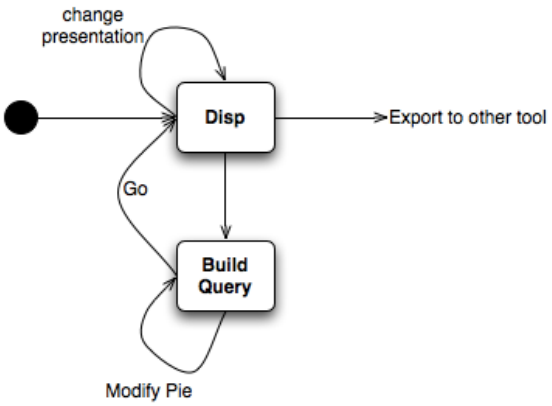


Figure 17: Flow Chart of the process of building a query.

4.1.3. Implementation

The user interface of the Visual Query Builder consists of a horizontal main panel and a vertical collapsible box that are filled with objects of the class *PieChartComponent*. Each of these objects represents an attribute of the loaded database for which a mapping in the application exists. In this case mapping means the connection between a unique identifier in the application

and the equivalent attribute in every database. This mapping is realized by model classes. Every database that is used needs a new model that maps its attributes to the available identifiers.

If the attribute is mapped a new component is created. This component consists of a content panel that has the attributes' name as title and a toggle button in the header that changes the state of the component between active and inactive. Below the header a collapsible list of the values is following as well as the actual visualization. This visualization is a pie chart from the GWT Visualization library and includes the pie chart itself, a legend and tool tips. The color of the chart is chosen randomly by an additional function of the utility class and changes only the brightness within itself. The deactivation of the component makes the list and chart grayed out and unclickable. Differing to the specifications the final version doesn't need a submit button because changing the selection of the list automatically effects in a recalculation of the pie chart.

The collapsible box mentioned at the beginning of the chapter is an instance of the class *PieChartBox* and contains the remaining chart components. It's located at the right border and can be toggled by a button that is located at the same position. The box has the same size than a usual chart component. That's why it includes the remaining charts in a smaller version so that more charts can be displayed at the same time.

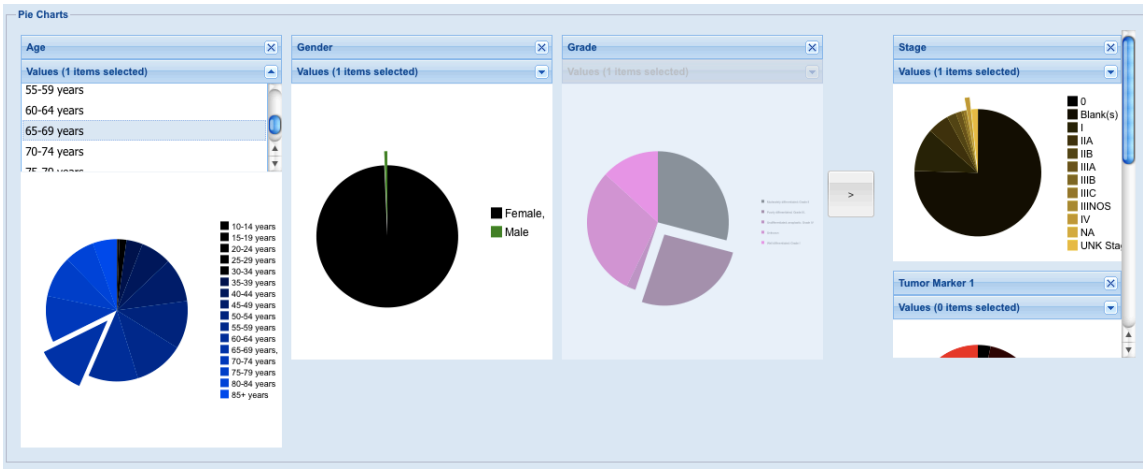


Figure 18: Screenshot of the implemented Query Builder.

4.1.4. Issues and Challenges

The biggest challenge in implementing the Visual Query Builder was to build it in an intuitive and attractive design that includes all the essential functionalities. This was the reason why we decided to make a main panel with only three chart components and put the remaining components in a separate box. We inhibited any controls like buttons so that the full concentration is on the charts.

An issue we couldn't solve completely is the use of Drag'n Drop to modify the order of the components. It is possible to change the sorting order within one of the lists, but if we try to put a component from one list into the other we face two problems. On one hand the size of the components differs and on the other hand the components in the box are implemented in a separate class what makes the transfer a lot more difficult. So we decided to deactivate this feature in the first release of our project but planning to fix it in one of the next releases.

Another issue was the use of different databases in different tools. Often the attributes in population based databases mean the same but they differ in the naming. So we have the gender in one database but sex in another. To solve this problem we build a mapping between the databases and our application, described in more detail in Chapter 4.1.3 Implementation.

4.2. Survival Curve Manager

The Survival Curve Manager (SCM) is a presentation tool that interfaces with the Visual Query Builder. It presents statistical survival rates of specific population groups in form of survival curves whereby each curve shows the percentage of patients that are alive at specific times. Each time a patient dies the curve drops.

The ability to manage multiple survival curves makes it possible to compare the survival data for several populations simultaneously. This is done by using the dynamically generated queries of the VQB to modify the input data of the Survival Curve Manager.

4.2.1. Motivation / Clinical Scenario

Survival Curves have great potential in visualizing the risk to the patient. They show the statistical progress of a disease to the physician and the patient. The comparison of several curves is even better because it is possible to visualize the differences of different therapies. This can help the physician to decide which therapy is possibly the right or it can help to explain to the patient why therapies are important and in which way they can help to extend his life.

4.2.2. Specification

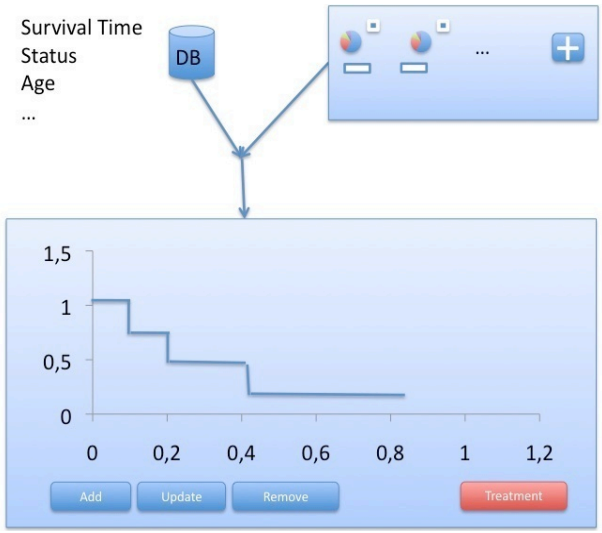


Figure 19: Overview of the Survival Curve Manager.

The Survival Curve Manager is a pure presentation tool, which only includes controls for adding, updating and removing curves. For this reason, two input types are required. At first a database is needed that contains the statistical information that should be presented. The second input is the query coming from the Visual Query Builder (see Chapter 4.1) that filters the data before it is used to calculate the individual curves. There are two restrictions that must be noticed. The database must contain the attributes survival time and status since they are necessary for the calculations and the attributes that are used in the query must be a subset of the attributes in the database.

The interface consists of a control panel that includes the buttons for adding, updating and removing as well as the main graph that will present the curves. The graph can manage several curves whereas the limit is set to five curves at the same time. Every curve is drawn in a different color so that they can be distinguished from another. A legend at the top of the graph shows the attributes each curve is filtered. Furthermore the graph shows the actual percentage of survival by moving the mouse over a curve.

As mentioned above the control panel contains options for adding, updating and removing curves. When adding or updating a curve, a new query will be requested from the Query Builder and included into the calculation process. The update and remove function are requiring a selection of a curve before they can be used. This selection can be made by clicking a curve.

In Figure 20. the procedure from getting a query to displaying the calculated function is drawn schematically.

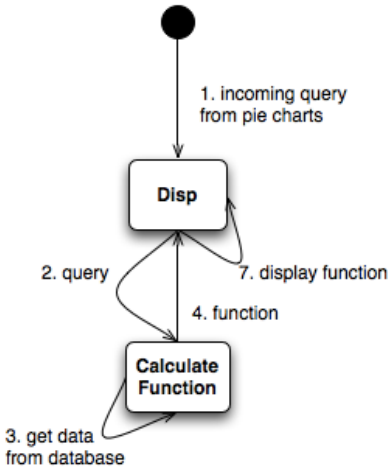


Figure 20: Flow Chart of generating a survival curve.

4.2.3. Implementation

The user interface of the Survival Curve Manager (SCM) consists of two panels. A control panel which contains the three buttons for adding, updating and removing curves and a chart panel. For implementing this chart, we created a new Visualization *StepChart* which inherits from the GWT Visualization type *LineChart*. *LineCharts* are usually used to create curves in charts, but they don't support the creation of stepped curves, that means you can only set single points that are connected by a rounded line. The new class *StepChart* uses all functionalities of the class *LineChart* but adds a new function *addStep(int x, int y)* that creates instead of a single point, one point of the same height than the last point of the curve with the distance *x* and a second point on the same width than this new point but width the height *y*. In this way *addStep* creates angular plots instead of smooth curves. The advantage of inheriting from the GWT Visualization class is that our new class adopts the functions for the legend and tooltips. The legend automatically shows every curves which is added to the chart in the same color than the curve. This color is chosen randomly by the method *getRGBColor()* of the class *Util.Color*.

The process of creating a curve is started by clicking the add or update button. After pushing one of the buttons the event *GetQuery* is fired. The tool waits until another tool caught the event and fires the answer event *SetQuery* which contains the query data. The SCM extracts the query and sends it to the backend, where it is used to collect the data from the database. In the current version of the tool the SEER database is used as data source but in the further development process other databases should be integrated. After the data is collected it will be translated into

a DTOSurvivalCurve Object that consists of a HashMap that contains the values. This DTO is sent back to the SCM, where it is used to calculate a new curve.

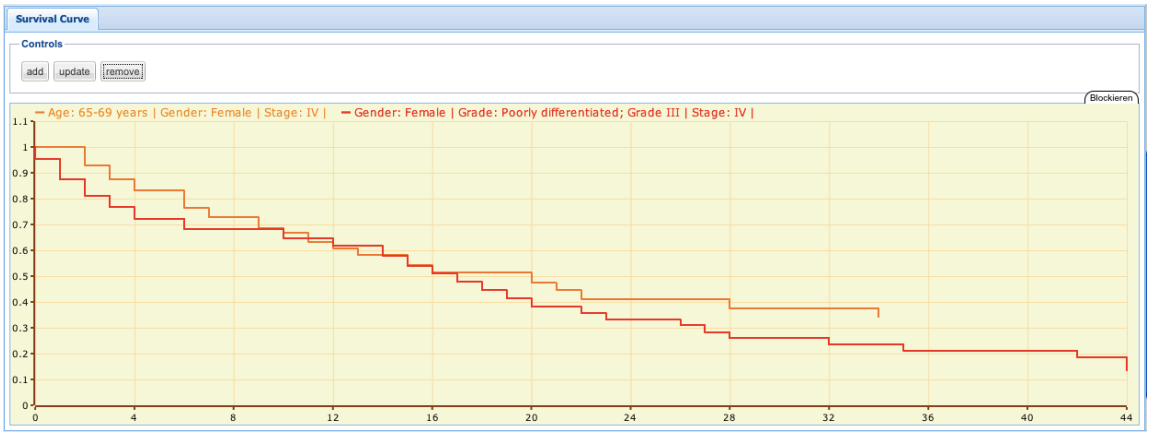


Figure 21: Screenshot of the implemented Survival Curve Manager.

4.2.4. Issues and Challenges

The challenge in the implementation of the Survival Curve Manager was to get the query from the Visual Query Builder into the Survival Curve Manager. This problem of requesting data from other tools was solved by the help of events which are fired at the right moment.

A second issue are the datasources. One of the goals of this tool is to compare the survival times for different therapies for a specific population. Unfortunately, until now the given databases don't support patient focused therapy information. So that we're only able to present curves for different patient attributes.

4.3. Biological Pathway Visualizer

In cancer research pathways are playing an increasingly important role in deciding treatment. The one-target, one-drug approach has not held up for the most types of cancer. Recent studies that deciphered the genomes of cancer cells have found a wide spectrum of different genetic mutations that can lead to the same cancer in different patients. Researchers hope that they can find the disrupted biological pathways caused by these mutations and use drugs that target the specific pathways for treatment. In this way researches could focus on their attention on just two or three biological pathways instead of dozens of mutations.

The Biological Pathway Visualizer (BPV) supports the physician with the ability to choose a biological pathway which is displayed with its genetic nodes and connections. This pathway can be analyzed with the patients' data to get a new look on the pathway which shows the mutations and deregulations and its effects. If available the BPV provides additional information to the deregulated gene like drug or study information.

4.3.1. Motivation / Clinical Scenario

Understanding disease at the level of biological pathways has a great potential in providing input to clinicians and improve diagnosis and treatment of cancer in the future. Using patient data and pathway activity relevant to the disease can highlight the state of such pathways and enable the clinician to focus on the impact of any deregulations. Presented in a visualization it is possible to show complex pathways with their deregulations in a way the physician can easily grasp and recognize the targets where he can interfere.

4.3.2. Specification

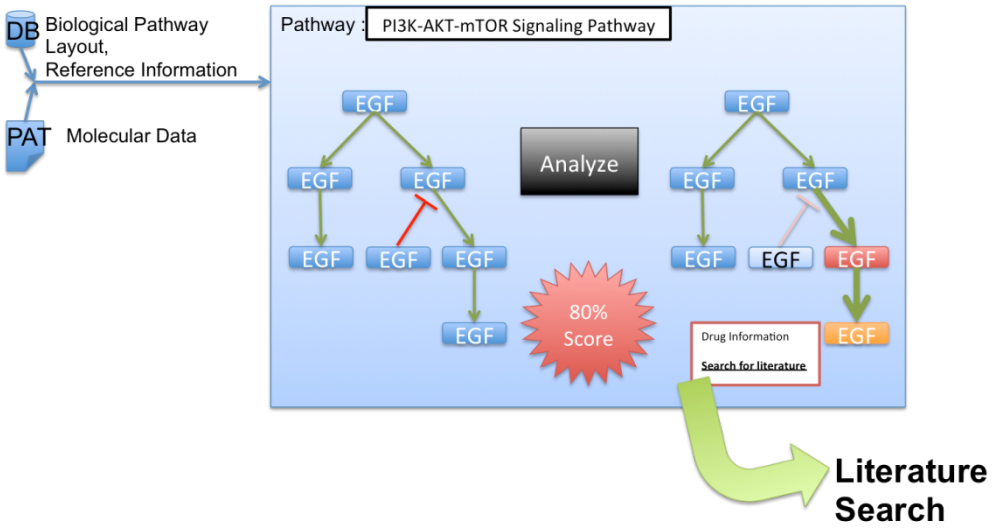


Figure 22: Overview of the Biological Pathway Visualizer.

The input of the Biological Pathway Visualizer consists of biological pathway layouts and reference information from a database and the patients' molecular data.

The interface is divided into two halves, the left shows the pathway in its original version, the right shows it after the analysis with the patients' molecular data. Between these two halves a

button is placed that starts the analysis and a score is displayed that shows the resulting risk of this pathway. On top of the screen is a selection box where the pathways can be chosen.
The different interactions of the pathway are drawn in different forms and colors. Activating interactions are shown in green lines with arrows at its end and inhibiting interactions are shown in red lines with an orthogonal line at the end. If a molecule is deactivated or not existent in the analyzed pathway the interaction and the element itself are drawn translucent. The effects of this deregulation are visualized in bigger lines for the interactions and in red colors for the molecules. An exception are deregulated molecules which contain additional information, these molecules are drawn in an orange color. Clicking the orange molecules opens a small text box which contains information to useful drugs or studies, as well as a link to the literature search. This link opens automatically the literature search with the information stored in the box as input parameters.

The analyze button in the middle of the screen uses the selected pathway and the patients molecular data and sends them to the backend where they are analyzed. The result will then be presented on the right halve of the screen.
Below this button a perturbation score is shown which presents the risk of this pathway in percent.

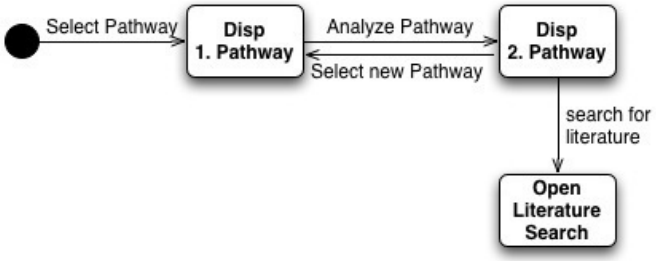


Figure 23: Flow Chart of analyzing a pathway.

