

CHALMERS



Visualizing Real-Time Data
Designing a Visual Analytics Tool for the Stock Market
Master of Science Thesis in Interaction Design

CAMILLA JOHANSSON
REBECCA NILSSON

Department of Computer Science and Engineering
CHALMERS UNIVERSITY OF TECHNOLOGY
UNIVERSITY OF GOTHENBURG
Göteborg, Sweden, April 2009

The Author grants to Chalmers University of Technology and University of Gothenburg the non-exclusive right to publish the Work electronically and in a non-commercial purpose make it accessible on the Internet. The Author warrants that he/she is the author to the Work, and warrants that the Work does not contain text, pictures or other material that violates copyright law.

The Author shall, when transferring the rights of the Work to a third party (for example a publisher or a company), acknowledge the third party about this agreement. If the Author has signed a copyright agreement with a third party regarding the Work, the Author warrants hereby that he/she has obtained any necessary permission from this third party to let Chalmers University of Technology and University of Gothenburg store the Work electronically and make it accessible on the Internet.

Visualizing Real-Time Data
Designing a Visual Analytics Tool for the Stock Market

Camilla Johansson,
Rebecca Nilsson.

© Camilla Johansson, April 2009.
© Rebecca Nilsson, April 2009.

Examiner: Staffan Björk

Department of Computer Science and Engineering
Chalmers University of Technology
SE-412 96 Göteborg
Sweden
Telephone + 46 (0)31-772 1000

Department of Computer Science and Engineering
Göteborg, Sweden April 2009

Abstrakt

Syftet med detta examensarbete var att undersöka och ta fram förslag på hur visualisering av stora mängder realtidsdata för aktiemarknaden kan te sig i ett visualiseringsverktyg som är utvecklat för statisk data.

Arbetet utfördes genom en iterativ arbetsprocess som startade med litteraturstudier och intervjuer med kapitalförvaltare i Göteborg. Arbetet riktades in på aktiemarknaden och den arbetssituation som kapitalförvaltare har.

Därefter utvecklades och utvärderades pappersprototyper och digitala prototyper i olika iterationssteg. En demonstrator utvecklad i Adobe Flash utvärderades med användare och experter innan ett koncept och designrekommendationer togs fram.

Studien visade bland annat att det var möjligt att använda samma visualiserings- och interaktionstekniker för realtidsdata som för statisk data. Det framkom även att data måste vara i realtid och att analyser måste gå snabbt och enkelt, samt att det är viktigt att kunna se utveckling över tid.

Rapporten är skriven på engelska.

Nyckelord: informationsvisualisering, realtidsdata, aktiemarknaden, finansiell data, interaktionsdesign, TIBCO Spotfire.

Abstract

The purpose of this thesis was to explore and develop proposals for how large amounts of real-time data for the stock market can be visualized, based on a visual analytics tool developed for static data.

The thesis work followed an iterative process starting with literature studies and interviews with asset managers in Göteborg. The work was targeted towards the stock market and the asset managers' work situation.

Paper prototypes and digital prototypes were then developed and evaluated in several iteration steps. A demonstrator developed in Adobe Flash was evaluated with users and experts before a concept and design recommendations were developed.

Some findings from the study were that it was possible to use the same visualization and interaction techniques for real-time data as for static data. It also showed that data must be in real-time, that analyses must be performed fast and easily and that it is important to be able to look at development over time.

Keywords: information visualization, real-time data, stock market, financial data, interaction design, TIBCO Spotfire.

Acknowledgements

This thesis work has been challenging and interesting; it has given us a great experience. We could not have completed the journey on our own and therefore would like to thank our two supervisors Maria Redström at TIBCO Software AB Spotfire Division, and Staffan Björk at the IT University of Göteborg for all the invaluable feedback and encouragement along the way.

Thanks also to the respondents in our evaluations and interviews, both at TIBCO Software AB Spotfire Division and at the banks in Göteborg, and to Mattias Bengtsson for proof reading.

Camilla Johansson & Rebecca Nilsson
Göteborg, April 2009

Table of Contents

1	INTRODUCTION	1
1.1	RESEARCH QUESTIONS AND PURPOSE	1
1.2	DELIMITATIONS	2
1.3	DISPOSITION	2
2	TIBCO SOFTWARE AB SPOTFIRE DIVISION	4
2.1	TIBCO SPOTFIRE	4
3	INFORMATION VISUALIZATION	6
3.1	VISUAL PERCEPTION	6
3.1.1	<i>Encoding data</i>	6
3.1.2	<i>Gestalt principles of visual perception</i>	7
3.2	DASHBOARD DESIGN	7
3.3	DISPLAY MEDIUMS	10
3.3.1	<i>Table</i>	10
3.3.2	<i>Line chart</i>	10
3.3.3	<i>Sparklines</i>	11
3.3.4	<i>Bar chart</i>	11
3.3.5	<i>Pie chart</i>	12
3.3.6	<i>Scatter plot</i>	12
3.3.7	<i>Treemaps</i>	13
3.4	VISUALIZING REAL-TIME DATA	14
4	THE STOCK MARKET	15
4.1	STOCK EXCHANGE	15
4.1.1	<i>Stock brokers</i>	15
4.1.2	<i>Asset managers</i>	16
4.1.3	<i>Analysts</i>	16
4.2	DATA AND INFORMATION	16
4.2.1	<i>Stock price and financial ratio</i>	17
4.2.2	<i>News</i>	17
4.2.3	<i>Indices</i>	17
4.3	INFORMATION SOURCES	17
4.3.1	<i>Websites</i>	18
4.3.2	<i>Software</i>	19
4.3.2.1	<i>Reuters</i>	19
4.3.2.2	<i>SIX</i>	20
4.4	PRESENT DISPLAY MEDIUMS USED FOR VISUALIZING STOCK MARKET DATA	21
4.4.1	<i>Open/high/low/close chart</i>	21
4.4.2	<i>Candlestick chart</i>	22
4.4.3	<i>Bar chart</i>	23
4.4.4	<i>Line chart</i>	23
4.5	UPCOMING DISPLAY MEDIUMS FOR VISUALIZING STOCK MARKET DATA	24
5	METHOD	26
5.1	THE INTERACTION DESIGN PROCESS	26
5.2	MODELS	27
5.3	TECHNIQUES AND METHODS	28
5.3.1	<i>Quantitative versus qualitative studies</i>	28
5.3.2	<i>Gathering data</i>	28
5.3.3	<i>Personal interviews</i>	29
5.3.4	<i>Observations</i>	30
5.3.4.1	<i>Usability testing</i>	30
5.3.4.2	<i>Contextual inquiry</i>	30

5.3.5	<i>Expert evaluations</i>	30
5.3.5.1	Heuristic evaluation.....	31
5.3.5.2	Cognitive walkthrough.....	31
5.3.6	<i>Personas</i>	31
5.3.7	<i>Scenarios</i>	31
5.3.8	<i>Brainstorming</i>	32
5.3.9	<i>Affinity diagram</i>	32
5.3.10	<i>Prototyping</i>	32
5.3.10.1	Low-fidelity prototypes.....	32
5.3.10.2	High-fidelity prototypes.....	33
5.4	PLANNING.....	33
5.4.1	<i>First iteration</i>	33
5.4.2	<i>Second iteration</i>	34
5.4.3	<i>Third iteration</i>	34
6	REALIZATION	35
6.1	FIRST ITERATION	35
6.1.1	<i>Pre-study</i>	35
6.1.1.1	Literature and software study.....	35
6.1.1.2	Interview in London.....	35
6.1.1.3	First interview at a Swedish bank.....	36
6.1.1.4	Meeting with internet trader.....	38
6.1.1.5	Second interview at a Swedish bank.....	38
6.1.2	<i>Requirement analysis</i>	39
6.1.2.1	Requirement specification.....	39
6.1.2.2	Scenarios.....	40
6.1.2.3	Evaluation of requirements and scenarios.....	40
6.1.3	<i>Design of the layout and plot interaction</i>	41
6.1.3.1	Low-fidelity prototyping.....	41
6.1.3.2	High-fidelity prototyping.....	44
6.1.4	<i>Evaluation</i>	45
6.2	SECOND ITERATION	46
6.2.1	<i>(Re)design of the layout, work flow and functions</i>	46
6.2.1.1	Low-fidelity prototyping.....	46
6.2.1.2	High-fidelity prototyping.....	47
6.2.2	<i>Expert evaluation</i>	48
6.3	THIRD ITERATION	48
6.3.1	<i>(Re)design of final prototype</i>	49
6.3.1.1	Low-fidelity prototyping.....	49
6.3.1.2	High-fidelity prototyping.....	51
6.3.2	<i>User test and expert evaluation</i>	51
6.4	FOURTH ITERATION	53
6.4.1	<i>(Re)design of concept</i>	53
6.4.1.1	Low-fidelity prototyping.....	53
7	RESULT	55
7.1	DEMONSTRATOR.....	55
7.1.1	<i>Layout</i>	55
7.1.1.1	Market view.....	56
7.1.1.2	Portfolio view.....	57
7.1.1.3	Stock view.....	58
7.1.2	<i>Display mediums</i>	58
7.1.2.1	Treemaps.....	59
7.1.2.2	Line chart.....	60
7.1.2.3	Scatter plot.....	61
7.1.2.4	Bar chart.....	62
7.1.2.5	Events.....	63
7.1.2.6	Best/Worst.....	63
7.1.3	<i>Features</i>	64
7.1.3.1	Alert.....	64
7.1.3.2	Time controls.....	65
7.1.3.3	Duplicate.....	67
7.1.3.4	Minimum/Maximum.....	67

7.2	CONCEPT	69
7.2.1	Layout	69
7.2.1.1	Market overview.....	73
7.2.1.2	Portfolio overview.....	74
7.2.1.3	Portfolio view.....	75
7.2.1.4	Stock view.....	76
7.2.2	Graphic design.....	77
7.2.3	Navigation and interaction	78
7.2.4	Display mediums	80
7.2.4.1	Treemaps	80
7.2.4.2	Line chart.....	84
7.2.4.3	Scatter plot.....	86
7.2.4.4	Bar chart	87
7.2.4.5	Sparklines	88
7.2.4.6	Events	88
7.2.4.7	Best/Worst.....	90
7.2.5	Features	91
7.2.5.1	Alert.....	91
7.2.5.2	Time controls.....	93
7.2.5.3	Duplicate	95
7.2.5.4	Minimum/ Maximum	95
7.2.5.5	Content control.....	96
7.2.5.6	Miniatures.....	97
7.2.6	Details-on-demand and filter panel	97
8	DISCUSSION.....	98
8.1	REALIZATION.....	98
8.2	RESULT.....	101
8.3	DESIGN RECOMMENDATIONS.....	104
8.3.1	General design recommendations for visualizing real-time data	105
8.3.2	Design recommendations specific for the domain and user group.....	106
8.4	FUTURE WORK	107
9	CONCLUSIONS.....	108

REFERENCES

APPENDIX 1. INTERACTION TECHNIQUES IN TIBCO SPOTFIRE
APPENDIX 2. INTERVIEW GUIDE
APPENDIX 3. PERSONA
APPENDIX 4. REQUIREMENT SPECIFICATION
APPENDIX 5. SCENARIOS
APPENDIX 6. PRIORITIZING LIST
APPENDIX 7. TEST SCENARIO

Table of Figures

FIGURE 1. TIBCO SPOTFIRE'S INTERFACE.	5
FIGURE 2. TABLE.	10
FIGURE 3. LINE CHART.	11
FIGURE 4. SPARKLINES (TUFTE, 2006).	11
FIGURE 5. BAR CHART.	12
FIGURE 6. PIE CHART.	12
FIGURE 7. SCATTER PLOT.	13
FIGURE 8. TREEMAP (BEDERSON ET AL., 2002).	14
FIGURE 9. GOOGLE FINANCE'S WEBSITE [3].	18
FIGURE 10. REUTERS FIRST [5].	20
FIGURE 11. SIX TRADER [6].	21
FIGURE 12. OPEN/HIGH/LOW/CLOSE CHART [7].	22
FIGURE 13. OPEN/HIGH/LOW/CLOSE BAR.	22
FIGURE 14. CANDLESTICK CHART [7].	22
FIGURE 15. CANDLESTICK BAR.	23
FIGURE 16. BAR CHART [7].	23
FIGURE 17. LINE CHART [7].	23
FIGURE 18. PANOPTICON'S TREEMAP [8].	24
FIGURE 19. THE HIVE GROUP'S TREEMAP [9].	24
FIGURE 20. FUSIONGADGETS DASHBOARD CONTAINING SPARKLINES [10].	25
FIGURE 21. MACKAY AND FAYARD'S FRAMEWORK FOR RESEARCH IN INTERACTION DESIGN (1997).	26
FIGURE 22. AN ITERATIVE LIFECYCLE MODEL (ADAPTED FROM PREECE ET AL., 2007).	27
FIGURE 23. TIME PLAN.	34
FIGURE 24. PERFORMING THE METHOD AFFINITY DIAGRAM.	39
FIGURE 25. THE FIRST PAPER PROTOTYPES OF THE LAYOUT.	41
FIGURE 26. A SKETCH MADE IN PHOTOSHOP OF THE LAYOUT IN THE STOCK VIEW.	42
FIGURE 27. A SCATTER PLOT DISPLAYING HIERARCHICAL DATA.	43
FIGURE 28. OUR ALTERNATIVE DESIGN DISPLAYING HIERARCHICAL DATA.	43
FIGURE 29. A TREEMAP DISPLAYING HIERARCHICAL DATA.	44
FIGURE 30. AN EARLY FLASH ANIMATION OF A SCATTER PLOT IN REAL-TIME.	45
FIGURE 31. AN EARLY FLASH ANIMATION OF A TREEMAP.	45
FIGURE 32. THE SECOND PAPER PROTOTYPES OF THE LAYOUT.	46
FIGURE 33. THE MARKET VIEW FROM THE FIRST FLASH PROTOTYPE.	47
FIGURE 34. PAPER PROTOTYPE OF THE FINAL LAYOUT OF THE MARKET VIEW IN THE PROTOTYPE.	49
FIGURE 35. PAPER PROTOTYPE OF THE FINAL LAYOUT OF THE PORTFOLIO VIEW IN THE PROTOTYPE.	50
FIGURE 36. PAPER PROTOTYPE OF THE FINAL LAYOUT OF THE STOCK VIEW IN THE PROTOTYPE.	50
FIGURE 37. PAPER PROTOTYPE OF THE DESIGN OF THE TOOLBAR IN THE CONCEPT.	54
FIGURE 38. THE MARKET VIEW IN THE DEMONSTRATOR.	56
FIGURE 39. THE PORTFOLIO VIEW IN THE DEMONSTRATOR.	57
FIGURE 40. THE STOCK VIEW IN THE DEMONSTRATOR.	58
FIGURE 41. THE TREEMAP IN THE DEMONSTRATOR.	60
FIGURE 42. A DRILL DOWNED TREEMAP IN THE DEMONSTRATOR.	60
FIGURE 43. THE LINE CHART IN THE DEMONSTRATOR.	61
FIGURE 44. ADD SYMBOL IN THE DEMONSTRATOR.	61
FIGURE 45. THE SCATTER PLOT IN THE DEMONSTRATOR.	62
FIGURE 46. THE BAR CHART IN THE DEMONSTRATOR.	62
FIGURE 47. EVENTS IN THE DEMONSTRATOR.	63
FIGURE 48. THE BEST/WORST TABLE.	64
FIGURE 49. THE ALERT AREA IN THE DEMONSTRATOR.	64
FIGURE 50. THE DIALOGUE BOX FOR THE ALERT.	64
FIGURE 51. A TRIGGERED ALERT IN THE DEMONSTRATOR.	65
FIGURE 52. THE TIME CONTROLS IN THE DEMONSTRATOR.	66
FIGURE 53. A REWINDED LINE CHART IN THE DEMONSTRATOR.	66
FIGURE 54. BEFORE A DUPLICATE IS MADE.	67
FIGURE 55. AFTER A DUPLICATE HAS BEEN MADE.	67
FIGURE 56. BEFORE DISPLAYING MINIMUM/MAXIMUM VALUES.	68
FIGURE 57. THE MINIMUM/MAXIMUM DIALOGUE BOX.	68

FIGURE 58. MINIMUM/MAXIMUM VALUES DISPLAYED ON A TREEMAP.	69
FIGURE 59. THE LAYOUT IN THE CONCEPT.	71
FIGURE 60. THE MARKET OVERVIEW IN THE CONCEPT.	74
FIGURE 61. THE PORTFOLIO OVERVIEW IN THE CONCEPT.	75
FIGURE 62. THE PORTFOLIO VIEW IN THE CONCEPT.	76
FIGURE 63. THE STOCK VIEW IN THE CONCEPT.	77
FIGURE 64. TITLE BAR.	78
FIGURE 65. THE TOOLBAR.	78
FIGURE 66. TOOLTIP DISPLAYED IN A TREEMAP.	79
FIGURE 67. THE TREEMAP IN THE CONCEPT.	80
FIGURE 68. A DRILL DOWNED TREEMAP IN THE CONCEPT.	82
FIGURE 69. A REAL-TIME UPDATE IN THE TREEMAP.	84
FIGURE 70. THE LINE CHART IN THE CONCEPT.	84
FIGURE 71. ADD SYMBOL IN THE CONCEPT.	85
FIGURE 72. CURSORS FOR THE LINE CHART.	86
FIGURE 73. SPARKLINES DISPLAYED IN A SCATTER PLOT.	88
FIGURE 74. EVENTS IN THE CONCEPT.	90
FIGURE 75. A TRIGGERED ALERT IN THE CONCEPT.	92
FIGURE 76. THE TIME CONTROLS IN THE CONCEPT.	93
FIGURE 77. A REWINDED LINE CHART IN THE CONCEPT.	94
FIGURE 78. CONTROL FOR CHANGING THE CONTENT IN THE VIEW.	96

1 Introduction

The first chapter gives an introduction to this thesis and presents the research questions and purpose of this study. Thereafter delimitations are described together with a review of the disposition of this report.