4.3.3. Implementation

The implementation of the Biological Pathway Visualizer differs to the implementation of the other tools since it's mixing the typical elements of GWT, the GWT Visualization API and the InfoVis toolkit described in Chapter 3.2.4 InfoVis. The standard interface with the toolbar at the top and the controls in the middle are still implement by using GWT elements. The toolbar uses a selection box to choose the pathway and a button to load it. The main screen is divided into three parts where the middle section is used for the controls. It contains an additional selection box for the therapy option, a legend that describes the particular elements of the pathway and a button for the analysis of the pathway. This analysis is done by an additional module developed

by the Philips research group, which provides methods to compare original pathway data with the data of the patient. An additional component, which is located below the legend, is the gauge that shows the perturbation score of the current pathway analysis. This gauge is an element of the GWT Visualization API and is integrated within the GWT code directly. The pathways are implemented besides this middle section, the original pathway without any modifications on the left side and the analyzed pathway with the modified molecules on the right side. As mentioned above we used the InfoVis toolkit to create the pathways. Since there aren't any interfaces or wrapper classes for the toolkit, we had to generate a new class *Pathway* which creates an iframe that hosts our visualization. We used an iframe because it offers the possibility to implement inside it completely independent and it is easy to integrate into GWT. After creating instances for the original and the modified pathway we can load the pathways with the function *loadPathway(String frameName, String json)* into the iframes. The parameter *frameName* is the unique identifier for the iframe and *json* is the data string that should be loaded. This method is a JavaScript Native Interface (JSNI) method that means it is declared *native* and contains JavaScript code in a specially formatted comment block. This special format is a directive to the Java-to-Javascript Compiler to accept any text between the comment statements as valid JavaScript code and inject it in the generated GWT files. In this way we can build a communication between the GWT code and the JavaScript files. The visualization itself is done in separated html and javaScript projects which are loaded into the frames. In this way the html file loads the InfoVis library and builds the container for the visualization which is loaded into a <*div*>- Element. The JavaScript file for the pathway is more complex it has a defined structure depending on the chosen visualization type. For the pathways we chose a SpaceTree as visualization type which has a root element and an unlimited number of children. It also provides options to modify the nodes and connections in colors and form. The json string from the *loadPathway* method is used as input and has a specific structure with node attributes and adjacencies, an example is shown in Figure 24.

```
[{
  'id': '1',
  'name': 'MEK',
  'data': ['$type': 'circle'],
  'adjacencies': [{
                'nodeFrom': '1',
                'nodeTo': '2',
                'data': {
                        '$type': 'arrow',
                        '$color': '#008B00'
                       }
                 }]
}
,
{
  'id': '2',
  'name': ‚PI3K‘,
  'data': ['$type': 'circle'],
  'adjacencies': [{
                 'nodeFrom': '2',
                 'nodeTo': '3',
                 'data':{
                         '$type': 'arrow',
                         '$color': '#008B00'
                        }
                  }]
}]
```

Figure 24: Example JSON string for pathway input.

Special attention was needed for the popup window which appears by clicking a highlighted molecule. The creation of the popup and the event which leads to the opening of the window was done by functions of the InfoVis toolkit. However, the integration of the Literature Search was more difficult because our GWT project has to react to an event which happened inside the iframe. To build this communication we created again a native method *registerLiteratureSearchClickListener(String name)* on GWT side which connects the method *literatureSearch(String term)* of the iframe with the method *fireLiteratureSearchEvent(String term)* of the class *Pathway*. Now, each time the „search literature“ button is clicked the event is forwarded to the method *fireLiteratureSearchEvent* which opens the Literature Search tool.

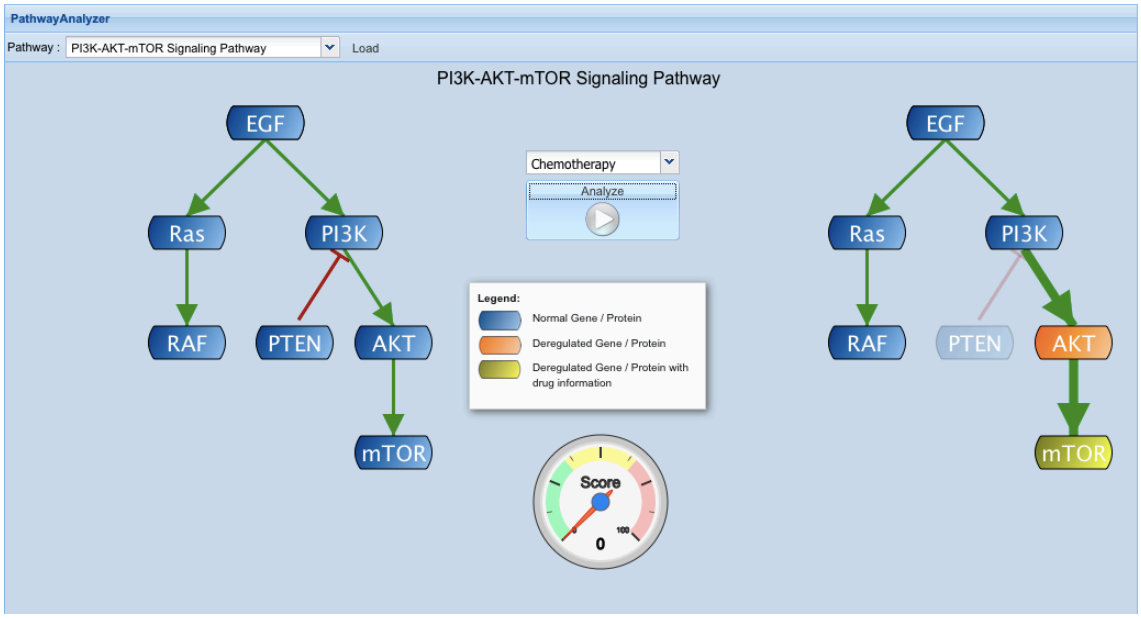


Figure 25: Screenshot of the implemented Biological Pathway Visualizer.

4.3.4. Issues and Challenges

The Biological Pathway Visualizer is the most complex tool of our collection. It handles a very new type of data and we're trying to present it in a new way.

So the first challenge that occurred was the integration of the already available pathway data into the tool. This data is mostly stored in the BioPax (Biological Pathway Exchange) format which is a RDF/OWL-based standard language that tries to standardize the creation of new pathways. Since the structures and interactions which are based on the pathways are very complex also the BioPax files are getting very big and complex. So it would take a lot of time only to extract the necessary information out of the files and to create an input stream for our tool. Since our focus is mainly lying on the visualization of data and not in pre-processing the available data we decided to use simplified example data for the first release. This example data is stored in a local database within two tables. One table for the molecular elements of the pathways and a second table that contains the interaction between these elements. In this way in later releases the BioPax data can be adapted and stored in the database to integrate pathways into the tool without changing the code. However, we also have developed a simple and extensible to use tab limited format for encoding pathway information that can be used for any other pathway.

Another challenge was the integration of the InfoVis toolkit into the GWT project. Since there is no wrapper or interface class yet we had to find a different way of integration. With the help of iframes and JSNI we created a complete different structure as in the other tools that make a communication between the two different languages possible. The exact implementation is explained in Chapter 4.3.3 Implementation.

After we integrated InfoVis into GWT we were facing to the question of which type of graph we should use. We tried several possibilities from simple graphs over special graphs like weighted graphs to more complex trees. But the problem we had with all these types was the ordering of the elements. The graph orders the elements randomly so every time we created a new graph we got a new ordering what is unhandy because it is very complicated to compare two graphs even if they differ in one element. The tree has a similar problem, here you can't remove a node that has a subnode since every element needs a parent. To solve these problems we decided to change our imagination of the pathway a little bit. Instead of removing an element completely from the visualization we decided to mark it as removed. This solution has the advantage that we can use a tree to draw the elements and as a second effect it is easier to see which elements differ from the original pathway because they are drawn in a translucent way.

4.4. Geographical Trial Finder

Clinical Trials are research studies in which patients and doctors work in an improvement of health and cancer care. Each study tries to answer a specific scientific question and tries to find better ways to prevent, diagnose or treat cancer.

The Geographical Trial Finder helps the user to search and find the right trials in huge databases like ClinicalTrials.gov. The results will be presented in a common list as well as a geographical list that shows the locations of the study. Additionally, this search can be filtered by specifying the location, choosing a study type and so on. Markers on the map are containing the description and information to the according trials.

4.4.1. Motivation / Clinical Scenario

Very often the last chance for cancer patients with difficult or uncommon cancer types is the attendance in clinical trials. These trials are using new drugs and treatment methods to find out whether promising approach is safe and effective. But also in the treatment of more common cancer types it is important to improve the different steps that lead to a cured patient. So the most cancer patients decide to attend in a clinical trial. In both cases it is important to know where appropriate trials are located so that patient can decide which research facility looks the best for him and where he feels most comfortable. Today, the www.clinicaltrials.gov is the repository for all registered trials in the United States. However it is difficult to navigate and does not provide even preliminary filtering options that are available in consumer-focused websites. Our motivation is to overlay clinical trial data onto a geographical maps interface such as google maps that would allow patients and caregivers alike to be able to identify clinical trials that are applicable to the current patient,

4.4.2. Specification

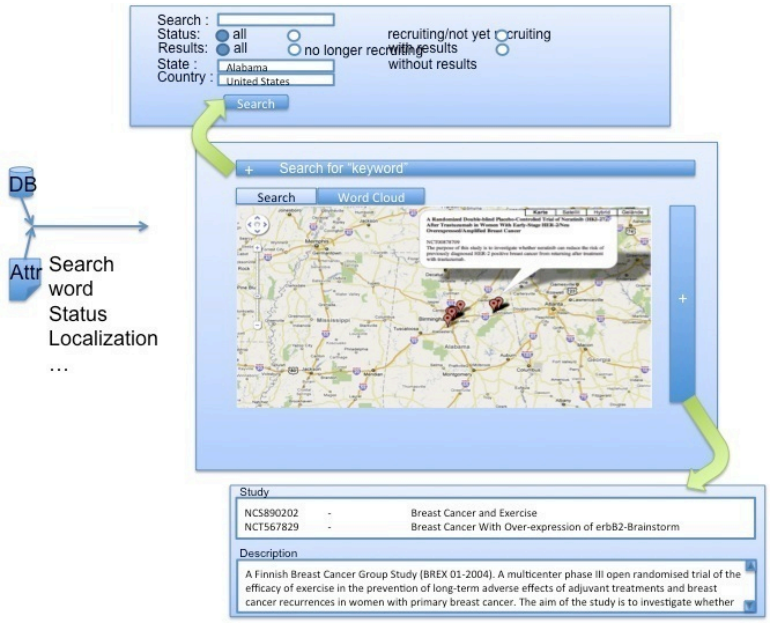


Figure 26: Overview of the Geographical Trial Finder.

As input the Geographical Trial Finder requires a study database like ClinicalTrials.gov that contains values for the title, description and localization. Additionally pre-defined values as from the patient data can be loaded to make a pre-selection of the search values.
The interface is divided into three parts. The search form, which is located on the top of the screen - the geographical map that takes the main part in the central of the screen and the search result panel which is located on the right of the screen. The search form and search result panel are collapsible that means they can be closed so that only a small bar remains. Then the map uses almost the complete screen what makes it easier to look around.

The search form includes the elements that are needed to perform a study search. These are:

- Search Term
- Status (all / open / closed)
- Results (all / with / without)
- State
- Country
- Gender
- Age Group
- Phase

These values are filled out automatically if the pre-defined values are set. The search term is set by external tools and the patient according values gender and age group are set by the patient data that is stored in the system.

The search result panel contains a paging list that takes a maximum of ten studies at the same time. The list shows the values

- Study ID
- Title
- Facility
- City
- Zip
- State
- Country

of every study, whereas the studies are grouped by their study ID. In this way each study with at least one location is shown in the list, but the study also can have unlimited other locations that are also listed.

The geographical map contains controls for zooming in and out and moving around. Each study location that is found gets a marker on the map. These markers include information about the title of the associated study and a short description that is opened in form of a cloud by clicking the marker. For supporting the user, the result list and the map are connected. If a study in the list is clicked, the map will zoom to the associated marker. On the opposite if a marker is clicked the associated study in the list will be selected.

The initial screen shows the search form with selected values if available and the geographical map. At this time the search result panel is closed. After initiating the search the search form closes automatically and the search result panel opens and shows the found studies. Then the locations of the results will be determined and positioned in form of markers on the map. In this way the map uses the most available space at any time. But of course the user can also control the state of the panels and open and close them as desired. The complete procedure of searching is shown in Figure 27.

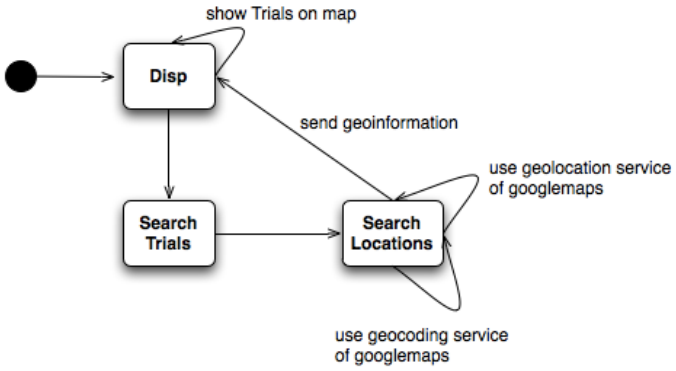


Figure 27: Flow Chart of searching and locating trials.

4.4.3. Implementation

The interface of the Geographical Trial Finder (GTF) is divided into three parts as described in the specifications. The search form is the simplest component it's a plain, collapsible panel that includes the different kinds of Text Fields, Selection Boxes and Checkboxes which represents the search parameters. During the initialization the patient data is loaded and the patient-specific fields such as gender and age group will be filled out automatically. The search result panel is a little bit more complex. It contains a pageable list that shows a maximum of ten trials on one page. These trials can be extended by clicking the „plus" in front of every trial to show all the locations of the according trial, with the facility name, city, zip and country. The third part takes the geographical map. It is realized by integrating Google Maps. Google Maps isn't part of the GWT but the official Google Maps API library for GWT provides a way to access the Google Maps API from a GWT project without having to write additional JavaScript code. After including the library we can instantiate an object of the class *MapWidget* which represents a single map with all features. These features cover zooming and moving the map as well as the creation of markers that is used to locate the trials on the map. Also these markers contain the title and description of the according trial which they show after clicking it.

The localization of the trials follows three steps:

In the first step the search is used to find the relevant trials. After submitting the search form, the GTF sends a request with the search parameters to the backend. The backend searches in the specified database for the trials, translates each of them into a DTOTrial object and put them into a list that is returned to the GTF. After getting the results of the search, the second step is following in which the trials are grouped by their ID and loaded into the search result list. In Step three the location attributes of each trial will be translated into geographic coordinates (latitude/ longitude) which can be pointed on the map. This translation is called geocoding and is also included in the Google Maps API. The geocoding service makes a call to an external server

where the sent data is used to query a huge database. In this database all the text based location data like cities, zip codes, streets, etc. are stored in different combinations with their according geographic coordinate. After the tool gets the coordinates it creates for every found point a new marker and sets its title and description text. This step is repeated every time the data in the list changes, what means every time the shown page is changed or a new search is started.

Above the search result list an additional button is available that uses the trial descriptions to generate a new Interactive Word Cloud.

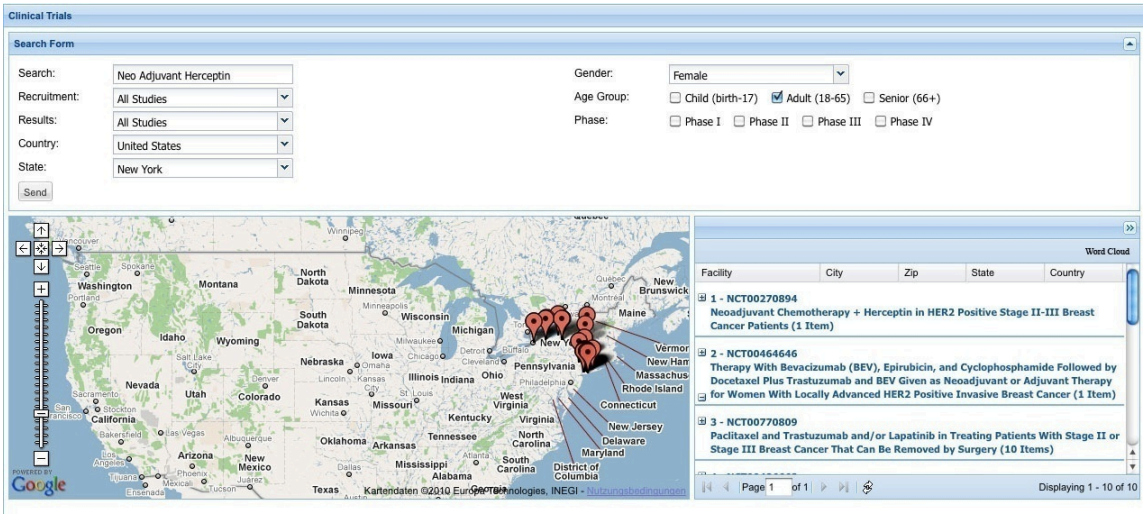


Figure 28: Screenshot of the implemented Geographical Trial Finder.

4.4.4. Issues and Challenges

The first challenge in the implementation of the Geographical Trial Finder was to get the data from the data source into the tool. In this first version of the tool we used ClinicalTrials.gov as data source. This caused the problem that the data is located on an external server which is not accessible via JavaScript thus to the Same Origin Policy (SOP). SOP is a security concept that prevents access to methods and properties that aren't in the same domain than the original page. So we implemented the search mechanism in the backend which has no restrictions for accessing external servers.

The next problem was the format in which the locations are stored at ClinicalTrials.gov. They are saving only the written addresses of the locations, means street names, city names, zip codes, etc., but geographic map (Google Maps) provides only methods that generate markers for geographical coordinates. So we had to translate these written addresses to geographic coordinates before we can use them. This can be done by a service called geocoding. The Google Inc. offers such a service in connection with GWT so that we could use it directly in our code. Unfortunately the Google Geocoding API is subject to a query limit of 2.500 geolocation

requests per day and not closer defined limit of request in a short interval. We tried to avoid this limiting by using other services like the Yahoo PlaceFinder, Bing Maps, OpenStreetMaps, etc. but all of them have restrictions in querying the data or it wasn't possible to use the service in a reasonable way. A different idea was to generate our own geocoding service. We took the necessary data for two states of the US from OpenPHI [24] and put them into a database. We used only the data of two USA states as a proof of concept for our prototype, but in theory this can be scaled up. Each data package consists of fields for the available addresses as well as the belonging geo coordinates of the state. Then we implemented a small interface that searched for the incoming addresses in the database and got the geo coordinates as result back. This approach was very promising but it needs a very huge storage and time to install which we didn't want to spend at this moment, so we decided to use the Google Geocoding API and perhaps develop this method for a later use.

4.5. Literature Search

The Literature Search is a tool that supports clinicians with a simple but powerful way of text based searching. It provides a simple UI with a search field but also with a large amount of filter functionalities that makes the search more precise. The results are shown in a paging list that shows the attributes title, author, publication and publication date. Each entry also includes a short description text which is shown by expanding it and can be opened in a full information page. The complete tool is connected to the Interactive Word Cloud and can summarize the results with the use of it.

4.5.1. Motivation / Clinical Scenario

In a time where dozens of books and publications are published every day it's almost impossible to read every new article or to remember every text you read. But it's very important for clinicians to stay up to date. They have to know the newest developments in medicine when a patient is asking. For this reason a tool is necessary that supports clinicians with a simple and fast access to these large repositories. Such a tool can help finding clinical relevant publications for a current patient without searching in all the databases currently available.

4.5.2. Specification

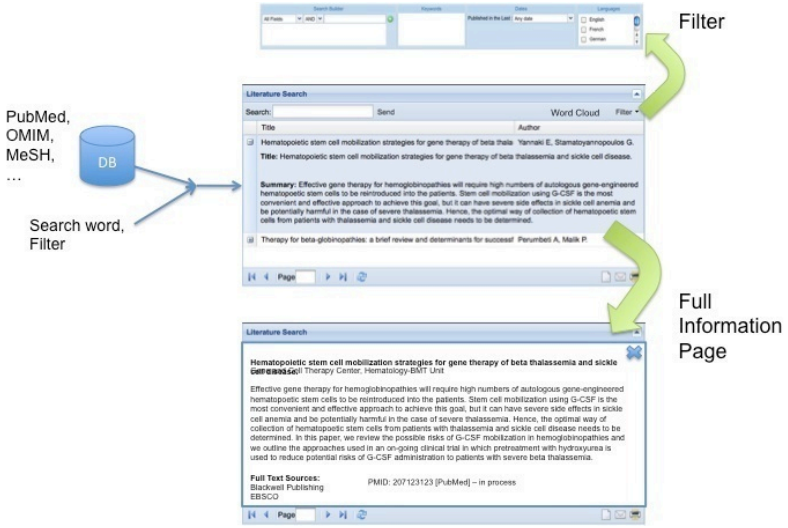


Figure 29: Overview of the Literature Search.

The input of the Literature Search consists of one or more text based databases which contains at least the attributes title and description. Additionally pre-defined values from external tools can be used to automatically fill out the search term and the filters.

The interface consists of a toolbar that contains the general controls, a filter bar and a pageable list for presenting the results.
The toolbar includes a search field in which one or more words can be insert, a button that transfers the results to the Interactive Word Cloud and a button that opens and closes the filter bar.

After opening the filter bar several options of filtering are available.

- A search builder that provides the possibility to search for a word that only exists or especially not exists in a specific field.
- A keyword list in which often occurring words can be excluded. This list will be created after the first search and can be used to specify a second search.
- A filter for the publishing date in which the publishing date can be limited to a specific range.
- A language filter that limits the searched texts by their language.

The result list shows a maximum of 20 entries on one page, the controls that enable the browsing through the pages are located at the bottom of the list in a second toolbar. This toolbar also contains three buttons for printing, emailing or creating a pdf of the results. The entries are represented in the list with their

- Title
- Author
- Publication
- Publication Date

By clicking an entry it expands and shows a short description of its content. Making a double click on an entry opens a new panel that contains the full information about the entry including.

A search can be started automatically if the tool is opened from an external tool or directly by clicking the search button in the toolbar. The search needs at least one word in the search field that it can be executed. The filtering can be configure before the first search or after every search to get a more exact result list. Helping to get an overview it's possible to summarize the results in form of an Interactive Word Cloud. The possible procedure of searching is shown in Figure 30.

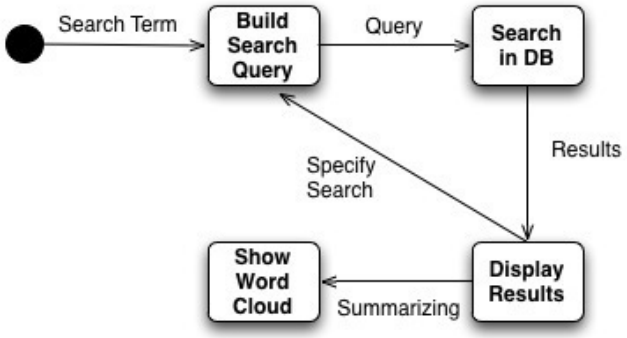


Figure 30: Flow Chart of searching for literature.

4.5.3. Implementation

The Literature Search tool is implemented in the three classes *LiteratureSearchView, LiteratureSearchFilterBar and LiteratureSearchDetailWindow*.

The *LiterSearchView* is the main class of the tool, it contains the UI elements and the services that communicate with the backend. The UI is structured in four graphical components which are ordered in a vertical line. The first element is a toolbar in which the search field, a button for the creation of Word Clouds and a filter button are placed. Below this toolbar a second collapsible toolbar follows that contains the possible filter options for the literature search. It can be opened and closed by clicking the filter button in the first row. Since the standard toolbar of GWT doesn't support the functionality of collapsing we created a new class

LiteratureSearchFilterBar which inherits from the class *ToolBar* and coded the functionality of collapsing on our own. The new filter bar currently consists of four filter types:

- A Search Builder that is realized by two selection boxes that include the fields (All fields / Title/ …) and the condition (And / Not) and text field for the search term. The created items are stored in a list of HashMaps in which every HashMap includes the fields name, variable, operation and value.
- A keyword list that shows the five most occurring words of the last search. Every time the result list changes, the service *getKeywords(List<DTOLiterature>)* is used to send the text to the backend and get a new list of keywords back.
- A date filter that shows a selection of dates like the last 30 days / the last 60 days / etc.
- A language filter that shows the common languages English/ French/ German/ Italian / Spanish with checkboxes so that none (same as all) or any number can be selected.

Every time a search is started or the refresh button is clicked, a DTOFilter will be created that includes all parameters of the filter bar and will be sent along to the backend.
The main element of the tool is the result list which shows the found literature in descending relevance. Every item shows the title, author, publication and publication date of the literature and every item is collapsible. That means by clicking the „plus"- button in front of an item extents that item and shows a summary of the according literature. Double clicking an item creates a new object of the class *LiteratureSearchDetailWindow* which represents a full information page of a literature item. It inherits from the class *Window* and in this way slides over the tool itself. The window shows again the title, authors and institution in which it was created as well as the full text if available or instead an abstract of the text. Furthermore it includes information about the journal or book in which it was published with the ISSN and publication date.
The bottom component is the paging toolbar which includes elements to control the paging and also includes buttons to print, email or create a pdf of the literature.

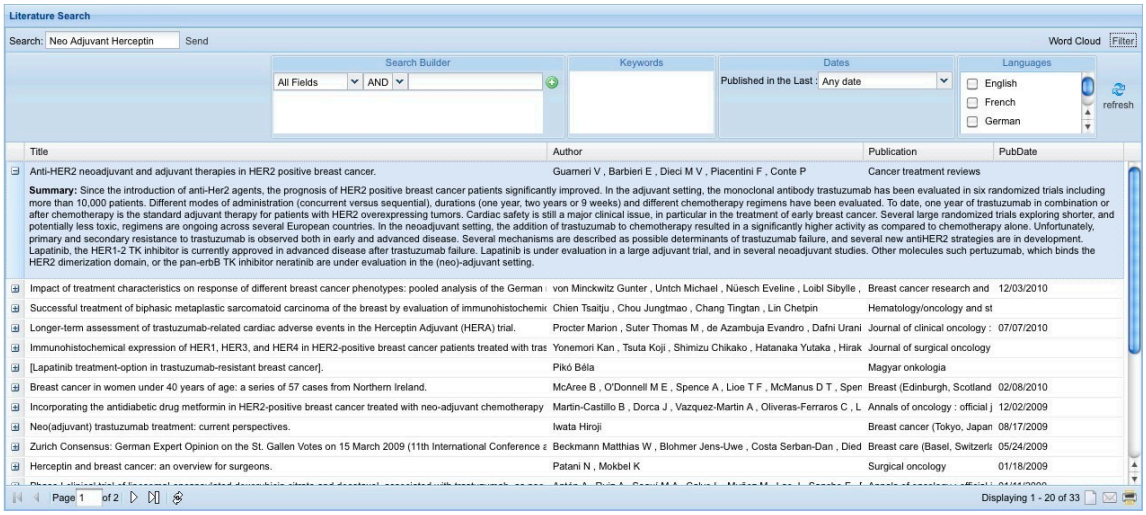


Figure 31: Screenshot of the implemented Literature Search.

4.5.4. Issues and Challenges

The first problem that occurred in the process of implementing the Literature Search tool was the fact that there are only a few databases out there which can be used by external tools. Most medical databases are on commercial use and have their own search systems. The first database which provides external access is also one of the biggest medical literature databases, the PubMed database [7]. But also PubMed doesn't support an open access to their databases, instead they offer different services which can be used to query the database. That means we didn't have to code our own search algorithm but use the PubMed algorithms. On one hand this is a big advantage because the algorithms used by PubMed are very professional and complex but on the other hand we can not transfer this search to other databases.

We implemented the tool as open as possible so that in the further development only search algorithms must be written for new databases but all other functions can be reused.

4.6. Interactive Word Cloud

A Word Cloud is a visualization of word frequencies in a given text as a weighted list. The higher the occurrence of the word the higher the weight and the bigger the font in which the word is shown in the visualization. In this way words that are occurring very often in the text are bigger than others and catching the eye.
The Interactive Word Cloud combines the features of a typical Word Cloud with two interactive options. One is to remove unimportant words, for example a date that occurs very often, and then recalculate the Word Cloud to get a new result in which now other words are shown bigger. The second option is to select one more words directly in the Word Cloud and start a new literature search that automatically uses these words in the search.

4.6.1. Motivation / Clinical Scenario

Time is a factor that is very rare in the work of clinicians. In this way the Interactive Word Cloud supports clinicians with a simple method of summarizing clinical terms related to a current search. This can be a search for literature, studies or any other text based search.

4.6.2. Specification

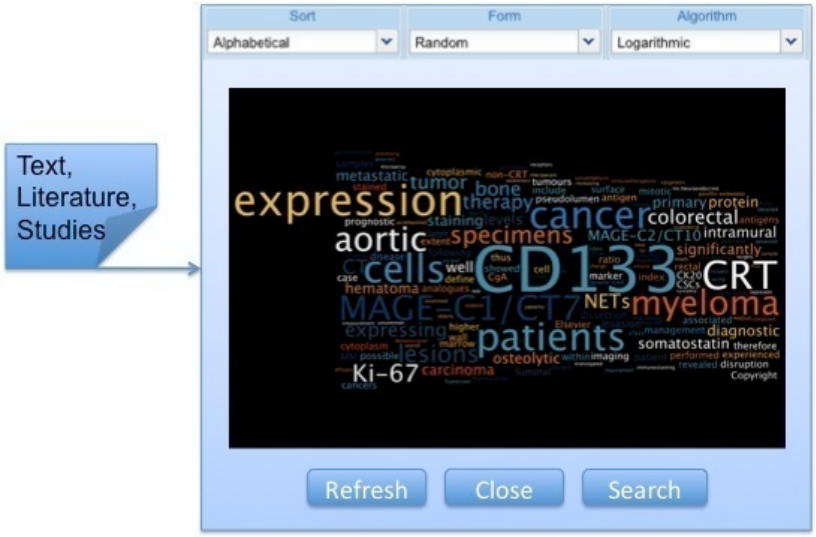


Figure 32: Overview of the Interactive Word Cloud.

The Interactive Word Cloud accepts all kinds of text based data as input. The texts can come from literature, studies or whatever is included in the application.

The interface consists of three parts. A filter bar in which the applied algorithm, form and sorting order can be changed, the Word Cloud itself and a button bar that contains a button for refreshing or closing the Word Cloud or starting a new search with the selected words.

The filter bar contains the following three possibilities with their options:

- Algorithm
  - Linear
  - Logarithmic
- Form (Currently only one form but should be extended in the future)
  - Random
- Sorting Order
  - Alphabetical
  - Descending Word Count

The changes of these options take effect after using the refresh button.

The Word Cloud allows two types of interaction. By clicking words with the left mouse button the words are getting selected an can be used to start a new search by clicking the search button. Clicking words with the right button marks them as removable and they won't be included in the next calculation of the Word Cloud.
A possible application of the Interactive Word Cloud is shown in Figure 33.

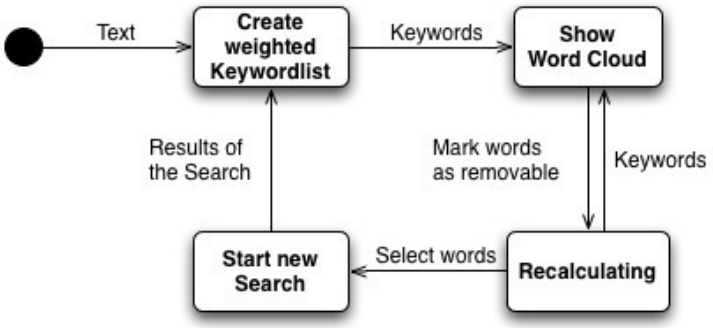


Figure 33: Flow Chart of using the Word Cloud.

4.6.3. Implementation

The Interactive Word Cloud (IWC) is implemented in a single class *WordCloud* that inherits the also self coded class *AbstractTextSummary*. We decided to write a superclass for the IWC because we assume that more other text visualization types are following and in this way we can reuse some methods like the sorting of the keywords. Furthermore, we can define the structure

of the subclasses and make methods required which helps to create new tools and to maintain them.

The IWC can be initialized in two ways, it can be created directly as an object of the class *WordCloud* what makes it to a simple LayoutContainer or as an instance of the class *TextSummaryWindow*. In this way it can be placed over any other tool or element and it gets a nice slide in and out effect. The IWC itself doesn't change no matters in which way it is used. At any time it consists of a toolbar with design options at the top, a button bar for interaction at the bottom and the Word Cloud itself in the middle of the tool.

There are three ways to modify the Word Cloud, switching the sort order between alphabetical and count based, changing the form (currently only a random form is implemented) or choosing a different algorithm. The algorithm is the heart of the Word Cloud it calculates the font sizes of the words by their word count. We implemented a linear algorithm which shows the size of the words in relation to their actual occurrence and a logarithmic algorithm that interpolates the counts of the words. The linear approach shows a good overview if the distribution of the counts is even, but if outliers exist they distort the result by highlighting these outliers and ignoring the smaller ones. In this way the differences in the word counts of the actual text are vanishingly small. The logarithmic algorithm balances this distribution by interpolating the word counts. This helps to get a distribution of font sizes covering the complete available range and inhibits the importance of outliers.

The Word Clod itself will be created from a list of *DTOKeywords* which consists of the name and the count of the word. Each of these words creates an instance of the class *WordCloudText* which inherits from the class *Text*. This special class has additional features like hover, selection and strikethrough styles. The selected and strikethrough elements are additionally stored in separate lists, so that the access is easier and faster. After the list of *WordCloudText* objects is created the words are placed in a panel according to the algorithm, form and sort options.

On the bottom of the screen is a small button bar which contains the components „Refresh", „Close", and „Search Literature". The refresh button uses the information of the option bar (algorithm, form, sort) and the strikethrough list to generate a new cloud. In this cloud the words of the strikethrough list are removed and the remaining words are presented in form of the selected options.

The close button usually just close the container but is overwritten by a slide out animation if the *TextSummaryWindow* is used.

The third button is the search for literature button which connects the Word Cloud directly to the literature search. It fires the *LiteratureSearchLoad* event with the words of the selection list as input that opens the literature search and automatically starts a search with the selected words.

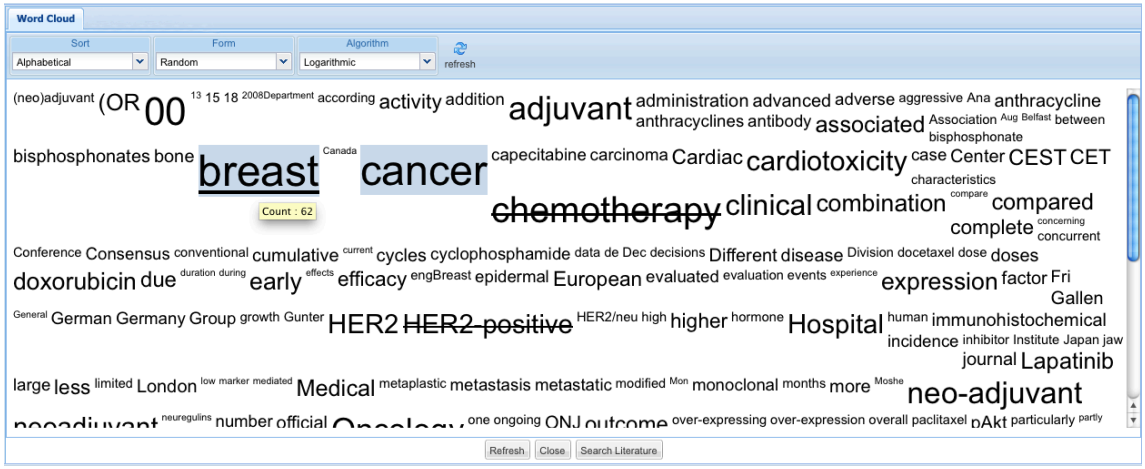


Figure 34: Screenshot of the implemented Interactive Word Cloud

4.6.4. Issues and Challenges

Word Clouds have two general challenges. Find the right algorithm that distributes the counts most evenly and find a form that shows the words in the clearest way.