With the help of computers it has become possible to easily access and handle very large amounts of data and information on a daily basis. As much as this is a great advantage, it can also be problematic since the constant flow of information can be overwhelming and difficult to interpret. Many areas dealing with large quantities of information, such as the stock market, banking, and logistics, are also dependent on constantly updated information, and often it is also important to find quick answers in the large amounts of data. This is especially true in the stock market, where making the right decisions based on a constant flow of information is crucial and must be made instantly; making wrong decisions can be economically devastating.

Information visualization is an aid for cognition and has become increasingly important along with the rapid growth of information available. Visual representations help us think and by adding interactivity to the visual representations it is possible to gain an even bigger insight into the data by rearranging and manipulating the data to find patterns and trends. This is also the idea behind the product TIBCO Spotfire, which is an interactive tool that visualizes large quantities of data and lets the user access, analyze and ask questions to the data. The user can find patterns, trends, etc in business data and be helped in making decisions based on the findings in the data. This tool is used for static data today, and visualization of real-time data is still relatively unexplored.

The actors on the stock market have much to gain from making faster and better decisions based on their data. Most data on the stock market today is displayed using tables. Little work has been done on exploring which visualizations that are suitable to represent the large amounts of real-time data handled in the stock market (Merino, Sips, Keim, Panse & Spence, 2006). This is therefore an area with great potential to benefit from a tool that can provide the actors with visualizations of real-time data.

In this thesis work we set out to explore the possibilities for visualizing and interacting with large amounts of real-time data for the stock market with the aid of a visual analytics tool similar to TIBCO Spotfire.

1.1 Research questions and purpose

This thesis is based on an assignment suggested by TIBCO Software AB Spotfire Division. The purpose of this study is to examine a relatively unexplored area; visualization of, and interaction with, real-time data, and to propose design recommendations for visualizing real-time data.

The primary research question of this thesis can be formulated as follows:

How can TIBCO Spotfire, a visual analytics tool designed for static data, be modified to provide real-time data for the stock market?

To be able to answer the primary research question of this thesis the potential user group within the stock market must be identified. This is important since the modification of the visual analytic software tool is very dependent on who the users are. The similarities and differences between visualizing and interacting with static versus real-time data also need to be explored to be able to design a well-functioning real-time visual analytics tool. Therefore the following two research questions must also be answered:

Who are the potential users of a real-time visual analytics tool within the stock market?

What are the similarities and differences between visualizing and interacting with static versus real-time data?

1.2 Delimitations

To keep this thesis work within a reasonable scope some delimitations were necessary. The financial market was the initial area of interest, but since this is an enormous field consisting of many very different sub-fields where the work situations differ greatly, the thesis work will focus on the stock market which is a part of the financial market.

Additionally, we have based our study on the Swedish stock market and more specifically on interviews with users in Göteborg. We are aware that the Swedish stock market is much smaller than many markets abroad, such as NASDAQ and the London Stock Exchange, but since the users we have had access to are all located in Göteborg; we have focused on their situation.

1.3 Disposition

An overview of the structure of the thesis is given here. The thesis consists of nine chapters, which are described below.

Chapter 1 - Introduction

An introduction to the thesis and a presentation the research questions for the study along with the purpose of the study. Delimitations of the thesis work are described together with a review of the disposition of the report. The chapter should give the reader an understanding of what areas the thesis will touch upon.

Chapter 2 - TIBCO Software AB Spotfire Division

The company whom the thesis was conducted in cooperation with is described in the second chapter. A description of the company's product, TIBCO Spotfire, is also given. This chapter is of interest for the reader who wants to learn more about the company and the product we have had as a starting point in our work.

Chapter 3 - Information Visualization

A general description of information visualization is given in this chapter along with more specific theories related to information visualization and relevant for the work. At the end of the chapter visualization of real-time data is processed. The chapter is aimed at readers who want to gain some knowledge and an overview of the area of information visualization.

Chapter 4 - The Stock Market

The domain for this thesis, the stock market, is described in this chapter. The description includes the actors on the stock market, the data and information available, the information sources and the visualization techniques used on the stock market today. This chapter is intended for readers who want to gain knowledge about the stock market.

Chapter 5 - Method

Describes how interaction design can be seen as a research field and different models describing the work process within this field. Various methods and techniques commonly used in interaction design which are relevant to the thesis work are also described. The chapter is of interest for the reader who wants to gain knowledge about the models, methods, and techniques commonly used within interaction design.

Chapter 6 - Realization

Describes our process throughout the thesis work, which has consisted of four different iterations. Each iteration has contained several different stages, such as gathering needs and requirements, requirement analysis, and design and evaluation. This chapter is aimed at readers who are interested in how we conducted the thesis work, which models, methods, and techniques that were used.

Chapter 7 - Result

The results of our work are described here. The results consist of a prototype developed according to the gathered requirements. The whole design concept which we have developed throughout the work is also described. The chapter is of interest for the reader who wants to know what our work resulted in.

Chapter 8 - Discussion

The thesis work is discussed from two viewpoints: the realization and the results. Design recommendations for visualization of real-time data in general and for the specific domain and user group are presented. The chapter is concluded with discussing possible future work. This chapter is intended to provide the reader with our view and thoughts of the thesis work.

Chapter 9 - Conclusion

The final conclusions about visualization and interaction with real-time data for the stock market are presented and the research questions for this thesis are answered in this chapter. The chapter is to give the reader an understanding of what the thesis work resulted in and what conclusions can be drawn from this.

2 TIBCO Software AB Spotfire Division

In this chapter the company whom this thesis was conducted in cooperation with is described. A description of this company's foremost product is also given.

TIBCO is a US company, which provides enterprise software that ties together systems and processes so that information can be processed and presented in real-time. Since 2007 Spotfire is a division within TIBCO, and is now known as TIBCO Software AB Spotfire Division. TIBCO Software AB Spotfire Division will further on in this thesis be referred to as the Company. The Company delivers interactive visual analytics tools. They present data visually and let the user filter the data instantly with dynamic query devices (Ahlberg, Williamson & Shneiderman 1992). The Company's software consequently help people who make business decisions to ask and answer unconstrained questions. The Company's interactive, visual software lets users discover trends, patterns, and relationships in data with speed and flexibility. The foremost business analytics software from the Company is called TIBCO Spotfire.

2.1 TIBCO Spotfire

TIBCO Spotfire is an interactive tool for accessing, analyzing and creating dynamic reports (TIBCO Software Inc., 2008). The tool helps you reveal and identify hidden patterns, trends, outliers, and unanticipated relationships in your business data. This can help you in your decision-making processes. You can access and import data into TIBCO Spotfire from a database, a text file or a spreadsheet. The imported data in TIBCO Spotfire is static. Visualizations of the data can be created in numerous forms of charts. The data that is imported into the software can be filtered interactively, which leads to a better understanding of the data. You can also share your results with your colleagues and customers by creating dynamic reports in TIBCO Spotfire.

TIBCO Spotfire's interface mainly consists of three different parts: the visualizations (1), the filter panel (2), and the details-on-demand window (3) (see Figure 1). Numerous different types of *visualizations* can be displayed simultaneously on one page. You can choose from 10 different types of visualizations to display: bar charts, tables, cross tables, line charts, parallel coordinate plots, pie charts, scatter plots, map charts, box plots, and summary tables. You can resize and move visualizations as well as arranging them evenly, side-by-side, stacked or maximize the active visualization. By using the axis selectors you can change what is to be shown on each axis in the visualization. The software also offers different statistical methods, which can be applied to the data, shown in the visualizations, and facilitates analysis. For instance average, median, standard deviation, percentiles, quartiles, outliers, adjacent values, variance, confidence intervals, summation, count, minimum and maximum values are some of the methods available in TIBCO Spotfire. You can get more detailed information about an item in the visualization through the tooltips that are displayed when hovering over the items or by selecting an item and read the displayed information in the details-on-demand window.

The *details-on-demand* window is used to display the details that are associated with the marked items in the visualization. The details-on-demand window displays numerical values and textual data for one or several marked items. Here you can receive more elaborate information about your data.

Filtering is used to select what you want to look at in the data set. By adjusting filters, you reduce the data seen in the visualizations and "drill down" on the things that interest you. Filters are powerful tools that quickly let you see various aspects of your data and make new discoveries. Filters appear in several forms, and you can select the type of filter device that best suit your needs. You can choose between range, item, hierarchy and text filters, radio buttons, and check boxes. When you manipulate a filter by moving a slider or by selecting a check box, all visualizations are immediately updated to reflect the new selection of data.

You can interact with the data in TIBCO Spotfire in several different ways. Different combinations of keyboard and mouse *interaction* can be used to mark items and manipulate the data. See Appendix 1 for a description of the different keyboard and mouse interactions that can be found in TIBCO Spotfire.

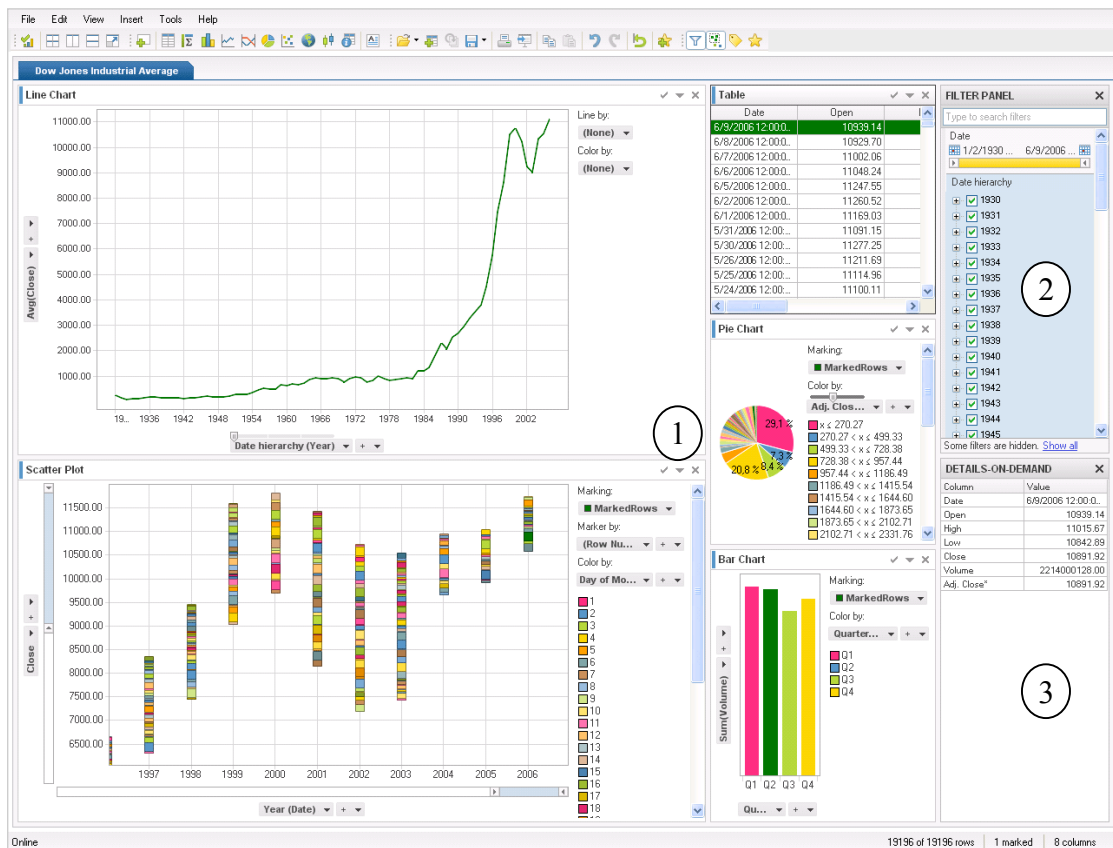


Figure 1. TIBCO Spotfire's interface.

3 Information Visualization

In this chapter a general description of information visualization is given. More specific theories related to information visualization and relevant for this work, such as gestalt principles of visual perception, dashboard design and different display mediums for visualizing data, are also described. At the end of the chapter visualization of real-time data is processed.

Computers offer a great help when it comes to providing us with data. But large amounts of data can be overwhelming, and this is where information visualization plays an important role. According to Card et al. (1999, p. 7) information visualization is “*the use of computer-supported, interactive, visual representations of abstract data to amplify cognition*”. Information visualization is an extensive research field, much too large to cover in detail within the scope of this thesis. For a deeper review of information visualization, see for example Card, Mackinlay and Shneiderman (1999), Spence (2007) and Tufte (1983; 2006).

Visual representations have served as aids for thinking for a very long time and are a great help when it comes to making sense of data. We often call understanding something to “see” it (Card et al., 1999). Spence (2007) recognizes three important issues in information visualization, namely representation, presentation and interaction. Representation is the way values are encoded visually. The presentation of data is the selection and layout of the encoded data in the available display area and available time. Interaction is used to be able to manipulate and rearrange the data to find patterns and gain insight into the information. Spence also recognizes the user’s visual perception and interpretation of the data as important when visualizing data, which will be further discussed in the next section.

3.1 Visual perception

Few (2006) and Spence (2007) talk about the importance of understanding visual perception when visualizing information. Here pre-attentive processing becomes important; which is to unconsciously perceive visual attributes at an extraordinary speed. To encode data to facilitate a rapid perception through pre-attentive processing and to have knowledge of how we perceive patterns and form, i.e. the gestalt principles of visual perception, is important within information visualization. If we understand the different rules for visual perception we can according to Ware (2004, p. xxi) “*present data in such a way that the important and informative patterns stand out.*”

3.1.1 Encoding data

Encoding data is all about making it easier for the user to interpret and perceive the data. Data can for example be encoded so that a rapid perception through pre-attentive processing is facilitated. The pre-attentive attributes of visual perception are grouped into four categories by Few (2006) and Ware (2004): *color, form, spatial position, and motion*.