We decided to implement two algorithms a linear and a logarithmic one to give the user the possibility to decide which one he prefers. The both algorithms are described in more detail in Chapter 4.6.3 Implementation.

The form is the second important options in implementing a Word Cloud. It is possible to place the words in a line, in a spiral or totally random. For the beginning we implemented a simple line design and kept other forms for future versions.

4.7. Tool Interaction

Building intelligent tools is one big step in the creation of a helpful web platform. Each of these tools helps to answer a specific medical question. But it is possible to even improve that usability by building connections between the tools. In this way, it is possible to use the output of one tool as input for another or simply switch from one to another. This creates a more intuitive flow of the application and reduces the unnecessary actions by the user. There are different ways of interaction, one way would be to request data or queries from another tool like the Visual Query Builder to modify the own data. A second possibility is the direct loading of one tool within another tool, as example starting a literature search by selecting a word in the Interactive Word Cloud. Figure 35. shows an overview in which way the tools can interact with each other and what types of data they are exchanging.

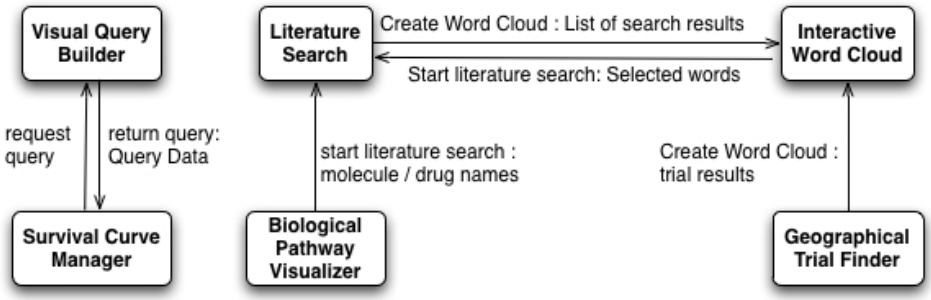


Figure 35: Interaction between the tools.

A challenge of this functionality is the changeability of the tools the requirements demand. It makes it impossible to implement static connections, since at their initialization none of the tools knows about the existence of the others. To solve this problem we used another functionality of the GWT that uses events and event handlers to communicate and exchange data between objects. All available event types are declared in the web application and can be called by any tool. The event handlers are declared within the specific tools, which has use for. If an event is triggered all tools that have a handler for this event will automatically execute their method for this event. If there is no handler in the complete web application for an event type nothing happens.

To make this process more understandable we look on the communication between Visual Query Builder and Survival Curve Manager as an example. Starting point is that the user wants to create a new curve in the Survival Curve Manager. To do this he selects the different values in the Visual Query Builder and uses the „New curve“ button within the Survival Cure Manager. Now the SCM triggers the *getQuery* event, which is sent to the main application controller that shares the event to each available tool. In this case only the VQB has a handler for the *getQuery* event. It executes the handler specific method, which creates a new query of the selected values. After the query is created the VQB triggers a new event *sendQuery* with the new query as content. Again all tools would have the possibility to handle this event, but since this query only sends data and doesn't load a tool in the foreground, it has only consequences for the visible tools. In our case the SCM that accepts the incoming event and uses the included query to create a new curve.

In a special case in which we had no handler for the *getQuery* event, the SCM would wait for the response until a defined time out and then shows the standard curve.

These events can also be used to load a tool into the foreground and give him a starting value. The literature search can be activated from each other tool by using the *loadLiteratureSearch* event. If this event includes a search parameter it starts automatically a literature search and shows the results. All currently available events are listed in Table 3.

Event Type	Description
Init	Initial Event for loading the Application.
EpiLoad	Event for loading the epidemiological tools.
PieChartLoad	Event for loading the Visual Query Builder.
SurvivalCurveLoad	Event for loading the Survival Curve Manager.
GetQuery	Event that requests the query from the Visual Query Builder.
SetQuery	Event that returns the created query to the active tool.
GeoMapLoad	Event for loading the Geographical Trial Finder.
LiteratureSearchLoad	Event for loading the Literature Search.
LiteratureSearchFilter	Event that requests the filter, which is used for the Literature Search.
PathwayLoad	Event for loading the Biological Pathway Visualizer.

Table 3: Available event types of the web application.

4.8. Backend

The backend is the control center of the web application. It is located on a web server and contains interfaces for accessing the different data sources, functions for pre-processing the incoming data and a controller that manages the incoming and outgoing events of the visualization tools. But why do we need to encapsulate the data processing out of the tools? The goal of each of our tools is only the visualization of data and not the processing of data. In this way the tools remain compact and can be loaded on the client side in a very short time. All the data processing and calculations are done on the server side and the results are sent to the client which also reduces the time the tools need to show the data. It gives also possibility to visual every data from every source if it is in the right input format. Another reason is the flexibility this design is giving us. Every function must only be implemented once and can be used in multiple tools. For example, we can use a search algorithm for the literature and also the same algorithm for the search of studies. Furthermore we can remove or add tools without affecting others.

In the decision of which technology is best for the backend, we decided to use Java Servlets. Since we already wrote the client side in GWT specific Java classes it is practical to use the same technology for the server classes what makes the code more legible and understandable. Moreover, the GWT supports the use of Java Servlets by providing a special communication framework, called GWT Remote Procedure Call framework (RPC). The RPC makes it easy for the client and server components to exchange Java objects over HTTP by using *services* on

server side and an automatically generated proxy on client side. Another advantage of the RPC is the ability of asynchronous calls. That means the client can retrieve data from the server in the background without interfering with the display or the tool.

The exchange of data between client and server is done by *DataTransferObjects*. These objects are simple serializable classes that only behavior is to store and retrieve its own data. They provide an easy way to collect the data in a single class that can be sent by services from the server to the client. All that is needed on client side are fill lists, selection boxes and other GWT components. Each object has at least one empty constructor and can be extended by an unlimited number of additional constructors. For every attribute in the object a getter and setter method is needed which change the value of the attribute.

Chapter Five: Evaluation of the results

The creation of visualization tools for presenting medical data was the main part of this thesis. However, we also wanted to evaluate these results and determine what we did good and what can be further improved. We did this in two steps: first we developed two additional tools that use the available functions and data of the backend but visualize them in a totally new way; and in the second step, we presented our web application to the three physicians we talked in the preparation phase before and asked them for their feedback.

5.1. Extensibility of the Application

One of the goals of the web application is to have a low redundancy of the developed functions. That means calculations and algorithms that are used by more than one tool are still written only once. A low redundancy also provides the application with a simple changeability and extensibility of new tools because only the presentation must be developed but the backend can be reused. For evaluating this we present two new tools in this chapter.

The first tool is a Survival Bar Chart Manager and is similar to the Survival Curve Manager. It uses the same input data as the SCM that means a statistical database and the output of the Query Builder but presents this data in a different way. It shows the amount of patients who „died on cancer“, died on other reason“, „left study alive“ or are „alive“ after 1 year, 3 years and 5 years in form of bar charts. In this type of visualization it is possible to see the progress of the cancer as well as the censored data, broken down by their type. Like in the SCM, the Survival Bar Chart Manager contains a control panel in which new charts can be created or already existing once can be modified or deleted. Figure 36 shows a screenshot of the Survival Bar Chart Manager with two different cancer groups.

Figure 36: Screenshot of the implemented Survival Bar Chart Manager.

The second tool is a text summarizing tool like the Interactive Word Cloud. Also in this tool the input is the same, it uses the same keyword function than for the Word Cloud to calculate the keyword counts. But instead of visualizing them in a cloud the tool generates a list of the occurring words and presents their counts in form of vertical bars. The higher the bar, the bigger the count of the word. Like in the IWC the list can be ordered alphabetically or numerically. Figure 37 shows a screenshot of the Keyword Count tool with the keywords for the search „Neo Adjuvant Herceptin“ ordered by their count.

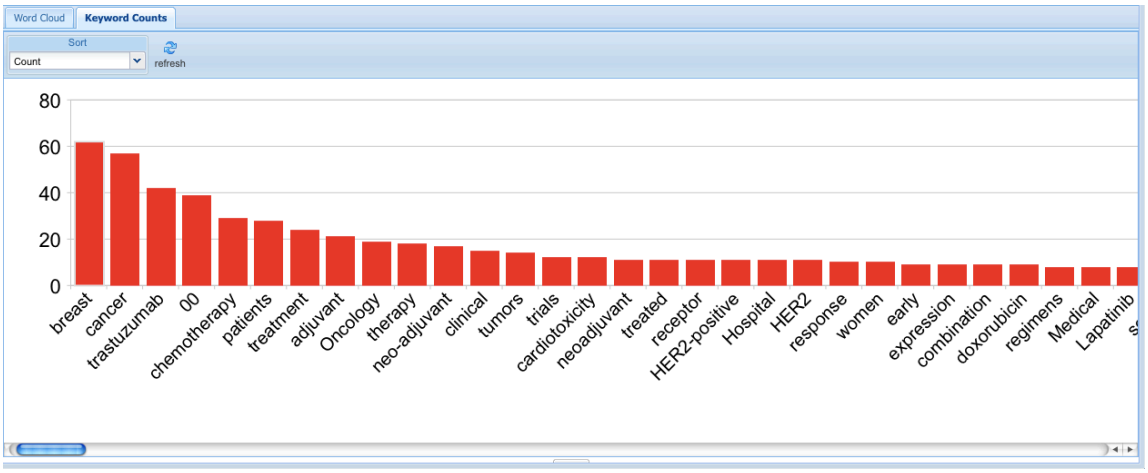


Figure 37: Screenshot of the implemented Keyword Count tool.

Both tools are using the existent functions and data types. In this way it was possible to create each of these tools in just a few hours. The development shows that now where the most functions are existing in the backend the implementation of new visualization types needs only few time but can show totally new sights of the data.

5.2. Interviews

As mentioned in chapter 2.2 cooperation with the end-user is unavoidable. In a second interview we want to present our finished web application to the three physicians we interviewed before to get their opinion whether we implemented the tools as they imagined or not. For the presentation a prepared system with a demonstration dataset was built, which enabled consistent basic conditions. The presentation followed the storyline of a hypothetical patient which we used to present every tool and showed in which way it could help the physician and the patient. After each tool we discussed the advantages and disadvantages of the particular tool. The following paragraphs are summaries of the physician‘s feedback. Transcripts of the complete interviews are in Appendix C.

Visual Query Builder with Survival Curve Manager

- Interesting interface and very intuitive.
- Great benefit in presenting the patient advantages of different therapy options.
- Not necessary for general decision finding but helps to validate the personal knowledge.
- Maybe an option for very uncommon cases.
- Very helpful for physicians or family doctors not intimately familiar with this data.

Biological Pathway Visualizer

- Very future oriented.
- Currently, not enough information are available to make valid decisions. And the information that are available information aren‘t evaluated.
- Not usable now but could be very important in the next five to ten years.
- If pathways can be used for treating patients a tool like this will be absolutely necessary because nobody can remember all the available pathways.
- Useful as eLearning tool for students or younger physicians.

Literature Search

- Nice and clear look.
- Very intuitive.
- More filtering options needed. Optimal solution would be different filters for different medical fields.
- Not really applicable to clinical practice but very nice for scientific research.
- Useful for very special cases.

Summarizing tools

- Nice idea and interesting interaction.
- Not sure about the benefit of such a tool.
- Wouldn’t be used in clinical practice but maybe to get an overview about a topic.

Geographical Trial Finder

- Nice components and look.
- Very high usability.
- Less relevant in Germany because of the smaller range.
- Nice for patients who wants to know where a study is offered.
- Results wouldn’t be of interest in such a tool because physicians have other sources.

General impressions

- Such tools would definitely be used.
- Not only for physicians but also for patients and family doctors.
- Should be available for everyone but with protected areas for medical personal.
- Nice looking application, especially the graphs that could help testing different medical scenarios.

Chapter Six: Discussion

The literature survey discussed in the beginning of the project provided us with four different data types, which could have great benefit from visualization. We used these data types to built six tools (chapter 4) that use different data to translate date into visual output. In these tools we implemented the newest types of interactive elements like animated charts that change their state dynamically depending on the incoming data. We built the tools to be intuitive for the clinician and provided automatization of the internal processes to a great degree. That means each tool uses the data of the current patient to automatically generate the output needed by the physician. This way, the physician can focus on decisions with all relevant information at hand. Furthermore, the tools aren't isolated programs that show only their specified output. Most of the tools provide special functions and output that can be used by other tools, so that an interaction between the tools is possible. For example, the Visual Query Builder doesn't only present statistical information, but also uses its graphs to create a query that can be used by other tools like the Survival Curve Manager. Knowing from the interviews that the needs for the tools depend on the medical field and the personal wishes of the physician, the tools are designed to be interchangeable. That means it is possible to integrate only the tools that are necessary for the work of the individual physician and hide those that are not needed without interrupting the interaction between the tools. For realizing this, the complete data processing and data formatting is encapsulated into the backend and only the presentational part is done in the particular tool. In addition, the communication cannot be done directly by method calling because the tools are unaware of the existence of the others. Instead we implemented an event driven communication system in which requests are broadcasted and every tool that has a handler for the event response to the request. In this way every tool constellation is possible.

After the development of the application we evaluated our results by testing the extensibility of the tools (chapter 5.1) and presenting them to the group of physicians we interviewed earlier (chapter 5.2). We first implemented two new tools by using the available data and functions but visualize them in a new way. This demonstrates the ease of extending the application with new tools since the necessary structures are available and only new ways of presenting them needed to be developed. Only one day of development was needed for each new tool. It is important to note that in this case we used only data sources that were already integrated into the tool. Switching to new data sources will likely require additional time for new tool implementation, but even in this case, given the implemented functionalities can be reused. For additional data sources, only the mapping of the new attribute names and values need to be adapted.

The underlying challenge for our medical decision support tools is the need of complete and valid datasets. If we take for example the Survival Curve Manager (chapter 4.2) the most desired feature of the physicians was the ability to compare different therapies and be able to show the different benefits to their patients. But even the SEER database doesn't provide the possibility of connecting patients with their therapies in their public datasets. Our approach shows the potential utility in treating future patients if the data on past patients and their performance was

available to clinicians in an intuitive and straight forward way. Identification of a clinical partner who has access to such kinds of data would unload the full potential for this tool. But the problem of providing the application with the right data wasn't the focus of this thesis. We aimed to create visualizations that can handle the available datasets and present them in a more informative way.

In the final interviews with the physicians we presented the complete application and showed them all functions and opportunities of this new type of technology. The physicians were very open to the idea of a decision support tool included in their complete workflow. But their idea of the utility of the tools differed slightly. The clinicians also foresee a use in helping the patient to understand the process by which therapy decisions are made. The interviews showed us that the tools went in the right direction and would find a place in the workflow of the physicians, but there are still challenges that should be addressed in future work. Regarding epidemiological data for example, there is a need for connections between the patient and his/her therapy. This would improve the statistical tools because it would be possible to show the different effects of the therapies and it would be possible to show the absolute benefit of a therapy within a specified patient group.

Chapter Seven: Conclusion

Data overload in medicine is a problem that's getting more and more important over the last decades. In the coming years the amounts of data will be increasing rapidly with the use of new data types like high-resolution sequencing. In this thesis we used a new approach of visualizing data to solve the problem of data overload. The tools that were developed in this thesis show the potential of this idea. Simple but intuitive interfaces were created to present large datasets in form of compact and easily understandable graphs. We also demonstrated the benefits of interactivity between different tool types by making their use even more intuitive and provide streamlined instruction for the user. During the evaluation we also demonstrated how easy it is to create new tools and to extent the whole application by using the available functions. From the feedback by the physicians to which we presented our results, we conclude that this type of presenting data has high potential and physicians would benefit from a decision support tool that provides them with effective visualization and interaction tools.

7.1. Future Work

In this thesis we developed first prototypes of tools that implement our ideas of presenting data in a new and helpful way in a clinical setting. During this development and also with input from a group of physicians we collected additional ideas for improvement and additional tools and functionalities. One example of this is extending the literature search tool by using a Search Result Clustering Engine like „carrot2"[25] that automatically organize collections of documents into thematic categories. In another example, additional graphical data representation modalities such as glyph based medical visualizations can be used. These are explored by T. Ropinski [26]. Finally, additional data sources can potentially be used. For example, the physicians suggested integrating clinical guidelines in our tools.

These tools can be provided to clinicians as a stand-alone application, but likely they will be used as components in a broader clinical decision support application. One possibility is integration of the tools into the PAPAyA platform [27] developed by Philips. This platform already integrates clinical tools around multiple molecular modalities, standard clinical parameters and contexts defined by clinical experts. Thus it's flexible architecture it's very easy to integrate new tools like the application created in this thesis.

References

[1] Ochs, Michael F., John T. Casagrande, and Ramana V. Davuluri. *Biomedical Informatics for Cancer Research.* Springer, 2010.

[2] Lindberg, DAB, and BL Humphreys. "Rising Expectations: Access to Biomedical Information." *Yearb Med Inform* 3.1 (2008): 165-72.

[3] Martin-Sanchez, F, I Iakovidis et al. "Synergy Between Medical Informatics and Bioinformatics: Facilitating Genomic Medicine for Future Health Care." *Journal of Biomedical Informatics* 37.1 (2004): 30-42.

[4] Müller, H, R Reihs et al. "Connecting Genes With Diseases." *2009 13th International Conference Information Visualisation*. IEEE, 2009. 323-30.

[5] Health, U.S. National Institues of. "Surveillance Epidemiology and End Results." <seer.cancer.gov>.

[6] Inc., Adjuvant! "Adjuvant! Online." 2003. <http://www.adjuvantonline.com/index.jsp>.

[7] National Center for Biotechnology Information, and U.S. National Library of Medicine. "Pubmed." <http://www.ncbi.nlm.nih.gov/pubmed/>.

[8] Editors, The PLoS Medicine. "Drowning Or Thirsting: The Extremes of Availability of Medical Information." *PLoS Med* 3.3 (2006): e165.

[9] McCandless, David. *The Visual Miscellaneum: A Colorful Guide to the World's Most Consequential Trivia.* Collins Design, 2009.

[10] Chen, Chun-houh. *Handbook of Data Visualization (Springer Handbooks of Computational Statistics).* Springer, 2008.

[11] Steele, Julie, and Noah Iliinsky. *Beautiful Visualization: Looking At Data Through the Eyes of Experts.* O'Reilly Media, 2010.

[12] Falkman, G. "Information Visualisation in Clinical Odontology: Multidimensional Analysis and Interactive Data Exploration." *Artificial Intelligence in Medicine* 22.2 (2001): 133-58.

[13] Chittaro, L. "Information Visualization and Its Application to Medicine." *Artificial Intelligence in Medicine* 22.2 (2001): 81-88.

[14] Chen, YJ, S Rajendran et al. "Multimedia Visualisation for Breast Cancer." (2006)

[15] W3C. "Html5 - a Vocabulary and Associated Apis for Html and Xhtml." 2010. Ed. Google Ian Hickson, Inc. <http://dev.w3.org/html5/spec/Overview.html>.

[16] Group, Stanford Visualization. "Protovis." <http://vis.stanford.edu/protovis/>.

[17] Bostock, M, and J Heer. "Protovis: A Graphical Toolkit for Visualization." *IEEE Transactions on Visualization and Computer Graphics* (2009): 1121-28.

[18] Fry, Ben, and Casey Reas. "Processing." <http://processing.org/>.

[19] Belmonte, Nicolas Garcia. "Javascript Infovis Toolkit." <http://thejit.org>.

[20] Crockford, Douglas. “Introducing Json.” <www.json.org>.

[21] Google. “Google Web Toolkit.” 2011. <http://code.google.com/intl/de/webtoolkit/>.

[22] Hanson, R, and A Tacy. *Gwt in Action: Easy Ajax With the Google Web Toolkit.* Manning Publications Co. Greenwich, CT, USA, 2007.

[23] Inc., Sencha. “Extgwt.” 2011. <www.sencha.com/products/gwt>.

[24] Solutions, DataStream Content. “Openphi.” 2010. *Open Census Geocoding*. <http://www.openphi.com/opencensusgeocoding.html>.

[25] Search, Carrot. “Carrot2 Open Source Framework for Building Search Clustering Engines.” <project.carrot2.org>.

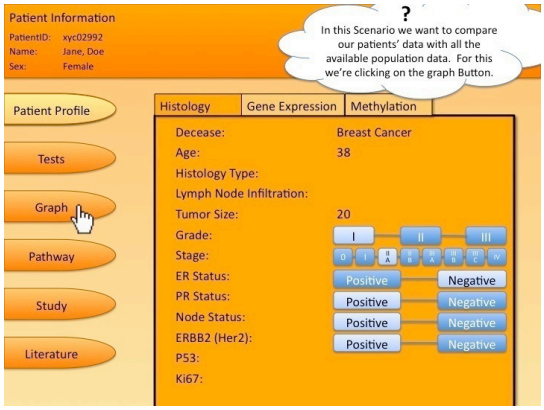
[26] Ropinski, T, and B Preim. “Taxonomy and Usage Guidelines for Glyph-Based Medical Visualization.” *Proceedings of the 19th Conference on Simulation and Visualization (SimVis08)*. Citeseer, 2008. 121–38.

[27] Janevski, A, S Kamalakaran et al. “Papaya: A Platform for Breast Cancer Biomarker Signature Discovery, Evaluation and Assessment.” *BMC bioinformatics* 10.Suppl 9 (2009): S7.