The use of different *colors* or different brightness and intensity in colors can emphasize certain data. When it comes to *form*, size and shapes can be useful. If you would like to rank the importance of the information size can be used. Shapes can be used to differentiate data from each other in e.g. a graph. *Spatial position* handles the objects' location relative to horizontal and vertical dimensions. The primary attribute for *motion* is flickering. Flickering is a powerful attention-getter since we easily perceive something that suddenly appears within our field of vision.

The attributes stated above can be used to facilitate a more rapid perception; however Few (2006) states that we should be aware that there are limits to these attributes. There is a limit to the number of distinct expressions of one pre-attentive attribute we can distinguish at a time; not more than five distinct expressions should be applied for one attribute.

3.1.2 Gestalt principles of visual perception

The gestalt principles deals with how we perceive pattern, form, and how we organize what we see. Six gestalt principles explain the visual characteristics that incline us to group objects together: *proximity*, *similarity*, *enclosure*, *closure*, *continuity*, and *connection* (Few, 2006).

The principal of *proximity* explain that objects that are located near one another, are perceived to be grouped together. This principle can be used to link information which is related. The principle of *similarity* means that we tend to group together objects that are similar to one another, in for example color, size, shape, or orientation. The *enclosure* principle points out that we group objects together which are enclosed by any form of visual border. The principle of *closure* states that we rather perceive objects as closed, complete, and regular, than open, incomplete, and unusual. The *continuity* principle explains that we identify objects as belonging to each other, if they are aligned and appear to be a form of continuation of one another. The principle of *connection* states that objects that are connected to each other appear to belong together (Few, 2006; Ware, 2004).

3.2 Dashboard design

One tool which has emerged within business intelligence in recent years is the information dashboard. The dashboard is primarily about communicating critical information to the user and consists of a visual display. The information is most often displayed in real-time. Few (2006, p. 34) defines the dashboard as follows:

A dashboard is a visual display of the most important information needed to achieve one or more objectives; consolidated and arranged on a single screen so the information can be monitored in a glance.

There is an extensive variety in what dashboards look like, but according to Few (2006) they do have some things in common. A dashboard presents information visually, with the help of both graphics and text. Graphics is dominating in a dashboard, since it can communicate information in a more richer and more efficient way than text alone. The dashboard is also to provide an overview of the most critical information fitted onto one screen, so that the user can monitor the information in a glance. The information most often consists of abbreviated data in the form of

summaries and expectations. Even if emphasis is on displaying an overview of the data, the user should be able to drill down on the information and see details. The dashboard shall quickly be able to indicate and communicate to the user what information that needs your attention. Dashboards are a tool which can help people to identify trends, patterns, and abnormalities. The tool can through analysis also supply reasons for what is displayed and guidance on what to do next.

The dashboard poses a new information visualization challenge, which is to produce a simply and instantly understandable display while fitting large amounts of data into a small area of space. For this challenge Few (2006) mentions a number of things to think about when designing dashboards.

The data should fit in one single screen. No scrolling should be needed, since it can result in a loss of critical information when you loose sight of data. You can only hold between three and nine chunks of information in your short-term memory at once. Information that can display meaningful relationship must be displayed together. To access additional and more detailed information in excess of the overview provided by the dashboard, separate screens can however be used.

Contextual information must be displayed for the data. If no context is supplied for the data, you have nothing to compare and relate the data to. You cannot know if the figures are good or bad. Here clear labeling and visual encoding becomes important.

Avoid excessive details and precision. The dashboard is to display high-level information to give the user a quick overview. Details and precision in this case only result in slowing down the user, since you then need to filter out the most important information. One of Tufte's (1983) grand principles is to ensure the quality, relevance, and integrity of the data, by only displaying the information the user really needs to reach a certain goal. Tooltips can be used to provide more detailed information which does not need to be visible to the user at all times (Cooper & Reimann, 2003).

Use measures which directly express the intended message. Visualizations should communicate the meaning to the user as clearly and efficiently as possible. To do this, a measure which most directly supports the user's needs must be chosen. If for example variance between variables should be displayed it is most efficient and clear to do this by using percentage, instead of displaying the actual amounts and force the user to calculate the difference themselves.

Choose an appropriate display medium. The most common mistake in all forms of quantitative data presentation is to choose an inappropriate display medium, for example using a pie chart when a table would be better. The display media chosen should also be able to serve its purpose when sized to fit into a small area.

Only use a variety of display medias if it is meaningful. You should always select the display medium that is best suited for what you want to communicate, even if this means that your dashboard only will contain the same kind of displays. The use of the same display medium actually provides consistency for the user, since the same perceptual strategy can be used to interpret the data. This both saves time and effort for the user.

Design the display medias to communicate clearly and efficiently. This involves several different things depending on which medium you are to design, but some general examples can be given. For example you should use labels directly in the graph so the user does not have to jump back and forth between the graph and a legend. This is something Tufte (1983) also mentions in one of his seven grand principles of information visualization. When using colors, you should also consider that bright colors stand out and that the use of too alike colors makes it hard to distinguish them from each other.

Encode the quantitative data accurately. If graphical representations are designed poorly, this can lead to inaccurate conclusions. For example in a bar chart comparisons often are made of the bars' length and if the whole scale is not shown, these comparisons will be misleading. The value in a bar chart should therefore always start at 0.

Organize and arrange the data to facilitate understanding. The most important data and data that need immediate attention should be prominent. Data that is related should be grouped together. Both Cooper and Reimann (2003) and Few (2006) mention that in western countries the user's eye scans the screen from left to right and top to bottom. This means that, for users from these countries, the top left corner of the screen is the most prominent and that the lower right corner is the least prominent. This must be considered when arranging the data in the dashboard.

Highlight important data effectively. The most important information in the dashboard should immediately stand out when you look at it. This can be done by using the pre-attentive attributes of visual perception described earlier in this chapter.

Avoid the use of abundant decoration. Useless decoration act as distraction and is often a waste of space, which already is limited in a dashboard. When designing graphical attributes for dashboards, such as icons, it is therefore important to reduce the number of colors, avoid excessive visual detail and maintain a modest size of the graphical attributes. When designing icons, metaphors can be used since they are understood intuitively and are useful for representing the purpose of for example a button in a toolbar (Cooper & Reimann, 2003).

Avoid misuse or overuse of color. Colors should only be used when they add something to the visualization, i.e. thoughtfully and not randomly. We should be aware of that we pay attention to color which appear as contrasts, relate the same colors to each other even if used in different places, and that we exclude 10% of the male population and 1% of the female population when using colors such as red, yellow and green as the only mean of interpretation. It is therefore important to provide a complementary way for users with color impairment to interpret important information (Cooper & Reimann, 2003).

Design attractive visual displays. Since a dashboard is to supply users with important information, it shall also be pleasant to look at. Most often simplicity is the way to go according to Few (2006).

Few (2006) means that well-designed dashboards are exceptionally well organized, specific and customized to users of the dashboard, and use concise and small media to

communicate to be as clear and direct as possible. The use of direct feedback and direct manipulation in dashboard design makes it easier for the user to understand and interpret the data in the design, which is very important (Cooper & Reimann, 2003).

The information found on a dashboard is characterized by consisting of summaries and exceptions. The most common summaries used are average and sums. Measures of correlation and distribution can also be of interest. A dashboard often displays the most critical values know as the exceptions. These values fall out of the realm of normality and require the user’s attention. Different display mediums are used to display the information in the dashboard.

3.3 Display mediums

Descriptions of seven kinds of display mediums used for visualizing data and relevant for this work are given below. These are: *tables*, *line charts*, *sparklines*, *bar charts*, *pie charts*, *scatter plots* and *treemaps*. They include advantages and disadvantages of each kind of medium. The mediums mentioned are some of the most commonly used today along with mediums relevant for the domain of this thesis, the stock market.

3.3.1 Table

Tables arrange data by putting it into columns and rows, see figure 2. This visualization technique is often used to structure text. By providing a structure to the information, the content can be communicated clearly. Tables are good at providing more detailed information about exact individual values. They facilitate comparison and further analysis of these individual values. However, they are not well-suited for giving an overview of the data and it is also difficult to discover patterns over time using a table (Few, 2006; Wallgren, Wallgren, Persson, Jorner & Haaland, 1996).

Date	Open	High	Low
6/9/2006 12:00:0...	10939.14	11015.67	10842.89
6/8/2006 12:00:0...	10929.70	11032.15	10726.15
6/7/2006 12:00:0...	11002.06	11107.48	10897.76
6/6/2006 12:00:0...	11048.24	11140.45	10890.24
6/5/2006 12:00:0...	11247.55	11259.96	11025.75
6/2/2006 12:00:0...	11260.52	11329.28	11158.06

Figure 2. Table.

3.3.2 Line chart

A line chart is ideal for showing development over time. It provides a good overview and reveals the shape of the data (see Figure 3). It is easy to recognize overall patterns, such as trends, fluctuations, cycles, and rates of change, in this type of visualization. The line chart can also easily show several different data sets in one visualization, and thereby facilitate comparison (Few, 2006; Wallgren et al., 1996).

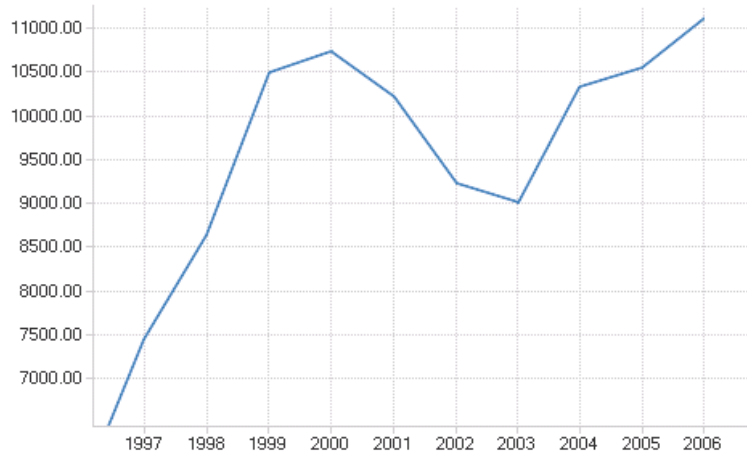


Figure 3. Line chart.

3.3.3 Sparklines

Sparklines display time series data as does the line chart, but it is simpler and more space-efficient. It does not contain any quantitative information and thereby lack in detail, but it does provide a good and quick overview of historical trends in the data. It is ideal to use sparklines in designs where you need to track and be able to compare development over time on a small area of space. This visualization technique can also be embedded in text or tables, see figure 4. By using sparklines in tables you can extend the attention span of the table from concentrating on displaying current values to include a more widespread span in time (Few, 2006; Tufte, 2006).



Figure 4. Sparklines (Tufte, 2006).

3.3.4 Bar chart

A bar chart is often used to show frequencies, sums and averages, since it is well-suited for summarizing categorical data. A bar chart can provide a good means of comparisons for these kinds of data and also for adjacent values. There are different kinds of bar charts, such as vertical, horizontal, grouped, overlapping, and stacked bar charts. Figure 5 displays a vertical bar chart. The bar chart best displays individual values and provide a good comparison between these kinds of values. The bar chart is also most suitable to show a small number of individual values in time. If you are to display a continuous development over time line charts are more appropriate (Few, 2006; Wallgren et al., 1996).

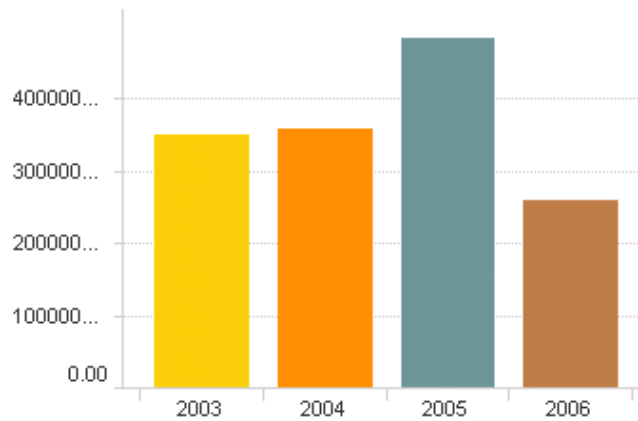


Figure 5. Bar chart.

3.3.5 Pie chart

A pie chart is an alternative to a bar chart when you are to show the relative sizes of the parts of a whole, such as percentage distribution (see Figure 6). It can however be difficult to perceive the exact measurements of a circle, and therefore a pie chart is most suitable for showing overview (Wallgren et al., 1996).

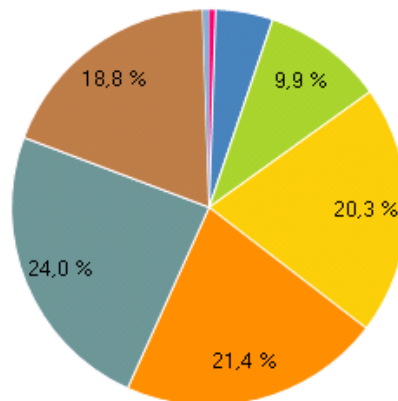


Figure 6. Pie chart.

3.3.6 Scatter plot

A scatter plot is good at showing relationship between two quantitative variables, see figure 7. Other additional variables can also be added to the visualization with the help of for example color, size, or shape. The scatter plot can have some problems with displaying large amounts of data, since the visualization then becomes very crowded and the dots may overlap. In these cases it can be good to show aggregated data, i.e. to categorize the data, in the scatter plot. The scatter plot can also show how the relationships have changed over time, by using animation and letting the values move to represent change through time. If you are to compare values at different points in time it is necessary that you can see the full path of the values from one point in time to another in the scatter plot. By providing this feature the scatter plot can for example show the magnitude, direction, shape, and velocity of change for the values over time (Few, 2007; Wallgren et al. 1996).

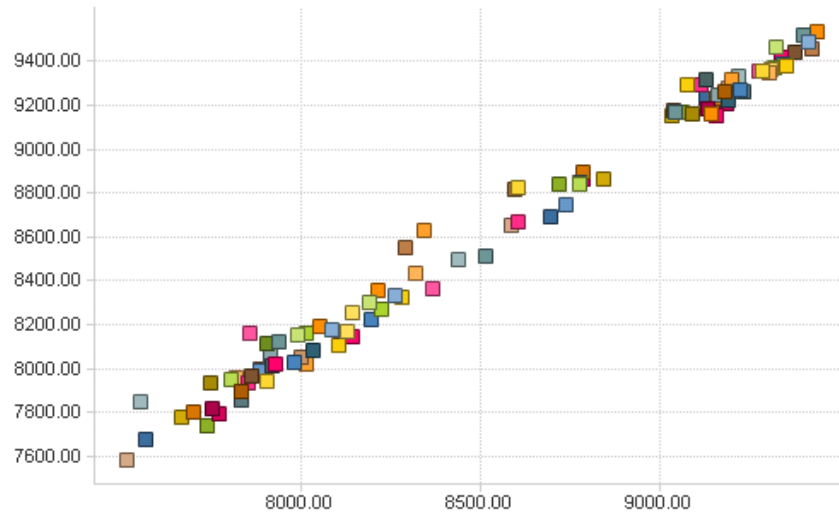


Figure 7. Scatter plot.