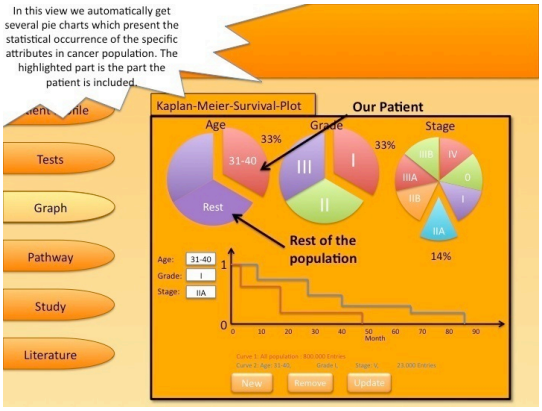
Appendices

Appendix A: Scenarios

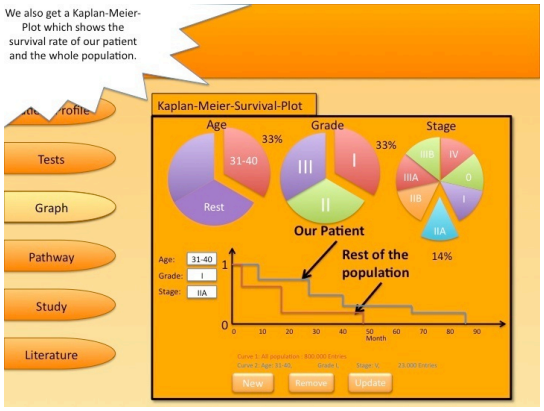
Scenario 1



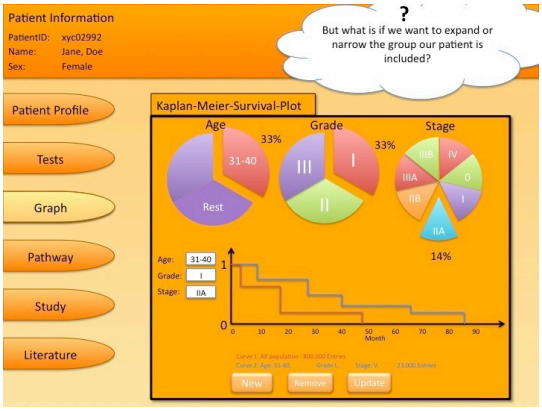
Slide 1



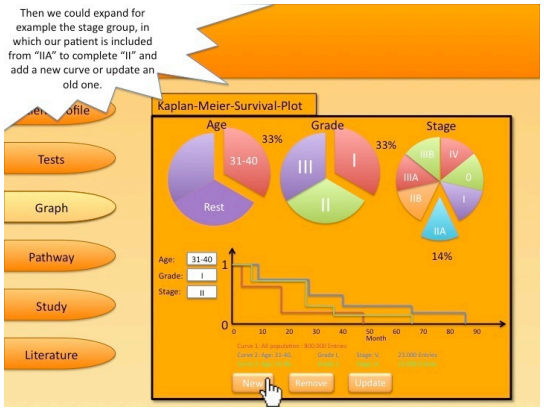
Slide 2



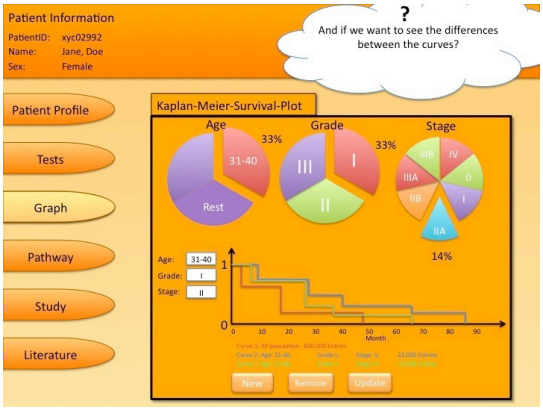
Slide 3



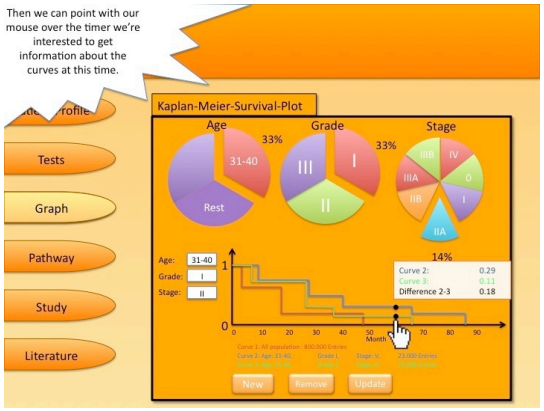
Slide 4



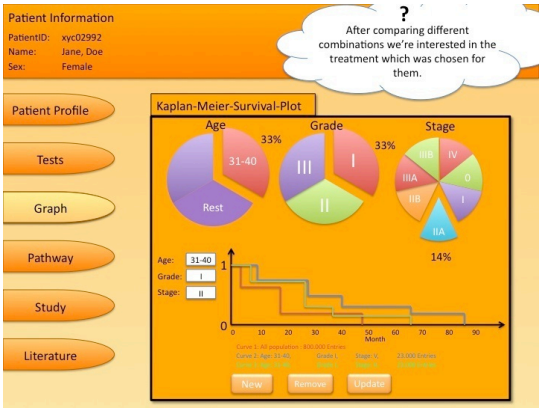
Slide 5



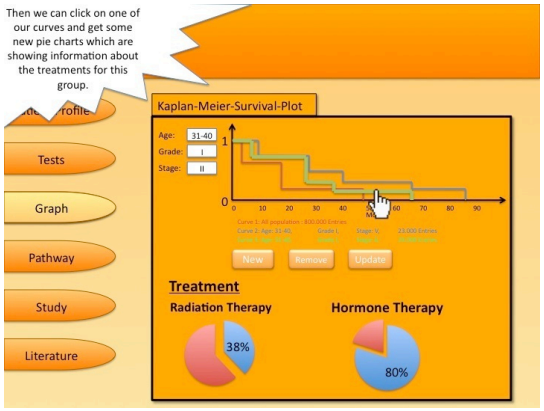
Slide 6



Slide 7

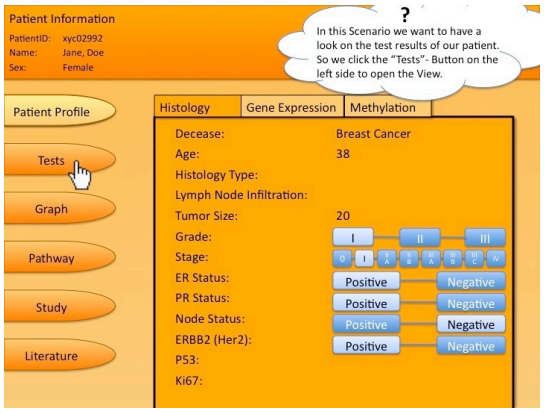


Slide 8

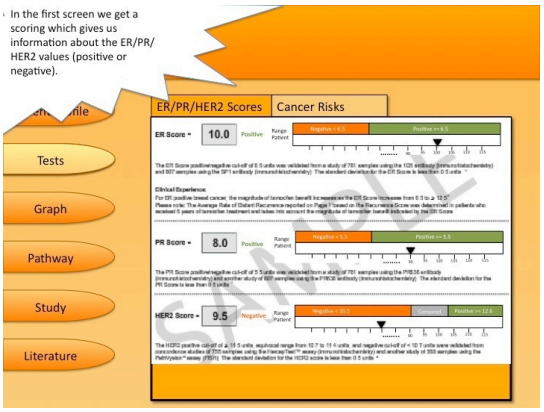


Slide 9

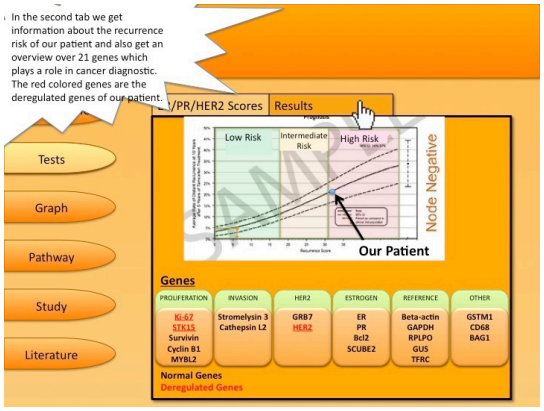
Scenario 2



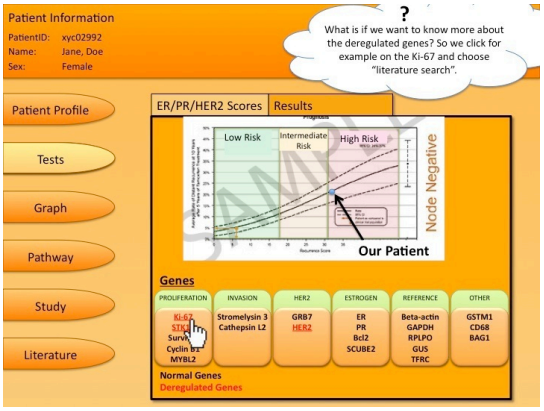
Slide 1



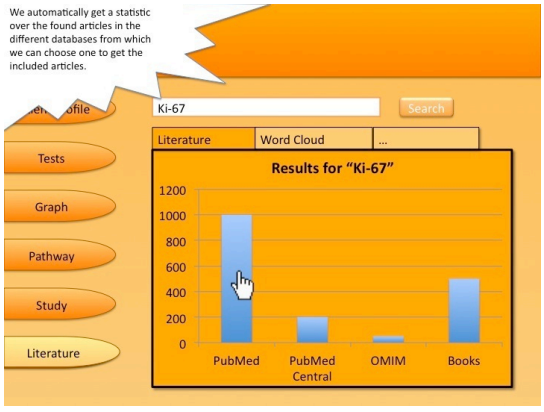
Slide 2



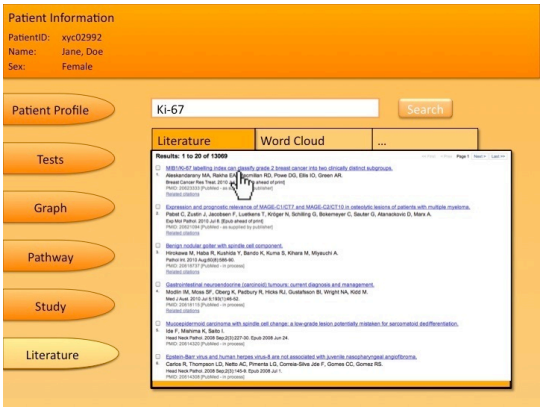
Slide 3



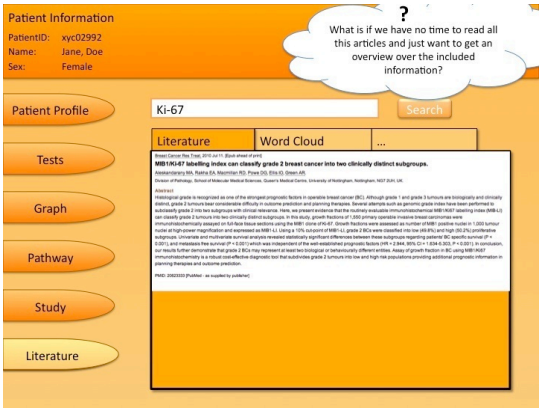
Slide 4



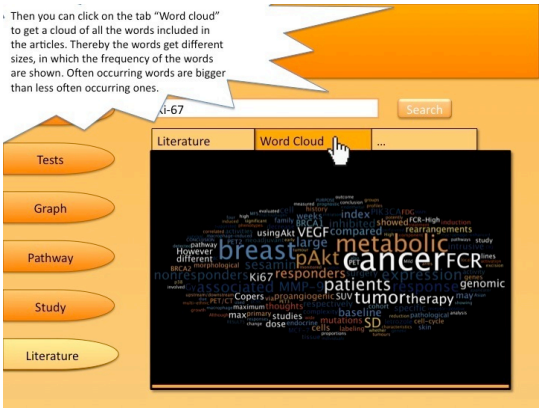
Slide 5



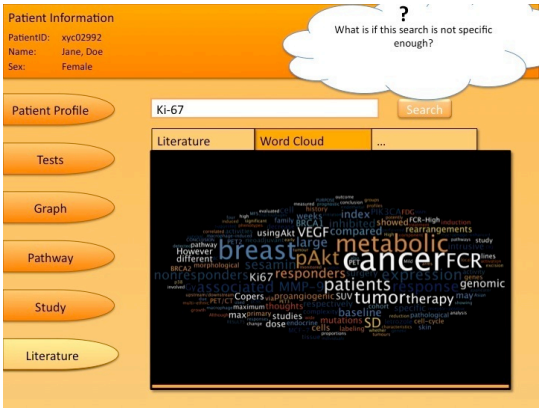
Slide 6



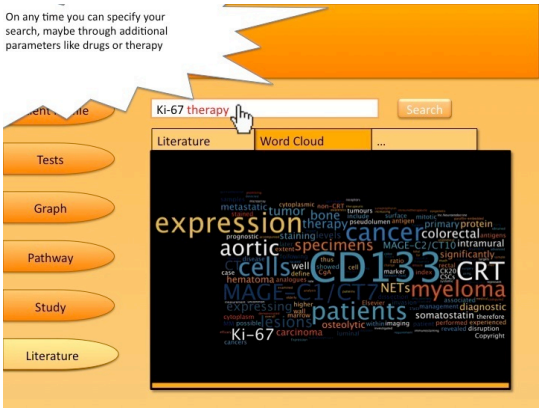
Slide 7



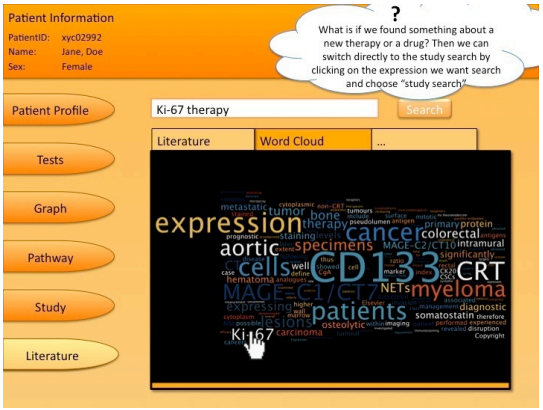
Slide 8



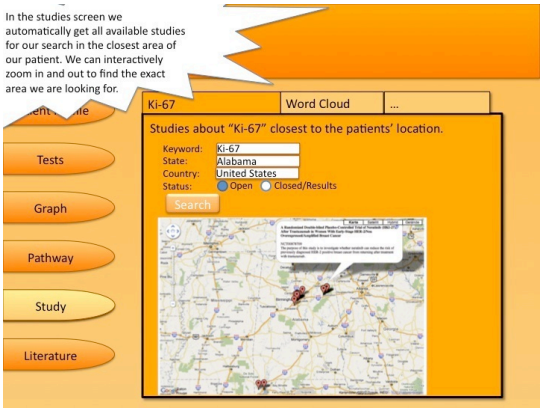
Slide 9



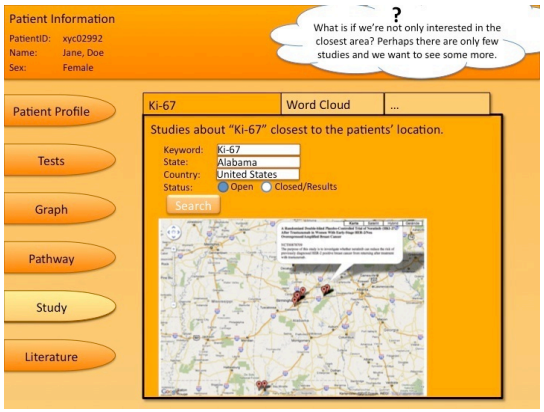
Slide 10



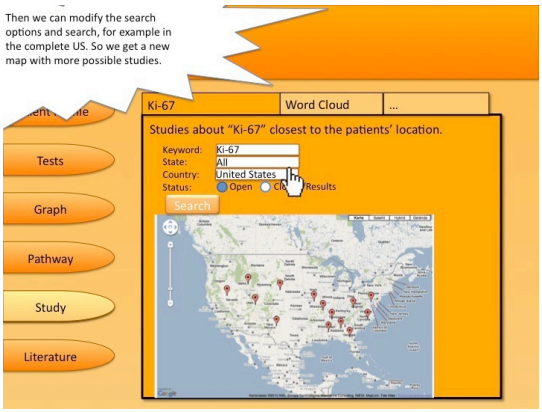
Slide 11



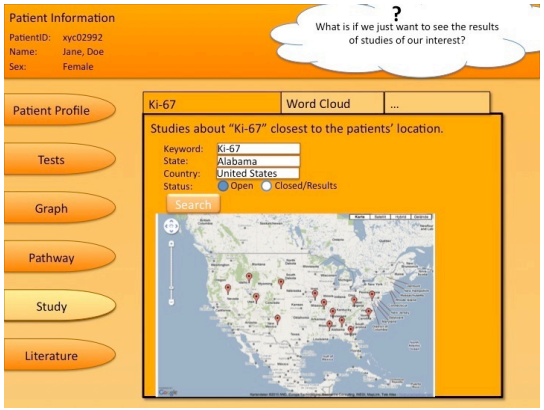
Slide 12



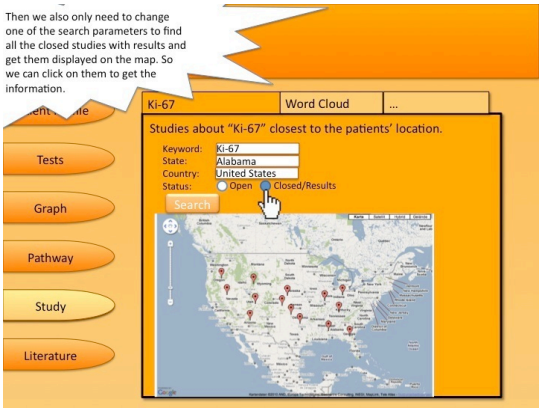
Slide 13



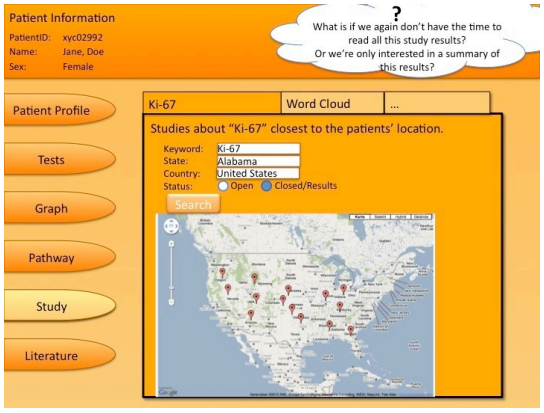
Slide 14



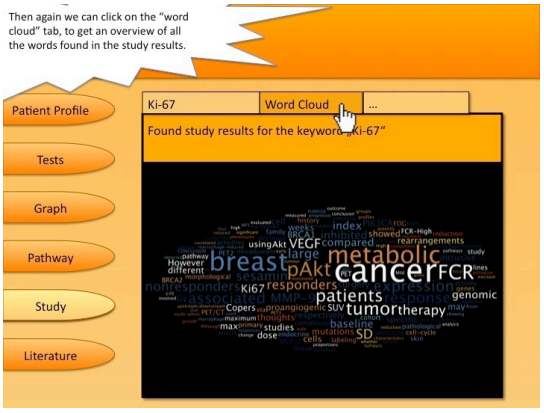
Slide 14



Slide 16

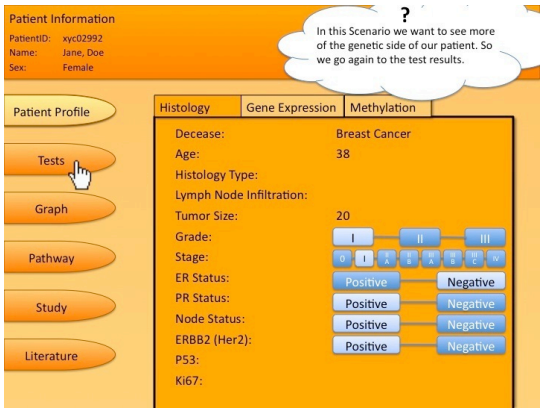


Slide 17

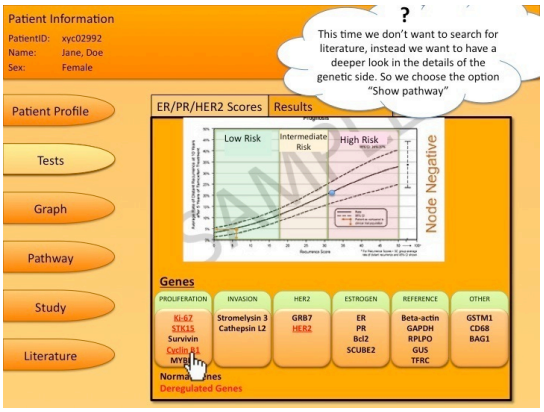


Slide 18

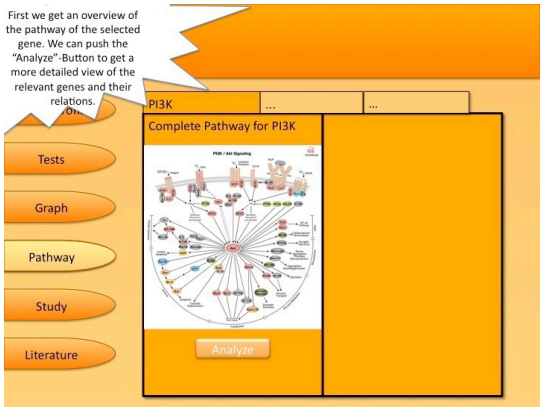
Scenario 3



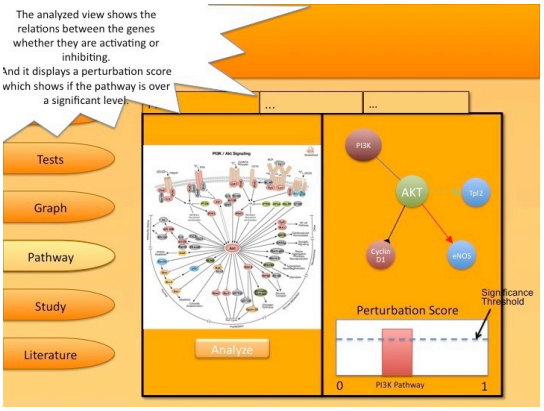
Slide 1



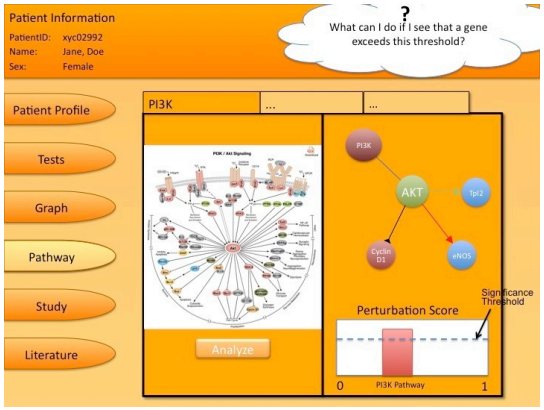
Slide 2



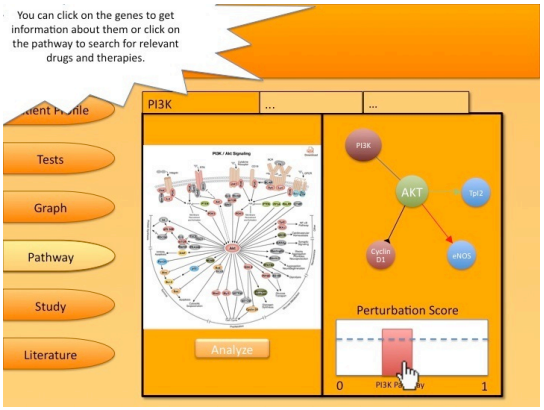
Slide 3



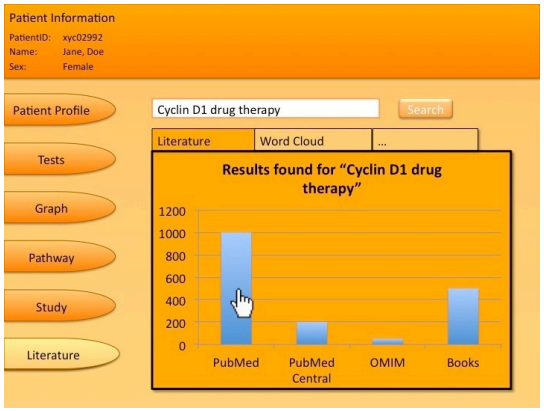
Slide 4



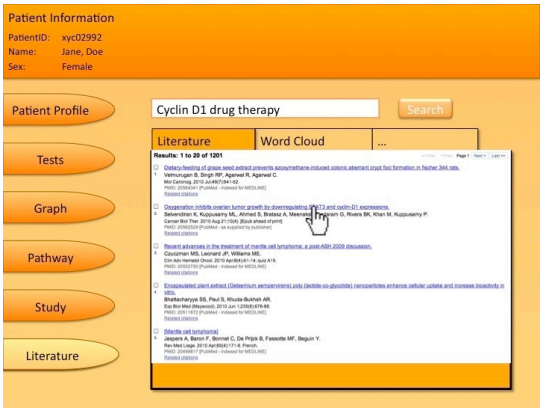
Slide 5



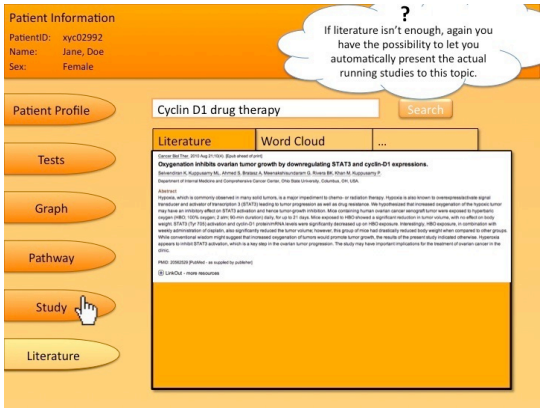
Slide 6



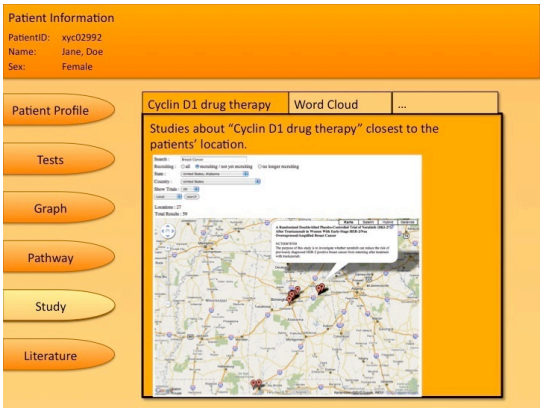
Slide 7



Slide 8



Slide 9



Slide 10

Appendix B: First Interviews

Interview Dr. Jörg Heil - 04.08.10

Interviewer: You're assistant physician at the Women Clinic of the University Hospital of Heidelberg?

Heil: Yes, exactly.

Interviewer: And in which medical field are you working?

Heil: I'm working in the fields of mamma carcinoma, diagnostic, surgery and system therapy (Chemotherapy).

Interviewer: What kind of data are you using in your daily workflow?

Heil: We're using general data like patient data, means age, medical conditions in general, medical history. Typical prognostic and prediction marker.

In this field there are huge amounts of data or marker or measurements whose significance is sometimes more or less unclear. The complete gene expression profiling and everything else in this field is only more or less evaluated. And the better the evaluation is, the higher is the likelihood that it is used in routine therapies.

Interviewer: Do you already have tools or programs that use these data types?

Heil: The only tool we're using in this context is Adjuvant! Online. But very infrequent. You can use it if you have very enlightened patients for the question - chemotherapy yes or no. No other tools.

Interviewer: These tools we're planing to develop should work as decision support systems. In this context one idea is to show survival curves of our patient in comparison with a bigger population. Would a tool like this be of interest for you?

Heil: What would you compare exactly? Therapies?

Interviewer: We're using the SEER database at the moment that includes parameters like personal attributes age, gender and also cancer specific attributes like tumor size or tumor marker. Unfortunately there are no therapies stored. But the purpose of the tool would also be to compare different therapies.

Heil: But then it's pretty similar to Adjuvant! Online. The only difference is that they don't show Kaplan Meier Curves but just the absolute benefits in comparison.

Interviewer: That's right. The idea is similar but we want to integrate more possibilities, not only benefits.

Heil: What do you mean with more possibilities?

Interviewer: We would implement different views of the data, in form of curves or other visualizations but we would also connect the tools among each other. As an example we could connect this tool with a literature search or a search for studies.

Heil: That would be great. So that you could immediately verify the evidence of these

results. I guess that's a good idea. Finally the question we're interested in is what is the initial risk of a patient, means we have diagnosis with stage and a type of tumor and a specific situation, age, etc. And what happens if we do nothing, what happens if we operate, what happens if we use radiotherapy or chemotherapy or something else. That's definitely the interesting thing and not only for the physician but to show the differences in talks with patients. I'm not sure what other software is available, but that's definitely an import field. That helps to make therapy decisions more transparent. And that's what everyone want.

Interviewer: Can you imagine any other data type that you're using and that could be visualized in any way?

Heil: In the end, I guess, the relevant criteria are survival, disease free survival, local recurrence free survival and metastases free survival. And the relevant data for this are the validated prognosis and prediction marker. And I'm sure there will be always new types of this markers. Currently the best validated data is still the histological data but sure, in the close future we will have others like gene expression analyzes and whatever.

There is really much research in this field but until something is validated so that you can make robust statements, still needs much more time.
Switching to Scenario 1. (Appendix A: Scenario 1)

Giving general information about the interface and the overview page.

Then switching to the population based graph tool.

Interviewer: What do you notice if you're looking at this tool?

Heil: Definitely the therapy dependency. That's the essential thing. If I can see that therapy A doesn't change anything on the expected survival time, I don't need to do it.

Interviewer: In Scenario 1 Slide 9 you can see the way thought of visualizing this data. The idea was to show the different therapies to one of the curves in new charts that visualize the success of this therapy. In this example you can see that radiation has only a 38% of success but hormone therapy has a 80% of success.

Heil: Okay. Yes that looks pretty good.
Switching to Scenario 2.