3.3.7 Treemaps

Treemaps is a visualization technique suited for presenting hierarchically or categorically structured information in a very space-efficient manner. All of the available space is partitioned into rectangular boxes which have been sized to encode a quantitative variable. These boxes create the structure of the treemap. Treemaps can have different layouts depending on which way you want to order your treemap. In Figure 8 an example of a squarified treemap can be seen, which is ordered by the size of the rectangles. The most important information is often given the most space in the treemap by letting the size of the boxes represent importance (Bederson, Shneiderman & Wattenberg, 2002). Treemaps use enclosure to present their hierarchical or categorical structure. This means that hierarchies and categories are represented in the treemap as rectangles contained by other larger rectangles. By using enclosure treemaps make efficient use of all the available space (Björk, 2000). Color is also often used in the treemap to add another quantitative variable to the visualization. The treemap has its strengths when it comes to spotting conditions that can be of interest. These are often the extremes, i.e. for example the smallest/largest or the most colorful/colorless rectangle in the treemap. Treemaps also often provide a possibility for the user to select a particular hierarchy or category of the treemap and drill down on this area to display more detailed information and enable analysis of lower levels of data (Few, 2006).

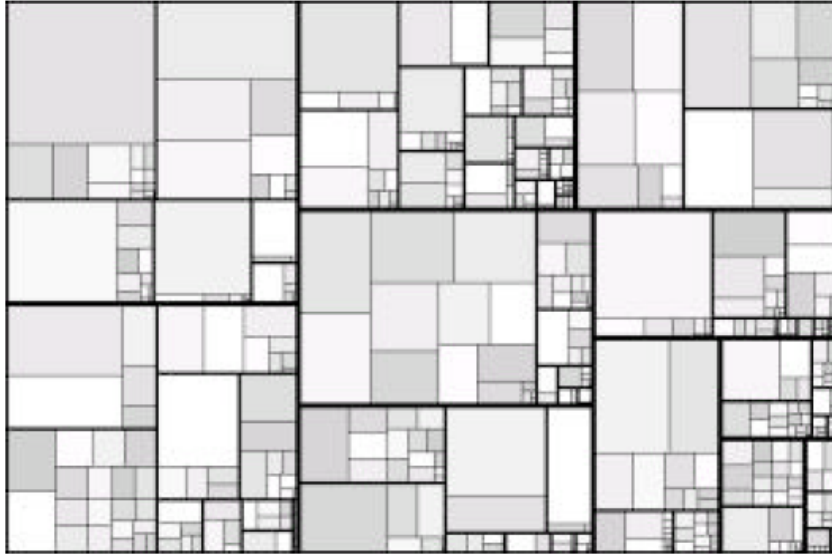


Figure 8. Treemap (Bederson et al., 2002).

3.4 Visualizing real-time data

Through technical developments such as the Internet it has become possible to make information available as soon as it is obtained, e.g. to present stock prices for the different actors on the stock market. The actors on the stock market are dependent on continuous information and the information needs to be updated as fast as possible, i.e. in real-time. Real-time data on the stock market is defined to be in the same moment as the data is changed, not a second later (Swedbank Markets, 2003).

While looking at the area of visualization of real-time data we have discovered that it is a fairly unexplored area. Little research has been done concerning visualizations of dynamic information sources and streaming data. Panopticon (2007, p. 31) believes the reason for this to be that “*visualizing real-time data is often prohibitively computationally intensive*” and that “*it is more difficult to handle a live stream instead of a static or snapshot dataset*”. Merino et al. (2006) confirms a lack of studies when it comes to exploring which visualizations that are suitable to represent the large amounts of real-time data handled in the stock market.

4 The Stock Market

The domain for this thesis, the stock market, is described in this chapter. The description includes the trade and the actors on the stock market, the data and information available, the information sources and the visualization techniques used on the stock market today.

The stock market offers an investment alternative for your capital and is the largest of three actors on the capital market. Its main purpose is to *function as a meeting point for stock trade*. A stock represents an ownership in the joint stock company. This means that investors, which place their capital in different stocks, get a shared ownership in a company along with a shared financial responsibility. A stockholder can both get an income and a dividend from the capital invested in the company (Sveriges Riksbank, 2007). This profit is dependent on how well the company and market is doing, and can vary heavily. The stock market is therefore an unsure market, where you as a stockholder can increase as well as decrease your capital both rapidly and greatly. The stock market is thus sometimes called the venture capital market (Andersson, 2006).

4.1 Stock exchange

Stocks get listed and traded on different stock exchanges, which are *corporations or organizations specialized in bringing buyers and sellers of securities together*. The world's leading stock exchanges are among others the *New York Stock Exchange*, *Tokyo Stock Exchange*, *NASDAQ*, and the *London Stock Exchange* [1]. *Stockholm stock exchange* is a part of the Nordic Exchange, OMX, which consists of one Nordic and one Baltic list. The Baltic list consists of companies listed on the exchanges in Tallinn, Riga, and Vilnius. The Nordic list comprises the companies listed on the exchanges in Stockholm, Copenhagen, Reykjavik, and Helsinki (Sveriges Riksbank, 2007). The different exchanges are also divided into lists based on the companies' size. These lists are called *Large Cap*, *Mid Cap*, and *Small Cap*. The Swedish Large Cap list comprises approximately 90 companies with an exchange value of over one billion Euro. The Swedish Mid cap list also contains around 90 companies but their exchange value lies between 150 million and one billion Euro. Roughly 130 companies are listed on the Swedish Small Cap list. These companies have an exchange value of less than 150 million Euro.

You can trade with stocks mainly in two different ways; Internet trading or through a security institute, such as a bank or an investment bank. Since internet trading is not of interest for this thesis, we will not describe it further. Trade performed by a security institute involves three different types of actors: *stock brokers*, *asset managers*, and *analysts* [2]. These three actors work together to be able to advise and find the best investment possibilities for their customers. The three actors will be described below.

4.1.1 Stock brokers

Traders can deal with stocks, currencies, or bonds. Traders dealing with stocks are often called stock brokers; traders dealing with currencies are called foreign exchange

members, and those dealing with bonds are called bond-dealers. Traders' work consists of *investing capital on behalf of a customer*, who can be institutions, companies, and private persons. For this work the trader withdraws a commission fee. A trader needs to be constantly updated on what is going on in the market and have knowledge about the companies in the market. News are therefore of great importance for traders. The profession can primarily consist of three different kinds of services: *execution-only*, *advisory*, and *discretionary dealing*. An *execution-only* service means that the trader only carries out the customer's instructions to buy or sell. *Advisory* service is when the trader advises his/her customers on what to buy and sell and the final decision is made by the customer. The trader must be talented at communicating with his/her customers. When providing a *discretionary* service the trader establishes the objectives of the customers and makes all decisions on the behalf of the customer.

4.1.2 Asset managers

Asset managers provide *professional management of financial assets*. Different types of assets can be recommended and invested in, such as stocks, bonds, real-estate, and currencies. The investments can be made on the behalf of private persons, institutions, pension funds, and insurance companies for example. Managers handle collective investment plans, such as funds and portfolios. The service asset managers provide includes an assessment of each customer's needs and appropriate risk level. Based on this assessment recommendations on what to invest in are then given to the customer. Managers provide similar kinds of services to the customer as the stock broker do, i.e. *advisory* and *discretionary management*. Asset management can primarily be divided into two types; the one that expect a relative return and the one that focus on an absolute return. The most common type of management expects a relative return. This means that the manager's responsibility is to perform better than the index he/she is being compared to. The fund or portfolio does thus not need to obtain a positive return, as long as it is higher than the comparison index. Within this kind of management customer contact is limited. A more uncommon kind of management focused on absolute return can also be carried out. This kind of management is for example used in hedge funds. The return should from this kind of fund be positive regardless of the state of the market. Asset managers are sometimes also called investment managers, fund managers, or portfolio managers.

4.1.3 Analysts

The term can consist of several different kinds of analysts such as risk analysts, technical analysts, and statistical analysts. However the term analyst is most often used for fundamental analysts, which are the most common category. The fundamental analyst has the *ultimate responsibility for all investment recommendations*. An analyst mostly observes one specific line of business and needs to be updated with the development and the companies belonging to this business. The analyst's work consists of gathering, packaging and communicating the information further. Important information sources for the fundamental analyst are news, annual reports, and meetings with corporate representatives and management.

4.2 Data and information

All actors on the stock market need to get continuous information about the stock market and the companies listed on the market. Analyses are then performed on the

gathered data to give a complete picture of the market. Two main types of analyses are performed on stock data; *fundamental* and *technical*. *Fundamental stock analysis* is based upon financial facts about the company, such as profits, potential growth, company management, and market situation. Annual and quarterly reports are very important sources of information. *Technical analysis* ignores the underlying factors and instead tries to find patterns in a company's stock price over time, to predict a stock's development (Hansson, 2005; Simunic, 2003). To be able to perform any analysis the actors on the stock market need to have access to different kinds of data and information, such as *stock price*, *financial ratios*, *news*, and *indices*.

4.2.1 Stock price and financial ratio

Information about *stock prices* is of *primary interest* to the actors on the stock market, especially when performing technical analysis. They also look at values such as the highest/lowest price at which the stock was traded during the day or in the past; for example the last year. *Financial ratios* for the stock are also very important. They look at for example dividend, yield, rate of return, P/E and P/S ratios. They are used to evaluate the overall financial condition of the company (Hansson, 2005).

4.2.2 News

News is *one of the most important information sources* for a stock broker, manager, and analyst, since they often change and affect the trend of the market. News comprises everything from product releases, company reports, insider activities, changes in company management etc. This kind of information often affects the market very much, and it is therefore very important that the actors have access to this kind of information instantaneously. This information is today gathered from the software available to the actors, but mostly from different websites (Hansson, 2005).

4.2.3 Indices

Another useful information source about the development on the stock markets is, according to Hansson (2005), stock market index which shows the *composite value of a certain number of stocks*. There are different kinds of stock market indices which can show the development for example of a stock exchange, a sector or a number of large companies listed on one stock exchange. Some well-known indices which can be mentioned are Dow Jones Industrial Average, S&P 500 and NASDAQ Composite. These are all American indices, the first two from the New York Stock Exchange and the last from NASDAQ. Nikkei 225 is a stock market index for the Tokyo Stock Exchange, FTSE 100 is an index for the London Stock Exchange and OMXSPI is one index for the Stockholm stock exchange.

4.3 Information sources

Having access to all this data and information about the stock market is very important. Stock market data can be gathered from several different sources daily, such as newspapers, television programs, radio, websites, and software. The primary information sources for financial professionals are websites and software (Hansson, 2005). The websites and software that financial professionals use regularly in their work today are described in this chapter to give an overview of what the users are accustomed to look at and work with.

4.3.1 Websites

Websites are good and well-used tools for accessing and gathering information about the stock market. Websites are primarily used by financial professionals to search for news about companies listed on the stock market. For example *Google Finance*, *Yahoo! Finance*, and *MSN Money* supply information about different stock markets and companies listed on stock markets all around the world. *Aktiespararna*, *Dagens Industri*, and *Dagens Nyheter* are websites providing information about the Nordic markets. Almost all of these also provide access to advanced charting tools which can be used to make comparisons and analyze the data. However, none of these websites provide information in real-time. There is a delay between 10-20 minutes on the data displayed on the websites. All charting on the websites are also static, not in real-time.

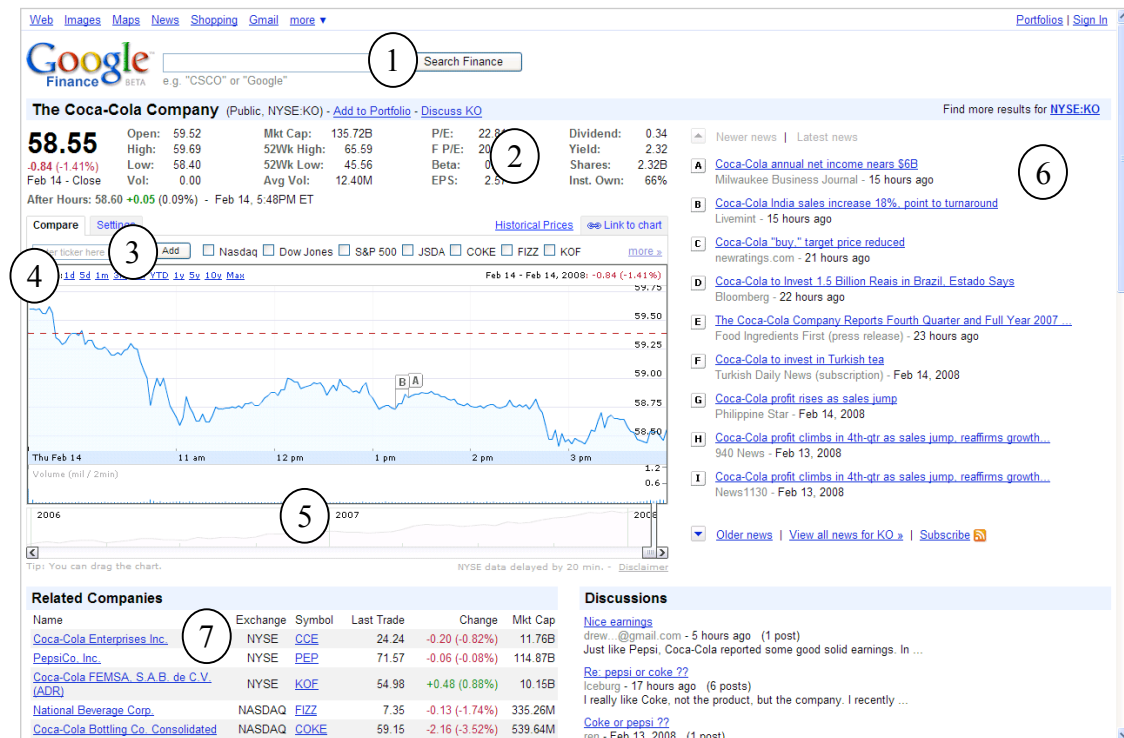


Figure 9. Google Finance's website [3].

In figure 9 we can see an example of Google Finance's website [3]. In this view a search has been made on *The Coca-Cola Company* and information about this company is now displayed in the view. Searches on both companies and indices can be made (1). Stock price and financial ratios for The Coca-Cola Company is displayed (2). The chart displays the stock price development for the company over one day. You can choose to change different settings of the chart. You can add one or several companies or indices to make comparisons with Coca-Cola (3). You can also change the time range of the chart to display between 1 day and over 30 years (4). Another useful feature in this chart is the time slider (5). With this slider you can change the time range of the chart and very easily go back and forth in time. News concerning Coca-Cola is also displayed in the view (6). The news is also marked on the chart. Information about companies related to Coca-Cola can be found in the table below the chart (7). The rest of the view contains information related to Coca-Cola, such as tables of the management of the company, key stats, income statements, balance sheets, cash flow and so on.

4.3.2 Software

It is important to be able to process and get more detailed analyses of the data, since it can give a more complete picture of the market and help you make successful decisions. According to Hansson (2005) you can either choose to subscribe to different stock analyses or perform these kinds of analyses by yourself. To be able to get access to, process and analyze the financial data, different kinds of software is used by financial professionals. There are three main categories of software: *trading systems*, *depot systems*, and *information systems*. The trading system is connected to the stock exchange and is used to place bids, which can also be placed using the telephone. The depot system is used to keep track of the clients and their activity. It contains information related to the client, such as stock holdings and transactions, both current and historical. The trading and depot systems are necessary in order to perform trade, but are not used so actively. The system that is primarily used is the information system. It provides the actors on the stock market with continuous information about stock market data. This kind of system offers a combination of real-time, historical, and fundamental data on stocks, indices, and news. There are different information systems available on the market; the most commonly used systems for Swedish stock trading comes from *Reuters* and *SIX*.

4.3.2.1 Reuters

Reuters have a number of different tools for the stock market, but the primary tool for financial professionals is *Reuters First*, which provides the user with real-time and historical price information, the ability to perform technical analysis and portfolio management. Information is mainly presented in tables, but also three types of charting are also available: *pie charts*, *bar charts*, and *line charts* [4].

Reuters First has a window based interface. The user can compose an own view by adding or removing different kinds of windows [5]. In Figure 10 an example of a view in Reuters First is shown. All windows are linked together and display data and information about the selected stock, in this case Nokia. You select a stock in the table to the left (1). The other windows will update dynamically depending on what company is chosen in this table. The other tables in this view display closures (2), the order book (3) and news concerning Nokia (4). A line chart shows the development in stock price for Nokia (5). Another line chart compares Nokia to an index and also shows the RSI and volume for Nokia (6). The last chart displays the broker stats for Nokia (7).

THE STOCK MARKET

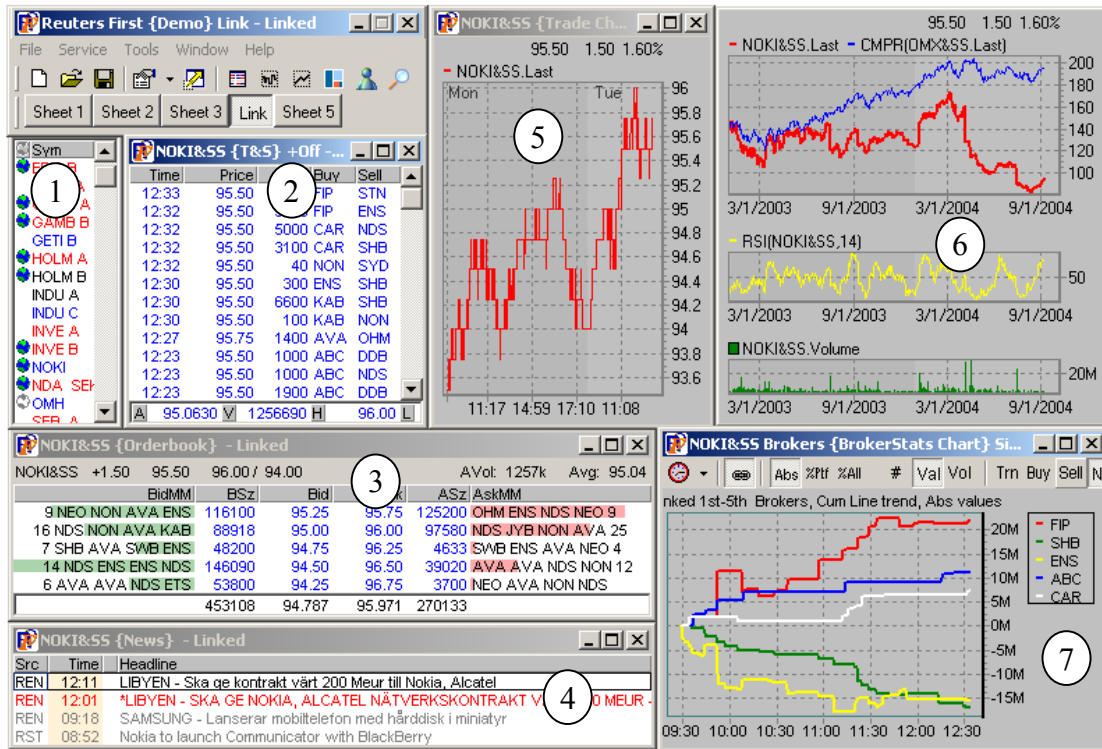


Figure 10. Reuters First [5].

4.3.2.2 SIX

SIX develops the second most commonly used information and transaction system in Sweden, called *SIX Trader*, which provides the same information as Reuters First, i.e. real-time and historical stock prices, and news. *SIX Trader* can combine and display different types of documents, such as tables, text areas, graphs, and event logs. Information is primarily presented using tables, secondarily by using charts. The most common charts in *SIX Trader* are line charts and bar charts [6].

Different types of documents in *SIX Trader* can be composed in one view to show the data and information the user wishes. Several different views can be composed and are stored on different tabs [6]. Figure 11 shows an example of a view composed in *SIX Trader*. It contains three tables, one text area, one event log and four real-time graphs. Information about stock prices and financial ratios in the form of stock tables are displayed (1 and 2). A table containing the incoming news is also shown (3). The news and articles are displayed and can be read in the text area (4). The event log (6) is used to monitor interesting data and information. Real-time graphs in the form of line charts and bar charts are also shown (5, 7 and 8).

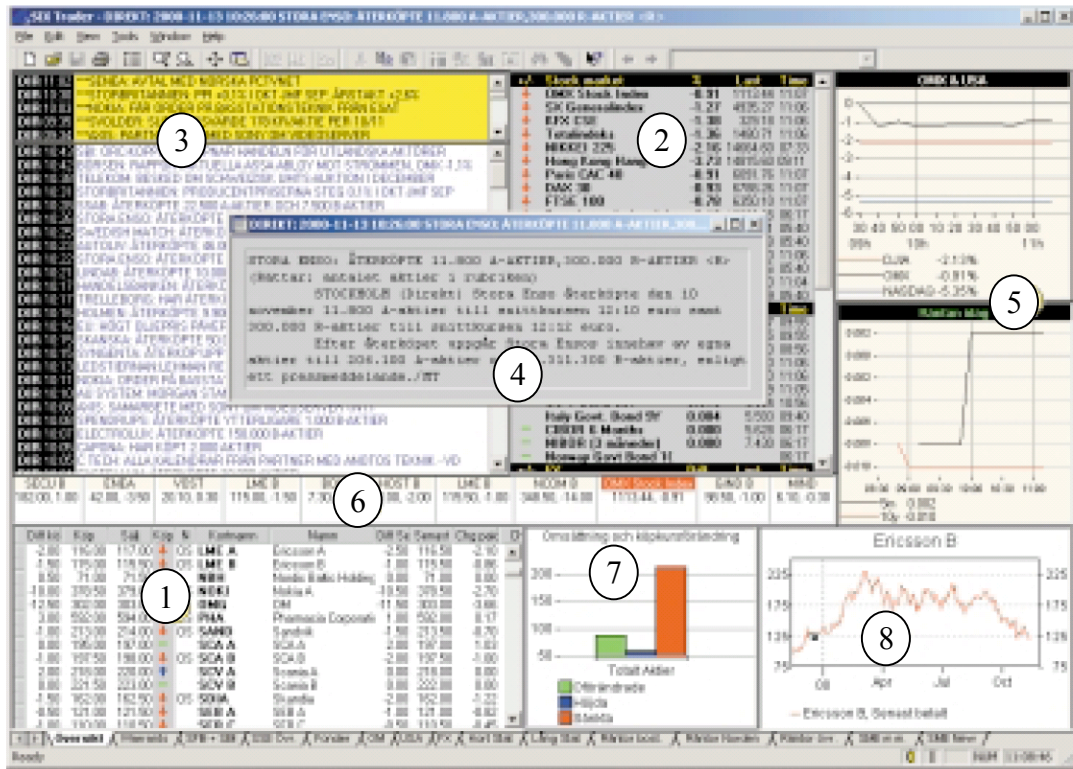


Figure 11. SIX Trader [6].

4.4 Present display mediums used for visualizing stock market data

The different information sources used by financial professionals today, such as websites and software, primarily present the information using tables. However, in some cases display mediums are also used to visualize the stock market data. The most commonly used at present are *high/low charts*, *candlestick charts*, *bar charts*, and *line charts* (Andersson & Svensson, 2001).

4.4.1 Open/high/low/close chart

Open/high/low/close, ohlc, charts show a linear representation of the development in stock prices. The y-axis most often represents stock price or stock market index, while the x-axis represents time (see Figure 12). The values shown in the chart are the highest and lowest stock prices during the day and also the stock price at the end and at the beginning of the day (see Figure 13). The ohlc chart is a basic chart, which is both simple to draw and understand (Swedbank Markets, 2005).

THE STOCK MARKET

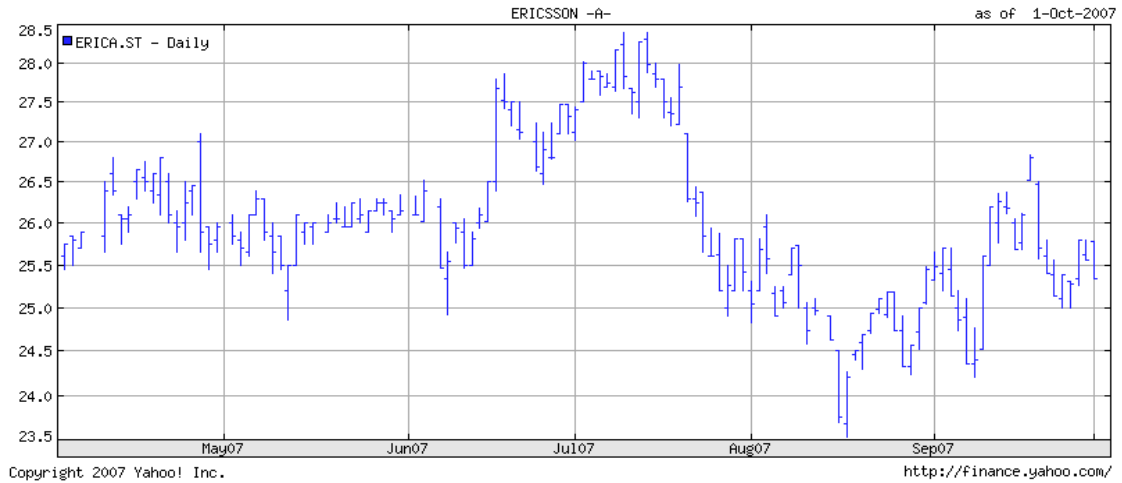


Figure 12. Open/high/low/close chart [7].

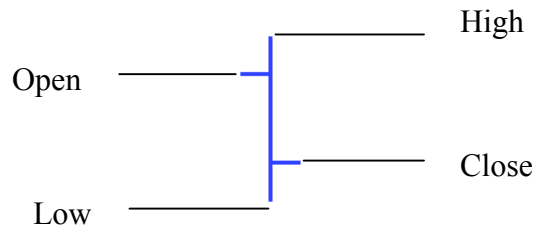


Figure 13. Open/high/low/close bar.

4.4.2 Candlestick chart

A chart that is similar to an ohlc chart and also used on the stock market is a candlestick chart (see Figure 14), which shows the same values as an ohlc chart, i.e. the highest and lowest stock price during the day and the stock price at the end and at the beginning of the day. Depending on whether the stock price has gone up or gone down during the day, the candles in the diagram have different colors, light if the stock price has gone up and dark if the stock price has gone down during the day, see figure 15 (Nesbitt & Barrass, 2004).



Figure 14. Candlestick chart [7].

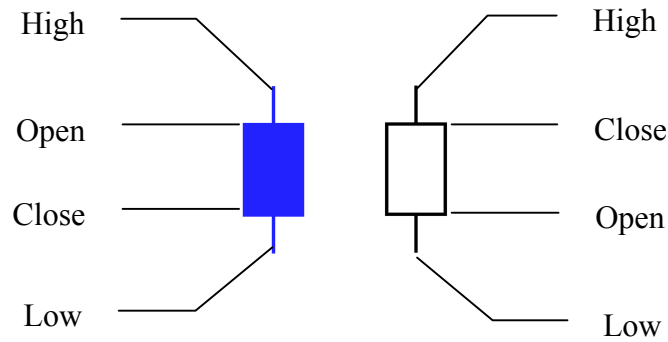


Figure 15. Candlestick bar.

4.4.3 Bar chart

Within the stock market bar charts are mostly used to display the order book and volume. The order book is often displayed using horizontal bar charts; while volume often is shown with vertical bar charts. Figure 16 displays volume for the Dow Jones index. The y-axis represents volume in billions, while the x-axis represents time.

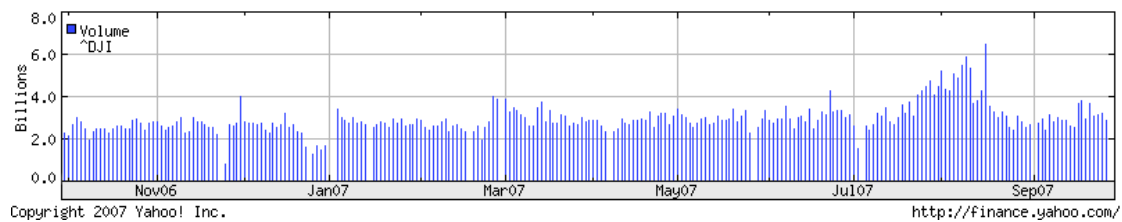


Figure 16. Bar chart [7].

4.4.4 Line chart

Another commonly used diagram is the line chart, which shows development over time, like ohlc and candlestick charts, but has the advantage of being easier to comprehend when making comparisons over time. They are often used for displaying and comparing different stock market indices and stock prices (see Figure 17).

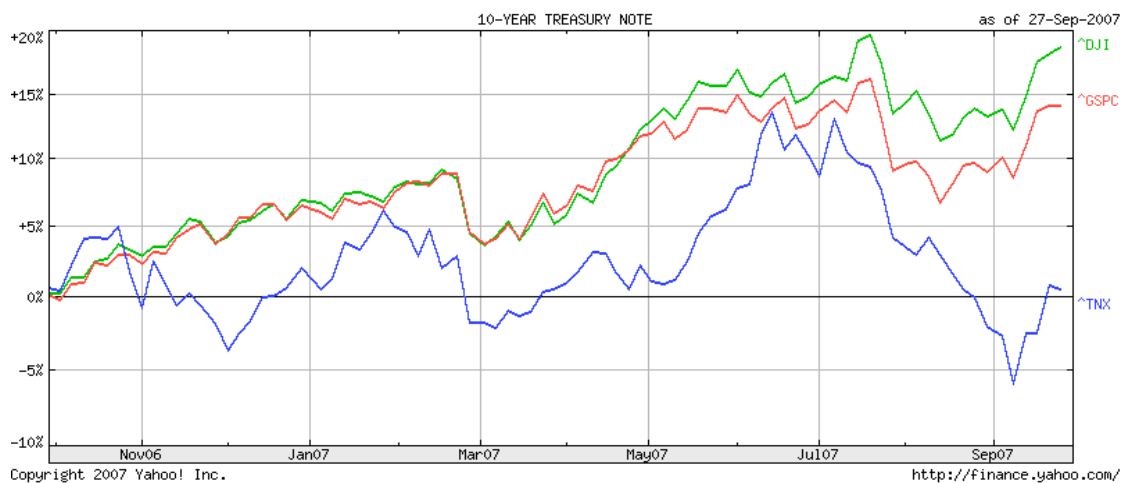


Figure 17. Line chart [7].

4.5 Upcoming display mediums for visualizing stock market data

The visualization techniques most commonly used today are good at displaying for example development over time, sums, and averages. Financial professionals also need to get an overview of the large amounts of hierarchical data and be able to find hidden patterns, anomalies, and outliers. Ohlc charts, candlestick charts, bar charts, and line charts are not suitable for this. Another visualization technique has therefore large possibilities to establish itself within the stock market and has also started to make its entry into this field. This visualization technique is called treemaps.

Some software containing treemaps have already been developed for the stock market. Among these are *Panopticon's* [8] solution for displaying financial data for trade monitoring, asset management, and risk analysis (see Figure 18), and the *Hive Group's* [9] solution for stock analysis and portfolio management (see Figure 19).