Explaining the literature search.

Heil: Looks like a classical search tool

Interviewer: Right. But the advantage is the connection to the other tools. It's possible to start a search from one of the other tools and our search tool automatically generates a search from the category you're coming from and the word you started the

search. But of course you can also modify the search parameters. We also planed to extent this tool with summaries. One possibility would be to present the found words in a Word Cloud that displays each word in a different size depending on the relevance in the text.

What do you think of that?

Heil: I'm not sure about that because I'm not really familiar with the functionality of such Word Clouds. I have no idea how good they are. But finally it's a tool to show the relevance and to make a relevance selection. And for this it's good.

Interviewer: From this point we could also jump directly to the clinical trials search and start an automatic search.

Explaining the study search.

Heil: If I understand that correctly you're target group were physicians so far. If you change, extent or adapt that target group to patients than this tool would be relevant in particular. Because physicians have the character that - the studies I do on my own will be delivered and less others. And if I send my patient to another than to one of my friends. That's why I think it's not that necessary for physicians. But for patients who can search on their own for studies their interested in it would be absolutely brilliant.

Interviewer: Maybe you're right that the use is bigger for the patient. We should think about that. But you agree that it's an approach of interest?

Heil: Yes absolutely.

Interviewer: Other ideas in presenting this data?

Heil: No actually not. Sounds good for me.

Switching to Scenario 3.

Explaining the pathway visualization tool.

Heil: I'm sure that's a very interesting approach but there are absolutely no data. As background information, our current procedure by advising therapies isn't to say, „ok we have this complex molecular biological mechanism and we have this drug that intervene there." That's not the way we're working. Instead we take drug A, doesn't matter how it's working, and give it to patient XY and look after 10 years if this patient lived longer than the patient without the drug. It doesn't matter why the patient lived longer. If they lived longer the next patients get this drug as standard. Shown absolutely simple and schematically. Maybe this approach in Scenario 3 is of interested for the people in the pharmaceutical industry who are developing but not for patient care.

Interviewer: This tool is definitely more future-oriented but we can see big potential in this type of data. And we think that pathways are a good way to visualize this data.

Heil: Yes of course. That's a huge are of development but currently it has absolutely no effect of the patient care.

Interviewer: That's the reason why we're talking of hypothetical data. We're aware of the fact that we still don't have all the requirements for this type of tool. But we assume that this will change in the next years.

Heil: Okay but we're talking of decades not of years. But it's an exciting idea.

Possibly you're right and it's the future. I can imagine that. Trastuzumab was the beginning of a multidirectional therapy and that's more or less the same this pathway model will do. I'm sure the future is to figure out the exact genetic problem and find an antibody that works. But after Trastuzumab there was nothing for a long time. There are many developments in this field but until these „products" are used in clinical routine procedures and especially in bigger amounts, I think we're talking about decades. But an exciting idea, definitely.

Interviewer: Finally I would like to ask you what you think of these tools and ideas in general. Useful approach or wouldn't you use it?

Heil: I think it depends on the support the tool can really provide at the end. If it's really helping in making therapy decisions transparent in a very detailed, nearly individual clinical situations then I'm sure it would be a sensation. That would be a further development and individualization of Adjuvant! Online. I'm sure that would be great and would be used in a wide field. But it depends on having a tool that can fast and visualized reflect an individual situation and helps in making considerations between no therapy and different therapies.

Certainly, that's a field that is very complex but that's the reason why such a tool would be so helpful.

Interviewer: Other comments?

Heil: The only thing is that I'm really surprised that you do that as a master thesis. You could spend much more time in this research because it's really a very important topic.

Interview Dr. Clemens Stockklausner - 06.08.10

Interviewer:	First a few questions to your person. Where are you working?
Stockklausner:	Currently I'm working in the pediatric clinic.
Interviewer:	And what is your medical field?
Stockklausner:	Children Oncology.
Interviewer:	What type of data are you using in you daily workflow?
Stockklausner:	We're using data like leukemias. Similar to the risk factors you mentioned in your description. We also have different risk groups for standard risk, intermediate risk and high risk and these groups are depending on genetic factors, the response, that means a defined area of success must be reached on a specific day of therapy. In other words, on day 8 the number of malicious cells in the blood count should be below a specific number, on day 15 the medulla should be more or less free from leukemia. And of course there are more genetic markers we are looking for. We have also PCR results that can detect minimal residual diseases in the process. Out of these factors we calculate a risk profile.
Interviewer:	And in which way is this data used at the moment?
Stockklausner:	At the moment we have this data in our mind and process them step by step. For example when we get a result we can look if the risk level changes or not. That's the way we are using it currently. This data aren't that complex but the amount is big.
Interviewer:	Would you prefer to get this data in a visual way to make it easier to work with it? Maybe you have a concrete idea?
Stockklausner:	Maybe in form of an interview, in which the system asks you questions like, „Is it a leukemia?" -> „Yes" „How is the response? Is it an A or B?" -> „B" In some kind of a tree structure in which you're going from branch to branch. And at the end you get the right indication for the right therapy. You could also also for the response on day 8. Either good or bad, if you choose bad you're automatically in the high risk branch. And from there you have new decisions. And there it's going to get pretty complex and tough to remember. It would be very nice for young physicians to go step by step through such a visu-

alization and could find the right therapy at the end.

Interviewer: Do you have any other idea of data types?

Stockklausner: No, not at the moment.

Switching to Scenario 1. (Appendix A: Scenario 1)

Giving general information about the interface and the overview page.

Then switching to the population based graph tool.

Stockklausner: I find this idea a little bit critical, because to calculate an exact prognosis is tough. Of course you can do that but I wouldn't push it to the foreground. You should know that neither we nor the patient always want to know the exactly prognosis. An integration into standard, intermediate or high risk is enough in principle for us. We wouldn't calculate statistical survival times for patients.

Interviewer: That means you can't see any advantages in getting statistical information about the possible survival times of your patient?

Stockklausner: Of course sometimes it's interesting for us and then we find this information but actually it's not necessary for us.

Interviewer: The idea behind this tool was not only to present survival times for the patient, but to compare our patient with patients of the population with similar parameters that differ for example in the therapy. So you could see what statistically happen if you choose therapy A or B.

Stockklausner: But the tumor therapy for children is a little bit more special, because there are standardized therapy methods that must be used. We're not allowed to look into a database what other people have already tried. That is more or less prohibited and is only allowed if there is a recurrence and we don't have an established therapy protocol for this. But that's a very special case. For this curiosities we're also doing literature surveys but I think that's a little bit too special.

Interviewer: That's depending on the available data but actually it's a good example where you could use our tool.

And you said you're doing literature survey for that special cases? That's perfect because as you can see in the other scenario we also provide a literature search tool which can help you searching for literature more automated.

Stockklausner: I guess it's very difficult to develop a program that automatically searches for the right literature. But it would be absolutely useful.

Interviewer: Of course the tool can't automatically know what you're looking for but it can use keywords from the topic where you are coming from to support you and it can provide you with helpful filters. Making the effort as low as possible.

Stockklaus- OK yes that seems possible. Or often you just want to find the reference to an

ner: established therapy method. For that you could create a link between the therapy method an the literature. Then you don't need to search for the literature or you could provide the text directly.

Interviewer: Additionally to the search we were thinking about a tool that summarizes texts. In the example in Scenario 2 Slide 8 you can see a so called „Word Cloud". That's a tool that visualizes the found expressions with their relevance. Words that occur more often are displayed in bigger letters.

Can you see the advantages of such a tool?

Stockklausner: No, not really. I can imagine a father that is searching with google for tumor information but I think an expert should have the basic knowledge that he doesn't need such a summarizing. Usually he's looking more specifically for articles.

This tool with the map, where you're searching for studies is also more interesting for adults. There is not really a need for. In the field of child therapy it's different.

Interviewer: What does that mean exactly? You don't use studies or you're using them in a different way.

Stockklausner: We're using only a specific type of studies but this studies are multicenter studies, means different centers are offering the same study. And I guess it is very difficult to create an overview about this centers. Only the particular center can give you the information who is offering the study.

Interviewer: Isn't there a central register that stores the available studies? In America there is the webpage clinicalTrials.gov that stores the studies in the USA and also international studies. And they're saving all information about this studies including descriptions, results and also the location.

Stockklausner: No, unfortunately not. Or, definitely not for children. I'm not sure how it is for adults, but for children I'm pretty sure there isn't anything like this.

Interviewer: The next tool we have is more future-oriented and also the data we use are more hypothetic. It uses molecular data to display genetic pathways. It should help to visualize the effects of deregulated genes in a pathway. Of course with looking on our patients data.

Are you already using genetic data or would you use it?

Stockklausner: Yes we're already using it but I would give you the contact information of one of my colleagues who knows more about it.

Interviewer: May I ask you anyway in which range you're using this data?

Stockklausner: I'm not using the data by myself, but they are used.

Interviewer: Why don't you use them? Because it's not your topic oder because you can't see the use of it.

Stockklausner: That's not so simple. We know that specific tumors have a molecular genetic risk profile. But the fact that a patient has a molecular genetic risk profile dissents a little bit the german gene diagnostic law. This law says that I'm not allowed to completely screen a patient for genetic defects, because that could be insurance relevant as an example. Instead of this I have to look on specific genes and for this the patient has to sign for every gene. All that means that the gene diagnostic law prevents the human to be classified completely as a risk profile.

Interviewer: But if you familiar with Oncotype DX, they're using only the 16 potential risk genes. Of course only for breast cancer but I think it's not necessary to check every gene.

Stockklausner: Of course, if we have a patient with a tumor disease and we recognize that the family is often affected by tumors, we're looking for this family risk genes and advise the family adequate.

Interviewer: But that means you're not reluctant to the ideas of gene profiles?

Stockklausner: No not reluctant. But I think that's dreams of the future because there isn't and I'm not allowed to do a global screening for risk factors. Furthermore this screenings are still very expensive and not really relevant. We don't do more tests because we have this or that profile.

Interviewer: The last question I have is more general, now where you have heard our ideas. Can you see the benefits of such tools and would you use them?

Stockklausner: Of course I would you use them. But it shouldn't be too unprofessional. It should be a specialized application and the purpose should be defined exactly. Then I could well imagine and think it's really good idea. But you couldn't develop one for all fields. Instead you would need an own tool for any medical field or tumor group and have to think very exactly about the needs of the particular.

Interview Dr. Florian Schütz - 10.08.10

Interviewer: You're senior physician at the Women Clinic of the University Hospital Heidelberg?

Schütz: Yes that's correct.

Interviewer: And in which medical field are you working?

Schütz: At the moment I'm the vice-chairman of the hospital of Salem.

In this position I'm reasonable for the whole field of the gynecology and perinatology. But the medical field I'm usually working in is the gynecological oncology and perinatal medicine.

I'm also a member of the AGO and working there in the development of guidelines in the breast commission of the german cancer society for breast cancer.

Interviewer: What types of data are using in the field of oncology?

Schütz: I think the main data we're using are study data, patient data, clinical data and histological data.

Interviewer: What is with other data like genetic data?

Schütz: Genetic data only in special cases. If we have genetic prepositions for example there is DSG 1 and 2. But that's very rare, only 5% of all mamma carcinoma patients.

Interviewer: That means genetic data aren't really useful for you at the moment?

Schütz: More in research. In this field we're using it quite a lot.

Interviewer: Do you already have any tools or applications that work with this data types?

Schütz: My brain.

Interviewer: So you don't have any „computer" program for that?

Schütz: No nothing special.

Interviewer: Would you use programs that support you in the work with this data?

Schütz: That depends on the fact whether I loose control over the data. But on the other side you have to know that this amounts of data that are stored in my mind were growing up with me. So that I think I can handle them. Of course, someone external could have problems with that.

Interviewer: So you think there is no use for you in using computer based systems?

Schütz: If I had to look up something more specialized like studies, then it's a different question. In this case I guess it would be helpful because of the huge amount of studies. I'm sure you know the PubMed database with all the journals.

Interviewer: Yes of course. That is a very important topic and we would definitely integrate the fields of studies and literature in our research.

Schütz: Yes I think that's the most important part, because it is so complex. The amount of data of the patient you have to know is actually very low. And the benchmarking, means looking for the quality you are producing, for this data there are already enough tools. But I think that's a different topic anyway. In your research you're not interested in cancer detection and quality assurance, right?

Interviewer: Actually we're interested in any kind of data. Our goal is to find ways to visualize this data. That's definitely not possible for any type of data but we have to figure that out. Today there are many ways to do that.

Schütz: Especially in tumor documentation there is a huge need for such tools. The problem is that there are already many providers and also national institutions that are instructed to develop tumor documentation systems but most of them had to understand that this subject is more complex than thought. Also the national center of tumor diseases in Heidelberg has promised lots of tumor databases. But untill now there isn't a big number of databases.

Switching to Scenario 1. (Appendix A: Scenario 1)

Giving general information about the interface and the overview page.

Switching to the population based graph tool.

Schütz: That seems similar to Adjuvant! Online. Which database is it using?

Interviewer: For the first implementation we'll use the SEER database, because it's a huge database with ca. 800 000 entries and it's free available. But the database should be easy changeable so that we could use a german database too.

Schütz: But in germany and also in Europe you don't have a good all-embracing database. The SEER database represents a huge population that covers everything. In germany you don't have such a population. Only the database which Mr Hölzl developed in Munich, but it covers only Bavaria. And that's already the biggest you will find in germany.

Interviewer: But we talk here about a research project. In this way we're sometimes thinking of hypothetical data and databases. We hope and assume that in the future the available data is getting bigger and is covering bigger populations.

Schütz: Maybe I could put someone in contact with you. A friend of mine is just visiting Mr Slamon at the UCLA. Mr Slamon is one of the most famous scientist worldwide for breast cancer and is researching especially in the field of the epigenetic. He is the great star of the oncology and a candidate for the nobel price. This friend is managing the mammography screening of the state of california. His name is Peter Fasching from Erlangen. He is also statistician and could tell you a lot of things to databases and visualizations.

Back to the Scenario and the graph tool.

Explaining how the survival curves are working.

Interviewer: Do you see the advantages of this tool for you?

Schütz: That depends on the tumor entity. It's very difficult to explain a patient a disastrous prognosis. You wouldn't do that with curves, you would speak with the patient and don't take the last hope. But in the field of breast cancer, if you know Adjuvant! Online, I use it to visualize the effects of a therapy in pie charts. Similar to your Kaplan Meier Survival Plots. Another benefit is that you can display the different therapies and show the patient in which way every therapy improves the prognosis. And in this, patients can see that doing a lot, improves the prognosis a lot. And at the end you can say that 80% of women are cured if they do everything, but still 20% are dying doesn't matter what we do. And we can't say at which point they are. I think women appreciate this visualizations and that's the important thing. The easier you can do it the better it is. I guess that your Kaplan Meier Curves are too complicated for a patient. And someone like me who knows a lot of patients and studies, doesn't need such visualizations because I know this curves. And I can calculate this numbers on my own, that's the reason why I don't use Adjuvant! Online anymore. I can't do it absolutely precisely but that's not necessary. But I think it could be very helpful for younger colleagues and patients.

Interviewer: You mentioned that the curves are too complicated. In which way would you visualize this data?

Schütz: Yes I guess it's too complicated in the way you have it in your slides. They only differ in the grade or in the age and that isn't of interest for the patient or the physician, because he doesn't has to know what could have happened if he had met the patient earlier.

Interviewer: But that's only an example. We would also include other parameters like therapies to show the differences between them.

Schütz: OK the integration of therapies would be much more of interest. In this case I can see the benefits of such a tool. But I would do it more patient oriented. That's very academic.

I also think it's very important to look to future technologies as you did it in the other Scenario with the pathways or methylations or gene expressions. I'm sure that will play an important role in the future. There are already tumor entities where it is used, in leukemia for example. Unfortunately not in gynecology but it will. The next will be Oncotype DX but it will take another 3-4 years. But I think you're on the right way with these tools. But you have to remember that

this amounts of data is not not always connected to each other. As an example you have four parallel studies with different gene expressions or methylations and each of them is about a different group of patients but never in comparison with another study. For example Oncotype DX has never been correlated in combination with a unimorphism from type 2D6. This are very special tests and shouldn't be seen additive.
So what would you do if Oncotype DX puts the patient into the intermediate risk level or into the lowest and another test would put her into the highest risk level?

Interviewer: That's true. In this fields a lot of research is needed. But as you said, it will be one of the topics in the close future and we're sure that such tools are also necessary to show the benefits.

Schütz: Of course.

And maybe I have another idea. If I'm right you want to show patient or the physician with this tool in which risk population the person is. For this you could additionally implement the quality of the probe and the classical risk factors as you showed in Scenario 1. And then you could make a summary how many factors result in a good and how many result in a bad prognosis.
Switching to the literature search and the Word Cloud

Interviewer: Are you interested in such ways of summarizing text?

Schütz: Oh yes, sure. Definitely a nice add-on.

Interviewer: What do you currently use for literature searches? PubMed?

Schütz: Yes.

Interviewer: The next tool is the search for studies. I learned it's more difficult in germany because there isn't a all-embracing study register.

Schütz: That's right. It would be great to have something like a centralized study register but that's a huge mission.

Interviewer: In the US there is the webpage clinicalTrials.gov that manages the studies.

Schütz: Meanwhile we have something similar in Europe called the EOSTC (European Organisation for Research and Treatment of Cancer). You have to register every study there, because without that you don't get an Ethics Commission Vote. So we already have that correspondingly to clinicalTrials.gov, but in germany itself there is no study register or something similar.

That's also something that is very annoying for our patients.