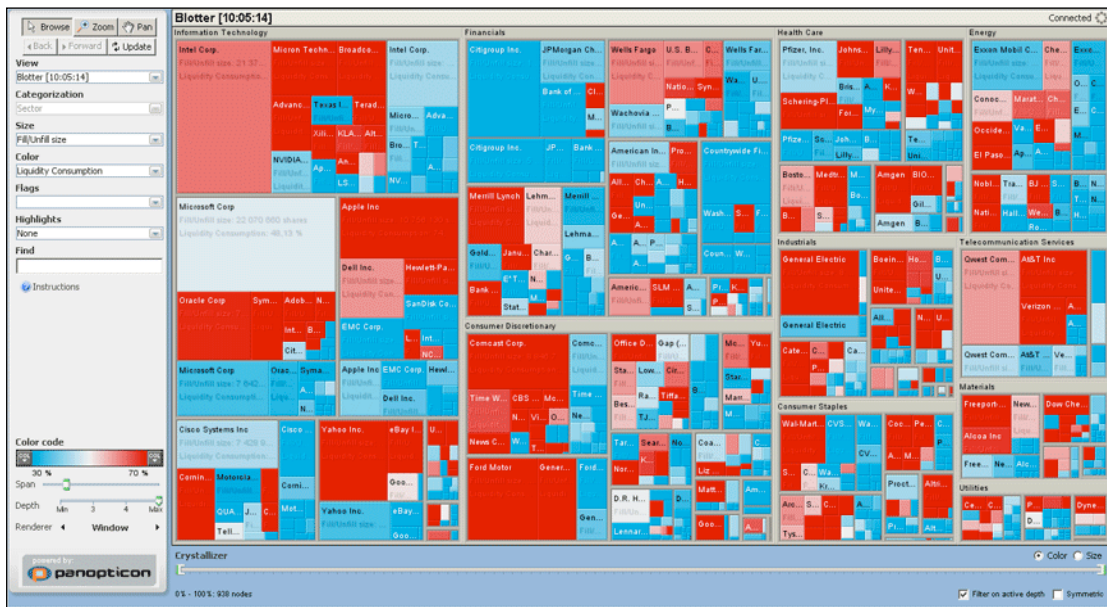


Figure 18. Panopticon's treemap [8].

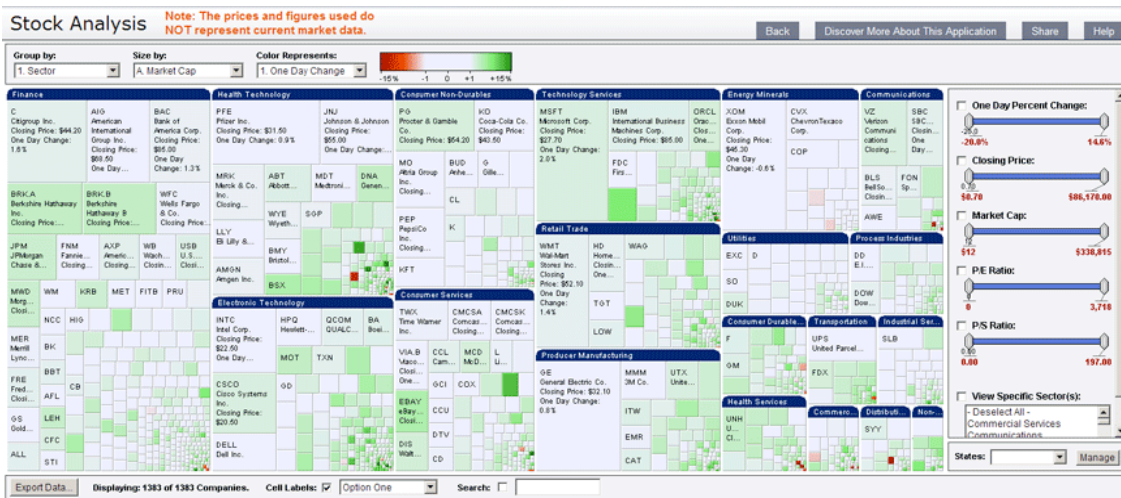


Figure 19. The Hive group's treemap [9].

Another upcoming visualization technique within the financial market is sparklines. They have the advantage of being very space-efficient, which is very suitable when displaying financial data since large amounts of data needs to be displayed. Sparklines are good substitutes for both bar charts and line charts when the space available is limited. Some applications containing sparklines have already been developed for the financial market. One example is *FusionGadgets* [10], an application that can be used for monitoring financial real-time data. See figure 20 for an example of a dashboard created with FusionGadgets containing real-time gauges and charts. At the bottom of the dashboard sparklines can be seen, both in the form of bar charts and line charts.

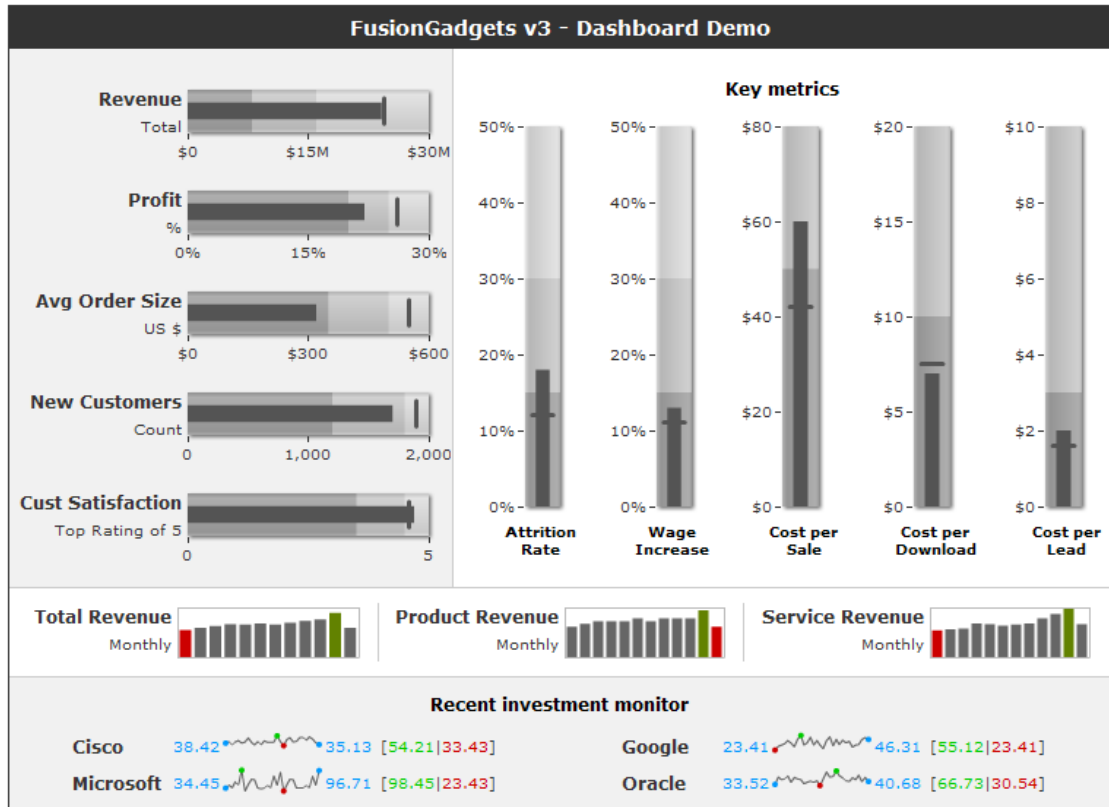


Figure 20. *FusionGadgets* dashboard containing sparklines [10].

5 Method

This section describes how interaction design can be seen as a research field and different models describing the work process within this field. Various methods and techniques commonly used in interaction design which are relevant to this thesis work are also described.

5.1 The interaction design process

There are different overall approaches on conducting research, and the two most well-known approaches to science are, according to Mackay and Fayard (1997), the *deductive model* and the *inductive model*. Mackay and Fayard however suggest another model for the field of Human-Computer-Interaction and interaction design, since these fields are multidisciplinary research fields. Interaction design is not pure natural science, nor a pure design discipline. It borrows techniques from other fields, and therefore neither the inductive nor the deductive model, without modification, can be said to describe research within interaction design. Mackay and Fayard mean that interaction design research is not solely to observe a phenomenon or to explain it. Neither is it solely about conducting experiments to verify a hypothesis. Interaction design studies interaction between people and artifacts. Within interaction design new artifacts are designed, and these same artifacts affect the actual interaction, between people and artifacts, that is under study. Therefore, Mackay and Fayard propose a framework that better suits the field of interaction design in combining the deductive model and the inductive model with the design of artifacts, see figure 21.

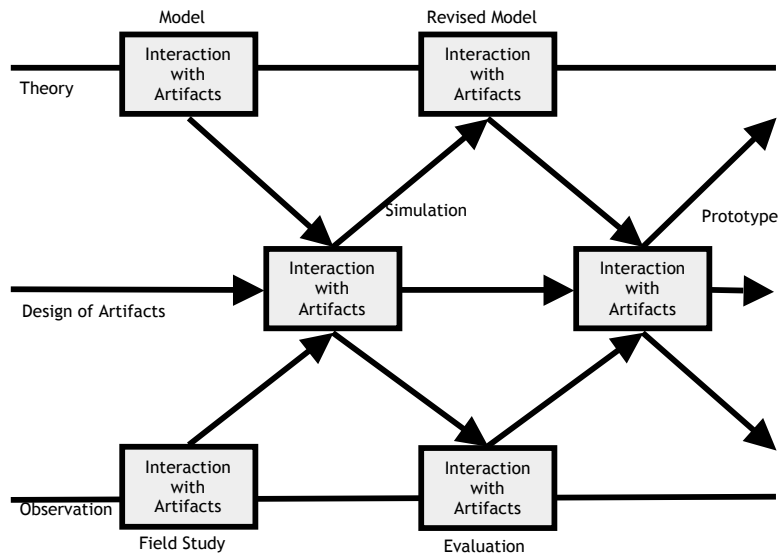


Figure 21. Mackay and Fayard's framework for research in interaction design (1997).

In addition to these overall approaches to research there are a number of more detailed models which describe the work process in interaction design. Some of these models will be discussed in the following sections.

5.2 Models

A model of the design process describes which activities it should consist of, and in which order they should be performed. There are many different models to choose from for the interaction design process, for example *software engineering models*, *usability models* and *iterative models*.

A well-known *software engineering model* is the *Waterfall Lifecycle model*. This model is sequential in character, which is rather impractical for an interaction design process since it often is required to go back and for example revise requirements later on in the design process. Further, the Waterfall Lifecycle model does not explicitly include users in the process (Gulliksen & Göransson, 2002).

Usability models originate from the field of Human-Computer Interaction. The *Star Lifecycle model* is one of these. In this model the different activities are not ordered and evaluation is essential. This model is extremely flexible, which may also be its main disadvantage since it reduces the possibility of getting an overall view of the process (Preece et al., 2007).

Iterative models are commonly used in interaction design since they are flexible in their nature and involve users in the whole design process, which is important within interaction design. Preece, Rogers and Sharp (2007) propose an iterative model for interaction design that takes into account the four basic interaction design activities, which are to identify needs and establish requirements, (re)design, build interactive versions, and evaluate the design. The iterative model also takes the three user-centered design principles into account, which are early focus on users and tasks, empirical measurement and iterative design. The iterative model proposed by Preece et al. can be seen in figure 22. According to this model a project starts with identifying needs and establishing requirements. Then alternative designs should be made conforming to the findings in the previous phase. The next phase is to develop interactive versions of the designs and evaluate them. After evaluation it is possible that there is a need to go back to find out more about requirements and needs, otherwise the design should be redesigned and evaluated again. The work will continue in this fashion until the final design solution is evaluated. The number of iterations in a project depends mostly on the resources available.

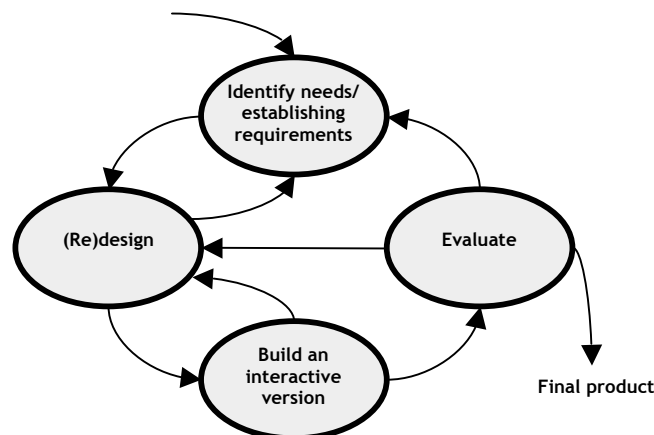


Figure 22. An iterative lifecycle model (Adapted from Preece et al., 2007).

5.3 Techniques and methods

Many different techniques and methods can be used throughout the design process. Which ones to use depend on many factors, for instance at which phase you are in the design process, and what the purpose of the method is. When gathering data to identify needs and establish requirements; interviews, questionnaires and observations are useful and common methods. During the design phase low- and high-fidelity prototypes are usually designed and redesigned to try out different alternatives. The prototypes are then evaluated using methods like usability tests and expert evaluations. The evaluations can be either formative or summative. A formative evaluation is used during the development process to find out if the design meets the user's needs and which improvements that can be made, in contrast to a summative evaluation which is performed to assess how well a finished product works (Preece et al., 2007).

5.3.1 Quantitative versus qualitative studies

The kind of project to be performed should decide which methods should be used and whether they should be qualitative or quantitative. The choice is not always easy or obvious but there are some key points to consider. According to Trost (1997), *a quantitative research method* should be used if the goal is to be able to say something about the frequency of a certain phenomena; how often, how many, or how common. *A qualitative research method*, on the other hand, is more appropriate if the goal is to understand the ways in which people reason and react to some phenomena, or to see variance in peoples' behavioral patterns.

In interaction design qualitative methods are more frequently used than quantitative ones for the simple reason that interaction design is very much about designing for the user, meaning we must find out what the user needs and wants, which is best done with qualitative methods. Since we have exclusively employed qualitative methods in our thesis work we will not further discuss quantitative methods.

5.3.2 Gathering data

Different data gathering methods have their advantages and disadvantages, and which ones we choose will affect the data we find. One method will capture one aspect of the situation but most likely miss another. It is therefore a good idea to combine techniques in order to get as unbiased data as possible as well as getting validity of the results. Combining techniques is called *triangulation* (Frankfort-Nachmias & Nachmias, 1996; Silverman, 2005). The relationship with the participants is another important issue. The purpose of the study should be explained carefully to the participant before starting the study; it is important to make the participants feel comfortable with the situation (Bell, 2000; Preece et al., 2007). Before conducting the real study it is important to conduct *pilot studies*. The reason for performing pilot studies is to find the problems, e.g. poorly formulated questions and unclear instructions, to be able to change the setup before conducting the real study. Ideally the pilot study should be conducted with a person from the same user group as the main study, but in case this is not possible it can be performed with someone else in order to get some input on the study (Bell, 2000; Preece et al., 2007).

When gathering data it is important to record it in some way. Taking notes, taking photos, video recording and audio recording are commonly used techniques. Which technique or combination of techniques to use depends largely on the situation.

Taking notes is least intrusive, but it can be difficult to keep up during an interview or observation. It can also be hard to understand the notes afterwards. There is always a risk that the person taking the notes can be biased, so it is a good idea to be more than one person gathering the data. Taking notes works well in interview situations and when gathering data in the field. *Video* and *audio recording* give a fuller view of the situation, but is very time-consuming to analyze and transcribe (Preece et al., 2007).

5.3.3 Personal interviews

Personal interviews are commonly used for data gathering and can vary very much in how controlled they are; the interviews can be structured, unstructured or semi-structured (Bell, 2000).

Structured interviews are standardized studies where every question is planned in advance and no follow-up questions are asked. Questions are kept short and should be worded exactly the same way to all respondents. *Unstructured interviews* are open and exploratory and the interviewer has less control over the interview. The outcome depends to a large extent on the interviewer's ability to steer the interview in the right direction, and asking follow-up questions in order to get answers to the questions. *Semi-structured interviews* are a combination of the previously mentioned interviews; both open and closed questions are used. The interviewer has a guide to follow, and questions are planned in advance, but the interviewer also probes, that is, follow up the questions to get more information (Frankfort-Nachmias & Nachmias, 1996; Karlsson, 2007; Preece et al., 2007).

It is often easier for a participant to discuss and answer questions if he/she has something to talk about. To use some kind of representation, a *mediating tool*, of the concept or product can trigger the participant(s) to talk during the interview. The representation could be anything from a verbal explanation of a concept, to a part of a prototype, or a fully functioning final product (Karlsson, 2007).

Some *advantages with interviews* as data gathering method are that the interviewer has, to a fairly large extent, control over the interview situation. Interviews give fuller information than for example questionnaires, since the interviewer can ask follow-up questions, called probing, as well as capturing spontaneous reactions and gestures from the respondents. One *disadvantage with interviews* is that there is always a risk for interviewer bias (Frankfort-Nachmias & Nachmias, 1996). When *planning the interview* and making the questions for the interview there are some things to consider. The interviewee must feel comfortable with and understand the language used by the interviewer. Questions should be formulated so that only one question is asked at the time. Long questions with compound sentences can be confusing for the respondent. Ask the questions in an impartial and non-leading way, otherwise the respondent may not say what he/she really feels about a certain issue. When preparing the interview it is advisable to think about possible answers to the questions and include them in the interview guide, as well as preparing for follow-up questions (Bell, 2000; Preece et al., 2007).

A commonly used technique is to arrange the order of the questions according to the *funnel technique*. The interview starts with general questions to make the respondent comfortable with the situation. This is followed by the main part of the interview with more detailed and specific questions. At the end it is a good idea to ask a few neutral

questions to round off the interview. These questions should be less sensitive and easy to answer (Karlsson, 2007; Preece et al., 2007).

5.3.4 Observations

Preece et al. (2007) suggest that observations can be used at an early stage to help specifying the users' requirements and needs as well as at a later stage for evaluation purposes. In this section we discuss observation methods relevant for this thesis, i.e. usability testing and contextual inquiry.

5.3.4.1 Usability testing

Usability testing is a commonly used method to assess a system in terms of effectiveness, ease of use and user satisfaction (Preece et al., 2007). It takes place in a *controlled environment*, for instance a usability laboratory. Focus is on details of what the user does, e.g. how long it takes to perform a certain task, which buttons are pressed, which menus are entered. Participants are asked to *perform a set of pre-defined tasks* with the system and while doing so he/she is observed and is often recorded for later analysis. The *think-aloud technique* is successful to use to get around the problem of not being able to observe thoughts and feelings. The participants are asked to talk out loud regarding what they are thinking while performing the tasks. This makes it possible to find out what they are intending to do, if they are confused about something etc.

When the participants have finished performing the tasks they can either be asked to fill out a questionnaire or an interview can be conducted to capture the respondent's opinions and feelings about the product (Preece et al, 2007). 5-12 participants are normally considered to be enough to find most usability problems, but conducting a test with less participants is also useful. Shneiderman and Plaisant (2005) points out two *problems with usability testing*. Since the participant is faced with a graphical user interface for the first time it is difficult to know how it would work after being used on a regular basis. Further it is only possible to try a limited amount of features in the system during one test session. It is therefore useful to combine usability testing with expert evaluations to get a fuller coverage of the usability problems.

5.3.4.2 Contextual inquiry

Contextual inquiry is a *combination of an interview and an observation* and is commonly used to gather requirements. It is especially useful when the context of use is of importance for the product to be developed. The goal is to get insight into and *understand the user's work situation*. A contextual inquiry takes place in the user's normal work environment. An interview as well as an observation is carried out while the user conducts normal work tasks. The interviewer interprets the observations and discusses them with the user to clarify the situation. Contextual inquiries are often recorded since it otherwise can be hard to capture both the discussion as well as the user's actions (Löwgren & Stolterman, 2004a; Preece et al., 2007).

5.3.5 Expert evaluations

Expert evaluations can be used to reveal usability problems with an interface if users are difficult to access. It is a cheap and quick method, which can be used at any stage in the design process and successfully in combination with user tests. Two common types of expert evaluations are heuristic evaluation and cognitive walkthrough (Preece et al., 2007).

5.3.5.1 *Heuristic evaluation*

Heuristic evaluation was developed by Jakob Nielsen and Rolf Molich and is a method where usability experts *evaluate an interface looking for usability problems* with the help of a set of principles called heuristics. It is recommended to use three to five evaluators in order to find most usability problems. Nielsen has revised the set of heuristics to the following ten [11]:

- Visibility of system status
- Match between system and the real world
- User control and freedom
- Consistency and standards
- Error prevention
- Recognition rather than recall
- Flexibility and efficiency of use
- Aesthetic and minimalist design
- Help users recognize, diagnose, and recover from errors
- Help and documentation

5.3.5.2 *Cognitive walkthrough*

In a cognitive walkthrough, *experts take on the role of the user* and perform a number of predefined tasks *looking for usability problems*. While performing the tasks the expert expresses whether he/she believes the design will be understood by the user. Based on these statements the design can be revised to better fit the user's needs. Preece et al. (2007) mean that the problems found in cognitive walkthroughs are usually of a higher level of detail than the ones found in heuristic evaluations and that cognitive walkthroughs can be time-consuming.

5.3.6 *Personas*

Preece et al. (2007, p.481) describe personas as “*rich descriptions of typical users of the product under development that the designers can focus on and design the product for.*” Further, they say that personas are not real people, rather they are many users mixed into one. They should be based on real users encountered during user studies. A persona should focus on user goals, but should also *describe skills, attitudes, tasks, and environment in detail*. It should describe a realistic person and not an idealized one. Therefore the made-up user should have a name, as well as a few details describing his/her personal life. It is often useful to make more than one persona for a particular design; one can then be used as the primary persona to design for.

5.3.7 *Scenarios*

Scenarios are *narratives about users and their activities*. Preece et al. (2007) suggest that scenarios can be used in various stages of the design process and for various reasons. They can for example be used for describing user tasks that has been established from data gathering, or being a basis for the overall design. After data gathering has taken place, scenarios can help when documenting what has been found. Preece et al. (2007) say that a scenario should describe the user's goals and not how he/she explicitly would interact with any technology to achieve that goal. The language should be informal to be easily understood by different stakeholders. Scenarios are a way to explore contexts, needs and requirements.

5.3.8 Brainstorming

Brainstorming is a commonly used method to *quickly generate many ideas* about a topic and can be used at any phase of the design process. A brainstorming session has three steps: a group of people is gathered, ideas are generated, and the ideas are systematized (Jones, 1992; Löwgren & Stolterman, 2004b). The participants should generate as many ideas as possible during the brainstorming session and it is important to declare in advance that no ideas or thoughts should be criticized. Instead, all kinds of ideas should be encouraged as well as attempts to improve others' ideas during the session. Löwgren and Stolterman (2004b) point out that the idea is that the participants should be inspired by each others' ideas and together come up with new ones. Each idea should be written on a separate note, since it makes it easier to sort them after the session. To sort and systemize the ideas from the brainstorming it is a good idea to use another design method, e.g. the affinity diagram.

5.3.9 Affinity diagram

The affinity diagram is successful to use to *systemize information*, e.g. ideas that have emerged during a brainstorming session, or information that has been collected during field studies. There are many variations of this method but the essence is that a group of people sort the ideas/information into groups in which they seem to belong. The participants should collaborate to find groupings for each piece of information or idea. It is common that many ideas seem to belong to more than one group and that some ideas need to be clarified, and it is therefore useful to try different ways to group them. When the participants have agreed on the groupings each group should be named (Löwgren & Stolterman, 2004b).

5.3.10 Prototyping

In interaction design, prototyping plays an important role. By building interactive versions of the design it is possible to try out design ideas and evaluate them. Designing, evaluating and then redesigning is an important cycle in the iterative design process. The more iterations, the better the system will turn out in the end (Preece et al., 2007). In a user centered design process prototypes should be used early on and throughout the process, starting with simple paper prototypes and later on as more complex prototypes, since prototypes are valuable in many ways. Gulliksen and Göransson (2002) state the following possibilities when performing prototyping: you can *explore new solutions, test functionality, find requirements, practice creativity, find weaknesses, test performance and layout, try sequences of commands and develop systems*. Prototypes can vary in complexity and it is common to talk about low- and high-fidelity prototypes.

5.3.10.1 Low-fidelity prototypes

A low-fidelity prototype is a very *simplified version of the final product* and is usually not made of the same material as the final product. Low-fidelity prototypes can be made in many ways, but one common way is to make paper-based prototypes to show the design of a interface. One kind of prototyping is storyboarding, which is a series of sketches showing, step by step, how users can perform tasks using the system (Löwgren & Stolterman, 2004a). The *advantages with low-fidelity prototyping* are that they are quick and cheap to make and also to modify, which makes them very suitable to use in early stages of the design process. At that point it is very important to try out and evaluate many different ideas and alternatives at a conceptual stage before the design gets more detailed. There are some *disadvantages with low-fidelity*

prototypes. For example, a user cannot explore the interface on his/her own since it is very limited in complexity, they are not very useful when it comes to usability testing, and it is difficult to simulate some kinds of interaction, such as direct manipulation (Löwgren & Stolterman, 2004a; Preece et al., 2007).

5.3.10.2 High-fidelity prototypes

A high-fidelity prototype *looks more like the final product* and its material is usually also more alike the final product. Adobe Flash, Visual Basic and such tools are often used to create high-fidelity prototypes (Preece et al., 2007). *Advantages with high-fidelity prototypes* are that they often have full, or at least some, functionality and a higher degree of interactivity than for example paper-based prototypes. Some *disadvantages* are that they are more expensive and time-consuming to develop, not efficient for proof-of-concept designs, or effective when it comes to establishing requirements. Another potential problem with high-fidelity prototypes is that if the prototypes have a lot of functionality and look like a finished product, the user may think that it is the real system. This could be dangerous because designers could then settle for the solution as it is, without exploring other alternatives; since they have found a working solution that the users' like (Preece et al., 2007).

5.4 Planning

We see our design process as following the framework suggested by Mackay and Fayard (1997), which can be seen in figure 21, since we want to start out with a theory as well as observing the real world. When it comes to models for the work process, we wanted to follow an iterative model, similar to the one proposed by Preece et al. (2007), which can be seen in figure 22. This is an accepted model within interaction design and it allowed us to divide our work into different iterations. However, we were aware that users could be difficult to find, meaning it could be difficult to meet with them many times to find more needs and requirements as the design process went forward. Therefore it would probably not be possible to work in complete iterations. The plan was therefore to work iteratively, but we were aware that it might not be feasible to follow the model entirely. We planned for performing three iterations, described further below. See figure 23 for a time plan of the work.

5.4.1 First iteration

In the initial phase of the thesis work we planned to study theory about the information visualization field in order to get a deeper understanding about available techniques and their possibilities; specifically looking at earlier work involving information visualization of real-time data. Getting acquainted with the product TIBCO Spotfire was another part of this phase, as well as gaining an understanding of the stock market, and finding a suitable user group for the work.

The next phase that was planned for was a study directed at the target group to gain a deeper understanding of the users and their work. The study was to be a contextual inquiry since this kind of study gives the opportunity to observe the users in their real context, which is very important. This method provides an opportunity to clarify many questions about the users' situation. As we had gotten the expression that it could be very hard to find users, since they were very busy, we decided not to specify an exact number of participants for this study in advance.

METHOD

After performing a user study we planned to analyze the result and put together a requirement specification to get an idea of what different needs and requirements the design had to support. This requirement specification was then intended to be brought into the design phase and a number of low-fidelity prototypes were to be designed to fit these requirements. As many different alternatives as possible should be constructed on paper in this low-fidelity stage of the design process.

The paper prototypes of the designs should then be evaluated primarily by users, but if not possible, through an expert evaluation. This evaluation should be formative and rather informal in nature. The reason for the evaluation was to get feedback on the designs and then decide which alternatives to develop further.

5.4.2 Second iteration

After performing an evaluation we wanted to enter our second iteration and start to develop high-fidelity prototypes. Since we did not think that we would be able to get hold of users at this stage and collect more requirements and reassess the requirements already collected, we planned to head right for the design phase and develop three high-fidelity prototypes in Adobe Flash, based on the result from the evaluation of the low-fidelity prototypes. The high-fidelity prototypes were then to be evaluated through usability testing. The usability testing planned for at this stage should be more extensive than the previous evaluation and include as many users as possible within the given time limit.

5.4.3 Third iteration

When we had performed our final evaluation of the high-fidelity prototypes we wanted to enter our third iteration. This iteration was to involve further development of the concept on the basis of the result from the evaluation at the end of the second iteration. During this iteration we wanted to look at how the prototype should be re-designed if work with the prototype was to continue. Since we did not know how much time we would have for this phase, we did not specify in advance how this development of the concept would be done; it could result in anything from a written description to sketches or another high-fidelity prototype developed in Adobe Flash.

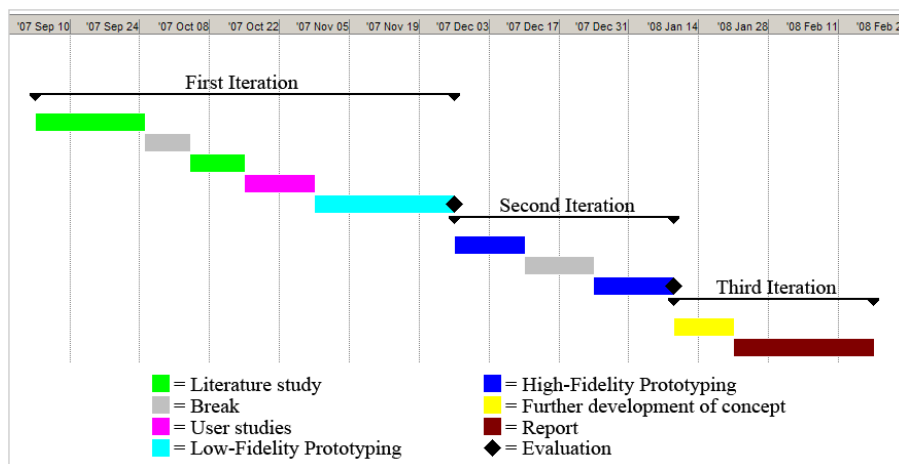


Figure 23. Time plan.

6 Realization

This chapter describes our process throughout this thesis work. Four different iterations have been conducted, each containing several different stages. Needs and requirements have been gathered through a pre-study, a requirement analysis has been conducted and several different design phases and evaluations has been performed.

6.1 First iteration

Our first iteration contained all of the four steps in the iterative model described by Preece et al. (2007), which are to identify needs and establish requirements, design, build an interactive prototype and evaluate the prototype. We started out with a pre-study to identify the needs of the users. After identifying the needs we analyzed them to be able to create a requirement specification. Thereafter we started to design and build a prototype. The prototype was evaluated at the end of this iteration.

6.1.1 Pre-study

The pre-study we conducted contained studies of relevant literature along with relevant software. To identify needs and requirements we also conducted several interviews with users.

6.1.1.1 Literature and software study

To get an overview of information visualization, the stock exchange and real-time data we read books, articles and looked at web pages. Examples of keywords used, in various combinations, in the databases ACM and IEEE were: *information visualization, stock market, stock exchange, real-time data, visualization*, etc. Additional keywords were used to search for articles in areas concerned with real-time data, e.g. security systems and nuclear power plants, for example *security, visualization, real-time data, intrusion detection, perception, nuclear power plant, dashboard design, control system, monitoring* etc. We also studied different kinds of software, starting with *TIBCO Spotfire* since we needed to get a good understanding of how the software works today. To get an insight into the tools used in the financial market we studied software like *Reuters First* and *Six Trader*, and applications like *Panopticon, Map of the Market, NASDAQ's heat map* and *The Hive Group*. Web pages concerned with financial data, i.e. *Google Finance, Yahoo! Finance, MSN Money, Aktiespararna, Dagens Industri*, and *Dagens Nyheter* were also studied.

Unfortunately the literature and software study was interrupted several weeks in advance, due to the fact that we got an opportunity to meet users in London.