Interviewer: The purpose for this tool is to make a filtered search for the specific parameters of our patient and to search for studies in a specific location. So that we only find studies that are relevant for the patient. We think that this would be very helpful for patients and for their physicians too.

Schütz: Yes I think so too. But in germany you have the problem that every physicians are a little bit selfish and lazy in connection with studies. So they prefer to keep the patient for themselves to earn money. I guess that's similar in America but maybe the patients are a bit more self-sufficient. That was my impression.

Interviewer: Okay. Maybe we shouldn't take germany as an example. Another idea we had for this tool is the possibility to integrate the results of the studies if available. That's not that important for the patient but we hope that physicians could have a benefit of this.

Schütz: Yes I think that's a nice idea.

Interviewer: Okay that was the last tool. Now you have heard a lot of things about our ideas. Could you imagine something else that we forgot?

Schütz: No, not at the moment.

Your tool that is similar to Adjuvant! Online looks really good and I'm sure it's still possible to optimize it. And if you integrate the studies and I'm sure you have the better conditions in America for that, you're definitely on the right way.

Interviewer: Finally, can you imagine the advantages of these tools?

Schütz: Yes of course. Especially for younger physicians. Also the literature search seems a nice idea.

Appendix C: Second Interviews

Interview Dr. Clemens Stockklausner - 25.11.10

Beginning with the presentation of the Visual Query Builder.

Stockklausner: Would be interesting if we could compare different therapies, as example with stem cell transplantation and without stem cell transplantation. If we had a button: Show data of patients with the same risk strategy with a more intensive therapy. And also might break down why the mortality is higher with or without a different therapy. Are they dying from the poison therapy or because the tumor is coming back? Does the tumor coming less back because of the more intensive therapy or is the result the same since many patients are dying of this therapy? And that's important because patients are asking exactly these questions. And in this case such a visualization would be helpful. Sometimes we have to make decisions at a threshold where you have to decide if you take the more intensive therapy with the higher mortality but then this effect must be at least compensated by less recurrences.

Interviewer: So you would use such a tool?

Stockklausner: Yes I could imagine that.

You need huge database with information from the literature like Marvin, a computer based system.

I think it's less important for us because we know these curves but thinking of the talks with the patients it would be good.

As an example I would show a curve with a weaker therapy and with a more intensive therapy and then only the recurrences of the tumor. In this case I can see that the recurrences is going back. And then I could click on the next screen and show the mortality of this therapy. In this way I can see the other side. By overlaying both curves I can see the advantages of the more intensive therapy for example. In this way I could imagine that.

Biological Pathway Visualizer

Interviewer: Are you using genetic data?

Stockklausner: Leukemia with BCR-ABL, but I know that there is a antibody.

Interviewer: What do you think of visualizing pathways in this way?

Stockklausner: I think there is more use as eLearning or for students and younger physicians, that you can click through these pathways.

That's not relevant for us since only less genetic influence.

Literature Search

Stockklausner: That looks good for Leukemia. Because there are many different gene constellations and many new antibodies in research. And you can't always know all of them. But I would need more filters like age groups under 10 and so on. So that the results are getting less and more specific.

Maybe you have to create different filter groups for the different medical fields. But that would definitely helpful.

Text Summarizing Tools

Stockklausner: I'm not sure about that. Looks nice but I'm not sure of the use of it.

Geographical Trial Finder

Stockklausner: In germany you have the website Kinderkrebsinfo.de which makes the studies in germany available in the field of children oncology.

But I think it's less relevant in germany because it is small and easy manageable. In the field of children I can't see potential benefits. If you would expand it to Europe for example then it's getting more interesting.

Interviewer: What do you think of integrating the results of the studies into this tool?

Stockklausner: In the field of children oncology it's a little bit different. I have one study which is representative for whole Europe. If a new type of Leukemia is appearing I can't choose between study A or B. I see the relevance more in the therapy of adults. In this case I have different possibilities and different medical centers.

Interviewer: Do you generally think these tools are useful and would you use them?

Stockklausner: Yes of course. You can see that others have similar ideas, for example in the field of studies.

I also could think of a good use in semi professional fields, so that parents and patients have a good possibility to inform themselves.

As an example they could look at the pathways to see why exactly they get this drug. That could help as an explanation.

You have also developed a great platform which can be used by interested persons affected or the family doctor who doesn't has to handle such diseases every day.

From my point of view, it's my job to have the information when a child with Leukemia is coming to us. But the family doctor who has a child with suspicion of Leukemia in his office must look who is responsible for that cancer

type. And then he could offer the parents different centers. After that the patient comes back with the recommendation and drugs. Maybe the patient is asking his family doctor why he is getting exactly this antibody and the family doctor can teach himself by using these visualizations because that are new drugs and people mostly don't know them. And then he could show this information to his patient in a nice visualization.

It's unlikely that it's allowed to make all data available for everyone but you could use a login like DocCheck (http://www.doccheck.com/de/) and make the different tools available for different types of groups. Creating a private, password protected area and a public area would address a much bigger range of people.

Actually I think that's a really good thing if you substantiate the complete thing and create different groups. A group with the pathways for us which must be always up-to-date and has good filters and a good database as backend, with password protected data. The literature search and trials could be public and available for everyone.

Interview Dr. Jörg Heil 25.11.10

Beginning with the presentation of the Visual Query Builder.

Heil: Looks very interesting.

Interviewer: Do you see the benefits of such a tool in your daily workflow?

Heil: Yes of course. In this moment the tool shows only the survival probability regardless of the therapy. Only this doesn't really improves the current procedure.

It will be more interesting if you can say: What are the benefits of an antihormonal therapy, how does the curve changes. What are the benefits of a chemotherapy or maybe different types of chemotherapy. That would be absolutely great.

Of course similar to Adjuvant!Online but much more funded.

Interviewer: Do you have other ideas of using that data?

Heil: For the therapy dependency logically.With the background of making things nicer, faster and more understandable in the patient talk. Meaning you could present curves to the patient which shows what happens statistically if she doesn't use a therapy and what happens if she uses one therapy or another or if she combines different therapies.

Interviewer: That means you see the use of the tool for helping patients to understand and less to help you in your decision?

Heil: Exactly. But that's an interdependence. If I'm convinced of a therapy decision I can explain it to the patient in a better way. That means in cases in which I'm not absolutely sure about the effect of a therapy I can make a quick look in that tool and see for example: The absolutely benefit is only 3% in the next 5 years. In this way you can control yourself sometimes about your feelings and thinkings of therapy benefits.

Only very few physicians can remember all these numbers. And I think the absolutely benefit is overestimated by many of them.

In this case, personal experiences do not lead to anything. You're treating maybe 100 women and have to do post-treatment about 10 years for each of them. That's only a small group of possible cases so most times you have absolutely new conditions.

You can get much better assessments by using studies or using tools like Adjuvant!Online.

Biological Pathway Visualizer

Heil: That's definitely the future. The problem is that our system obliges the use of randomized controlled studies. These studies need 5 years until they can even begin and again 5 years until you get first results.

I think until we used genetic data in therapy decisions it will take more than 10 years because our demands about the evidence is very high.
Trastuzumab was the first breakthrough in the anti body therapy nothing but after that there was more or less nothing.

Pertuzumab and Avastin and all of them don't have the breakthrough till now.

Interviewer: What do you think in general of this idea?

Heil: If these information are available and you can interfere in a complex pathway then it's very important. Because then the therapy decision gets much more complex as it is today and you definitely can't remember all this data. Impossibly.

Interviewer: So you don't use any genetic data at the moment, right?

Heil: No we don't. At least not officially. We don't use it for making decisions but in special cases we're telling the patient the way of the guideline and what we results we have on genetic side and how it could change the therapy in the next years. So she can decide what she want to do. But that's in the earliest days.

But first the experiences of the cancer on molecular level must be generated. That means we need kits with which you can analyze the patients' genome in a fast and cheap way. And exactly this will be relevant for the clinical practice if it can be realized for a wide range.

Literature Search

Heil: PubMed is irrelevant in the clinical practice. Means for patient care because it is absolutely guideline oriented and if the guideline can't be used the decisions will be done by using the studies of the center. Of course everybody is using PubMed but more for scientific research or if you have a case which is absolutely special.

I'm not using it for the clinical practice either. Only for scientific research.

Summarizing tools

Heil: Definitely not in clinical practice but to get a general overview about a topic it seems very nice.

Geographical Trial Finder

Heil: That looks interesting. Specially for knowing where you can find the trials.

Interviewer: Do you use studies?

Heil: Yes sure but not in the way that I say, okay I have a patient with these criteria, there are five fitting studies. You recommend the study which is at you center. I think for 90% of patients it is not necessary. There are special situations, mostly hopeless cases for which you don't have a study and in which say: If you really want to do something there is a last opportunity.

Interviewer: At the end, do you have any notes or ideas?

Heil: I think applicable in closer time will be the survival curves if you connect them with the therapies. The pathways are extremely futuristic but not unrealistic. The complete thing looks very interesting and I'm excited about the result.

Interview Dr. Florian Schütz 30.11.10

Beginning with the presentation of the Visual Query Builder.

Interviewer: First impressions of this tool?

Schütz: I have the data in my mind and don't need it visualized but if I want to explain the benefits of the therapies to a patient, these graphs are very nice. If you make it available to family doctors who doesn't work with this data every day it would be helpful. I guess you're using a similar idea like Adjuvant! Online.

Interviewer: Do you have ideas what we could change or expand to make it better?

Schütz: No I think it's pretty good in this way.

Biological Pathway Visualizer

Schütz: It seems very interesting to me. I think that's the future. Definitely a very future oriented tool but I'm sure it will need five to ten years until it will be of importance.

Link to Literature Search

Schütz: Very nice and very helpful.

Interviewer: Ideas how to make it better?

Schütz: No very good in this way.

Summarizing tools

Schütz: I can't see the use of that tools.

Geographical Trial Finder

Schütz: Great for patients. I often have patients who are calling me and asking whether we offer this or that study. I think there is less use for family doctors. Either you know where a study is offered or you have your contacts which you can call and ask where you find the study. And there are competing studies too. So a breast center would usually try to offers studies for every situation but no competing studies. That means I don't have two similar studies with the same patient collective where one study disturbs another.

Interviewer: That means you think it's only of interest for patients?

Schütz: Yes I think so. Because family doctors normally aren't interested and not included in studies. And physicians usually know the studies or someone else who knows them. Although in special cases it could be interesting to find facts about substances that are only available in the US or in Swiss for example. As an example I have a patient who I referred to America because there she could get a therapy in a study which wasn't available in Europe yet. This study for example included foreigners, the same study existed in Britain and this study included

only british people. This is a problem you should think of.

Interviewer: Would it be helpful if we integrated the results of the studies directly into the tool?

Schütz: I don't know. I think there are already other possibilities. Maybe if it would be a very simple thing but it depends on who you want to address. If you want to have the experts it's difficult because most of them have other sources where they get this information. If you address the patient it's possible that you expect too much and the family doctor or simply a physician of a different medical field would also have difficulties in understanding all the facts.

I would advise you to create specific tools for the particular situation of the patient by integrating the different guidelines available in the world.

So you could show the german guidelines to your specific patient but you can also visualize what the american or british would do. Because that's something really difficult to find and you have to skim a lot of sources to get the information. That would be great. You would have all advices with one klick.

Interviewer: Summarizing in short words what do you think is good and what could be better?

Schütz: What I really like are these nice looking graphs and also the possibility to go through different medical scenarios with the help of these tools, especially in talks.

But also the idea of the guidelines sounds great for me and should be realized.

Appendix D: JavaScript code for a pathway

```
var labelType, useGradients, nativeTextSupport, animate, ht;

(function() {
  var ua = navigator.userAgent,
      iStuff = ua.match(/iPhone/i) || ua.match(/iPad/i),
      typeOfCanvas = typeof HTMLCanvasElement,
      nativeCanvasSupport = (typeOfCanvas == 'object' || typeOfCanvas == 'function'),
      textSupport = nativeCanvasSupport
        && (typeof document.createElement('canvas').getContext('2d').fillText == 'function');
  //I'm setting this based on the fact that ExCanvas provides text support for IE
  //and that as of today iPhone/iPad current text support is lame
  labelType = (!nativeCanvasSupport || (textSupport && !iStuff))? 'Native' : 'HTML';
  nativeTextSupport = labelType == 'Native';
  useGradients = nativeCanvasSupport;
  animate = !(iStuff || !nativeCanvasSupport);
})();

var Log = {
  elem: false,
  write: function(text){
    if (!this.elem)
      this.elem = document.getElementById('log');
    this.elem.innerHTML = text;
    this.elem.style.left = (500 - this.elem.offsetWidth / 2) + 'px';
  }
;

$jit.ST.Plot.NodeTypes.implement({
          'round-rect': {
            'render': function(node, canvas) {
              var width = node.getData('width'),
                  height = node.getData('height'),
                  pos = this.getAlignedPos(node.pos.getc(true), width, height),
                  posX = pos.x,
                  posY = pos.y,
                  deregulated = node.getData('deregulated'),
                  drug = node.getData('drug');

                  var ctx = document.getElementById("infovis-canvas").getContext("2d");
                  ctx.beginPath();
                  ctx.moveTo(posX, posY);
                  ctx.quadraticCurveTo(posX-10, posY+(height/2), posX, posY+height);
                  ctx.lineTo(posX+width, posY+height);
                  ctx.quadraticCurveTo(posX+width+10, posY+(height/2), posX+width, posY);
                  ctx.closePath();
```

```
                var gradient = ctx.createLinearGradient(posX, posY, posX+width, posY+height);
                if(drug != 0)
                {
                        gradient.addColorStop(0, "#7a7d11");
                        gradient.addColorStop(1, "#f7fc27");
                }
                else if(deregulated == 'true')
                {
                        gradient.addColorStop(0, "#ff7100");
                        gradient.addColorStop(1, "#fcc79c");
                }
                else
                {
                        gradient.addColorStop(0, "#15428B");
                        gradient.addColorStop(1, "#99BBE8");
                }

                ctx.fillStyle = gradient;
                ctx.fill();
                ctx.strokeStyle = '#000';
                ctx.stroke();



        );

function init(){

    var infovis = document.getElementById('infovis');
    var w = infovis.offsetWidth - 50, h = infovis.offsetHeight - 50;

    //init Spacetree
    ht = new $jit.ST({
        //id of the visualization container
        injectInto: 'infovis',

        //By setting overridable=true,
        //Node and Edge global properties can be
        //overriden for each node/edge.
        Node: {
            overridable: true,
            'transform': false,
            height: 30,
            width: 60,
            color: "#00008b",
            type: 'round-rect'
,

        Edge: {
            overridable: true,
```

```
    lineWidth: 3,
    dim: 20,
    type: 'arrow'
,


offsetY: h/2,
levelsToShow: 10,
levelDistance: 70,
siblingOffset: 30,
orientation: 'top',
//Change the animation transition type
transition: $jit.Trans.Quart.easeInOut,
//animation duration (in milliseconds)
duration:1000,
//Enable zooming and panning
//with scrolling and DnD
Navigation: {
  enable: true,
  //Enable panning events only if we're dragging the empty
  //canvas (and not a node).
  panning: 'avoid nodes',
  zooming: 20
,
Label:{
  type: 'HTML',
  size: 20,
  color: '#fff'
,
PopUp:{
    enable: true,
    type: 'auto',
    //add positioning offsets
    offsetX: 20,
    offsetY: 20,
    //implement the onShow method to
    //add content to the tooltip when a node
    //is hovered
    onShow: function(popUp, node, isLeaf, domElement) {
        if(node.getData('drug') != 0 && node.getData('drug') != ""){

            var html = "<div class=\"tip-title\">" + node.getData('drug') +
                    "</div><div class=\"tip-text\"><br> <a href=\"\" on-
                    click=\"literatureSearch('" + node.getData('drug') + "');\"> search
                    for Literature</a></div>";
            popUp.innerHTML =  html;
        }
        else{
            popUp.style.display = 'none';
        }
```

```
,
        //This method is called on DOM label creation.
        //Use this method to add event handlers and styles to
        //your node.
        onCreateLabel: function(label, node){
            label.innerHTML = node.name;
            //set label styles
            var style = label.style;
            style.color = node.getLabelData('color');

            style.textAlign= 'center';
            style.paddingTop = '3px';
            if(node.getData('drug') != 0)
            {
                  var width = node.getData('width'),
                height = node.getData('height'),
                posX = node.pos.getc(true).x,
                posY = node.pos.getc(true).y;

                  var ctx = document.getElementById("infovis-canvas").getContext("2d");
                  var gradient = ctx.createLinearGradient(posX, posY, posX+width, posY+height);
                              gradient.addColorStop(0, "#000");
                              gradient.addColorStop(1, "#fff");
                  node.setData('$gradient', gradient);
                  style.cursor = 'pointer';

,

        onPlaceLabel: function(label, node) {
            var style = label.style;
            style.width = node.getData('width') + 'px';
            style.height = node.getData('height') + 'px';
            style.color = node.getLabelData('color');
            style.fontSize = node.getLabelData('size') + 'px';
            style.textAlign= 'center';
            style.paddingTop = '3px';
            if(node.getData('drug') != 0)
            {
                  var width = node.getData('width'),
                height = node.getData('height'),
                posX = node.pos.getc(true).x,
                posY = node.pos.getc(true).y;

                  var ctx = document.getElementById("infovis-canvas").getContext("2d");
                  var gradient = ctx.createLinearGradient(posX, posY, posX+width, posY+height);
                              gradient.addColorStop(0, "#000");
                              gradient.addColorStop(1, "#fff");
                  node.setData('$gradient', gradient);
```

```
                style.cursor = 'pointer';


);
}
```