6.1.1.2 Interview in London

Two employees at a large international investment bank, a customer to the Company, had agreed to meet with us at their office in London. This was a great opportunity for us, but preparing for this meeting was difficult since we were still many weeks away from when we planned to finish the pre-study, i.e. still studying the stock market and therefore not as read up on the subject as we would have wanted. It was difficult to know if we would and could ask the right questions. In planning the thesis work we

had hoped to make contextual inquiries in the pre-study since this would give us an understanding of the tasks the users performed and in which context. But since the Company had set up this meeting and it was not possible to get hold of the respondents, we knew very little about what kind of situation to expect, as well as who the respondents were in terms of work description etc. However, we put time and effort into preparing for a variety of situations, both interview and observation, and constructed two different interview guides, depending on the time we would have for the meeting, and one observation guide. The *interview guides* were semi-structured and were constructed following the funnel structure, starting with “easy” questions about the respondent, then questions about current software, how trade is performed, real-time data, which data and information they look at, how it is visualized, and ending with rather broad questions about problems and possibilities (see Appendix 2). The *observation guide* had the same blocks of questions as the interview guides. We did not have access to anyone with domain knowledge to conduct a real pilot test with so we tried the interview guide on each other to get an idea of how long the interviews would take, if the formulation of the questions worked and which questions needed to be reworked. The problem of not having access to someone with domain knowledge persisted during the thesis work and we tested out our guides on each other in all the interviews.

Unfortunately, one respondent cancelled our meeting, so we only interviewed one person at the bank. The meeting took place at his workstation in an open landscape office. We were accompanied by our supervisor as well as another member of the user experience team at the Company, and we took turns asking questions. It turned out that our respondent was not a part of our intended user group since he worked in the IT department of the bank. This made it impossible to ask most of our prepared questions, and therefore, the visit to London was unfortunately not as rewarding as we had hoped.

Some of the *main findings* from the interview were that whenever real-time data was shown, it was presented in tables, updates were shown by flashing colors, he did not look at line charts since he was in the bond market where changes over time was of no interest, updates should be automatic and about half of the data he looked at was in real-time.

We also tried contacting other sources throughout the pre-study, such as sales people working at the Company, since they have much contact with people in the financial market, but this was without success since they are very busy and did not have time to answer questions.

6.1.1.3 First interview at a Swedish bank

After the London visit we continued our literature study and started planning for meeting more users. We contacted a number of Swedish banks and got a positive response from the person in charge of the western region’s stock trade at a large Swedish bank. We would have wanted to conduct a contextual inquiry, but this turned out to be impossible in all banks that we contacted and therefore we conducted interviews instead. We used the same *interview guide* as in the London interview, but in a revised form, although the same areas were covered. We used a *mediating tool* in the form of pictures of different visualization methods, such as line charts, bar charts, candlestick charts, tree-maps, heat maps etc. because we hoped it would be easier for

the respondent to talk about visualizations if he had some examples of different alternatives. According to Karlsson (2007) this is exactly what mediating tools should be used for.

We interviewed our respondent in a meeting room at his office in Göteborg and the interview was documented by taking notes since the respondent did not feel comfortable being recorded on tape, due to secrecy policies at the bank. Both of us acted as interviewers, although one of us had the main responsibility for each area we discussed. This made it possible for the one not having the main responsibility to concentrate more on taking notes.

Some of the findings from the interview were that the asset manager's main responsibility is to protect the client's interests and their role is first and foremost as *advisors* to the clients. They surveil the clients' stocks, customize yield and find an appropriate risk level for the clients. About 80 % of the transactions are made with stocks; the remaining 20% are made with other kinds of securities. They have a responsibility to always be available to their clients during trading hours. The asset managers have one computer and two screens running the same system on both screens. Usually, an asset manager has contact with about 50-70 persons and communication is mainly done over the telephone. The environment is cooperative and information is shared among the managers in the open and rather noisy office environment.

Three kinds of systems are important in their work: *information systems*, *trading systems*, and *depot systems*. The information system (usually SIX or Reuters) is their main tool, it provides the asset managers with continuous information about the market. The respondent mentioned that SIX was more commonly used, even though he thought Reuters was more user friendly, because Reuters cannot supply the information as fast as SIX. The default view has an overview of indices in a line chart, tables showing the current data for stocks, tables showing closures, tables showing currencies, a table with news headlines, and the clients' order books. They use no visualization tools and data is almost exclusively presented in tables. Their systems do provide visualizations, but the settings are too time-consuming and the visualizations are not updated in real-time and were therefore not useful.

The respondent pointed out that most important is to have a good *overview* and also *details* about the stocks. *Extreme values*, such as best and worst performance was very important. The information in between, is not that important, since no business decisions can be made from them. The most important information sources were considered to be: *news*, *stock price*, *closures*, *key figures* such as P/E ratio, yield, and *indices*, since it gives something to compare a stock price with and give a good overview. They focus very much on last trade price and key figures, always monitored in real-time, which to them mean updates every millisecond. Updates in the tables blink green or red depending on if the price has gone up or down. When they look at historical data it is mainly stock price that is of interest, sometimes several years back in time.

The mediating tool worked very nicely, the respondent talked about different visualizations and especially liked the heat map, because he thought it had an appealing and good overview for indices. He also liked candlestick charts because it is a useful combination of numbers and color and gives a good overview of the fluctuation in

stock price, but does not think that their system supports candlestick charts. He did not like pie charts at all. He also remarked that he thought it is necessary to combine color and form in visualizations because it would speed up the gathering of information. A feature that he thought would be useful was an alert feature since it would make their work more effective.

6.1.1.4 Meeting with internet trader

Even though the interview at the bank was fruitful, some questions came up when reviewing the results and we met with a person who trades on the Internet to clarify some issues. He also had insight into TIBCO Spotfire. From this meeting we got a clearer picture of the different roles on the stock market, such as stock brokers and analysts. We also got tips on where to look for further information about the stock market.

6.1.1.5 Second interview at a Swedish bank

We set up a meeting with an asset manager at another Swedish bank and we revised the interview guide from earlier interviews since some questions of the more general sort already had been clarified and thus did not need to be included. This time we wanted to focus on work flow and the tasks the asset managers perform in different situations and therefore included a set of scenarios which we wanted the respondent to comment on, so we could get a view of what activities would be triggered in these situations. The scenarios evolved around the following areas: *stock prices, financial ratios, news, indices and customer requests*.

The interview took place in a meeting room at the respondent's workplace in Göteborg. The respondent had an equal role to the respondent at the first Swedish bank, i.e. in charge of stock trade in the region. The work environment is very similar to the one described in the earlier interview, and the respondent pointed out the *importance of cooperation*, they work as a team. They base their decisions very much on *reports and analysis* which they get from fundamental analysts who receive large amounts of data about companies and the market from *news, annual reports, etc.* The analysts comment on news as soon as they are published, to give the asset managers an indication of how the market will react. At this point the asset managers must make decisions based more on intuition than analysis until the analyst gives a full report on the subject, which happens within a few hours of the news' arrival. The fundamental analysts are located in Stockholm. The respondent also says that nowadays the asset managers have access to very much information through the Internet, email, all their different tools, etc, but even if they get more information, it does not mean that they know more. It depends both on the way they get the information and that it is difficult to sift through the information. They do not have time to pay attention to details.

The respondent also confirmed the findings from the earlier interviews, such as which data and figures they are interested in, the systems they use in their work, that news is very important to them, and that hardly any visualizations were used. He also said that he looked very much at *volumes*.

Unfortunately the scenario-part of the interview did not work as well as we hoped. The respondent had problems talking about their work flow, and we think this is mainly because he would have needed to explain by showing us on the computer. It seems obvious that it would have been better to conduct observations than interviews,

but it was unfortunately not possible at any of the banks throughout the thesis work, due to confidentiality reasons at the banks. In fact, we could not even record the interviews on tape, so observing them in their work was even less possible.

6.1.2 Requirement analysis

We went through our findings so far and evaluated them to get an overview of the situation at this point.

6.1.2.1 Requirement specification

We defined the *user group* to be advisory and discretionary stock brokers and asset managers. These work as advisors to their customers and therefore need to be able to have access to thorough analysis of companies listed on the stock exchange before making decisions on how to invest. Analyses are mainly supplied to the stock brokers and the asset managers from the analysts. We considered the analysts to be a secondary user group, which we have not been able to explore since they are not located in Göteborg. A *user situation* where we found that a real-time analytics tool would be useful to the user group is the kind of situation described by our respondent at the second bank, i.e. the time span from when important announcements and news about for instance a company is published until the analyst give the asset managers a full report on how to act. This is a situation when they need to act on their own and make their own analyses of what decisions that should be made. At other times the users do not perform any analyses on their own; they act on recommendations given by the banks, i.e. the analysts' reports. We also created a *persona* to be able to refer to during the design process (see Appendix 3).

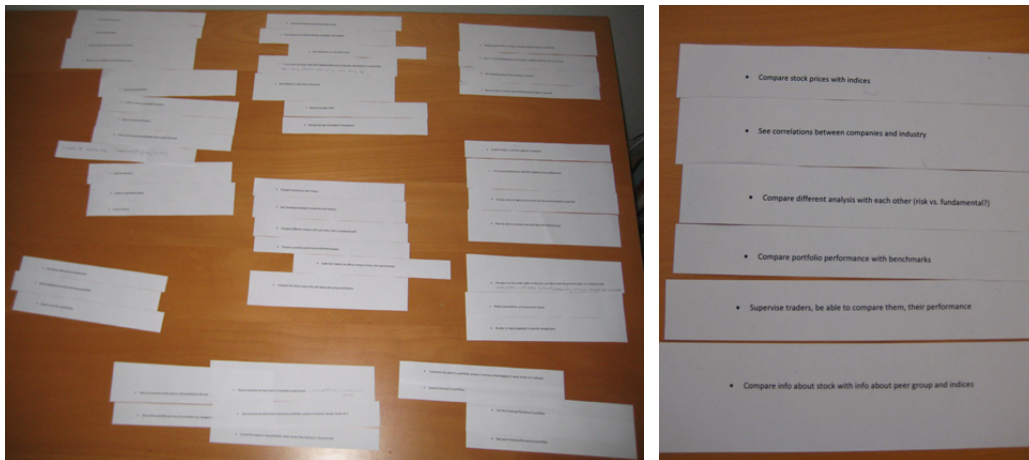


Figure 24. Performing the method affinity diagram.

We then listed all the *requirements* gathered so far and performed the affinity diagram method to categorize the requirements (see Figure 24). The discussions while organizing the requirements into groups gave us a better overview. After grouping them we looked at each category and tried to see if there was something we had missed to capture related to each category. In this way we were able to generate some additional requirements. The eight categories we grouped the requirements into were:

1. Data/information which the users need to have access to and monitor in real-time
2. Data/information over time
3. Overview of portfolio data
4. Details of the data in the portfolio
5. Searches in data
6. Comparisons of data
7. Work situation
8. General

The full requirement specification can be found in Appendix 4. We presented a description of the potential user group and the requirements gathered so far to the Company and after a consultation with them we decided that we should continue into the design phase based on the findings so far.

6.1.2.2 Scenarios

At this point we wanted to create scenarios which could be likely to happen in the specified user situation mentioned above. We had a brainstorming session where we came up with eight such scenarios. We then had one brainstorming session for each of these scenarios where we came up with possible activities associated with the particular scenario. We chose five scenarios to take further: *overview of portfolio performances*, *investigate different information sources*, *correlate stock price and index*, *compare volumes*, and *overview of a specific portfolio*, see Appendix 5 for the activities associated with them. These five scenarios were meant to be the base for our future prototype along with the requirements and we therefore wanted to test and get confirmed that this was what we should concentrate on in our future work. The requirements and scenarios were thereby evaluated.

6.1.2.3 Evaluation of requirements and scenarios

To evaluate the requirements and scenarios, we contacted *sales people* in the US at the company, since they have a lot of contact with actors on the financial market. We asked them to comment on the scenarios and requirements and answer questions related to them. The scenarios were also presented during a *conference* held at the Company where the management and the user experience team discussed the Company's future direction. Both of the evaluations gave the feedback that our requirements and scenarios were reasonable and that we should continue with this focus for our design.

At this time we also attended an *open lecture* on technical analysis, organized by Unga aktiesparare, held at the School of Business, Economics and Law at Göteborg University. This was mainly an opportunity to get additional information about the user group and their requirements, to be able to evaluate the previously gathered information. A portfolio manager at Volvo stiftelse-förvaltning held the lecture. Even though the lecture as a whole was not entirely relevant to the study we did get some of our ideas verified.

Earlier on, we had been in contact with the School of Business, Economics and Law at Göteborg University and found out that they have a *financial laboratory*. We now got the opportunity to visit the laboratory. Since we had not been able to observe asset managers in their work at any of the banks, due to classified information, this was a

great opportunity to observe a similar environment and see some of the software asset managers' use, and to confirm our earlier findings. We met the person in charge in the financial laboratory after a lecture on portfolio management, which we also attended. He demonstrated software that they had in the laboratory, mostly Reuters, and we asked questions mainly about which data is interesting, at which times, suitable visualization methods as well as our scenarios.

He confirmed that the scenarios and requirements we had established were reasonable as well as our earlier findings concerning for example the importance of news, overview of the data and real-time updates. He also saw a need for a new way to visualize data in real-time to get an overview and see correlations between stocks and other securities. He did point out that persons working at banks in Göteborg may have a slightly different work description than those at larger banks, for example in Stockholm and abroad. In Göteborg the work consist of a broader span of duties than does the work at larger banks where they are focused on specified tasks. The observation gave us some new ideas on how to design for this user group and situation.

6.1.3 Design of the layout and plot interaction

When entering the design stage we started to look at the overall layout and test the interaction with each kind of plot. We started out with low-fidelity prototyping to do this but soon realized that we needed to move on to high-fidelity prototyping to be able to explore the real-time aspect.

6.1.3.1 Low-fidelity prototyping

Based on the requirement specification and the scenarios, we started by defining what data and information that needed to be in the design. We categorized the information and could clearly distinguish three kinds of categories: data and information concerning *the market*, *a specific portfolio* or *a specific stock*. We decided that our design should contain three different views to display this information: *the market view*, *the portfolio view* and *the stock view*.

We then developed low-fidelity prototypes, paper sketches, with ideas for the layout and then chose two examples for each layout. Figure 25 shows the two stock view layouts.

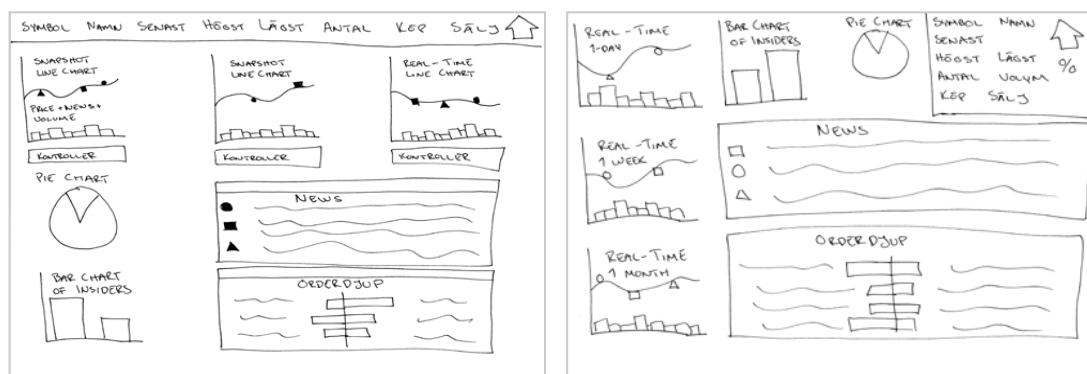


Figure 25. The first paper prototypes of the layout.

The next step was to see how our layouts would look on screen, especially in terms of if the information would actually fit in one screen in each view. This can be hard to

REALIZATION

determine when sketching on paper, since some parts in the interface can take up more space than expected. We made rough sketches in Adobe Photoshop of the layout, (see Figure 26).

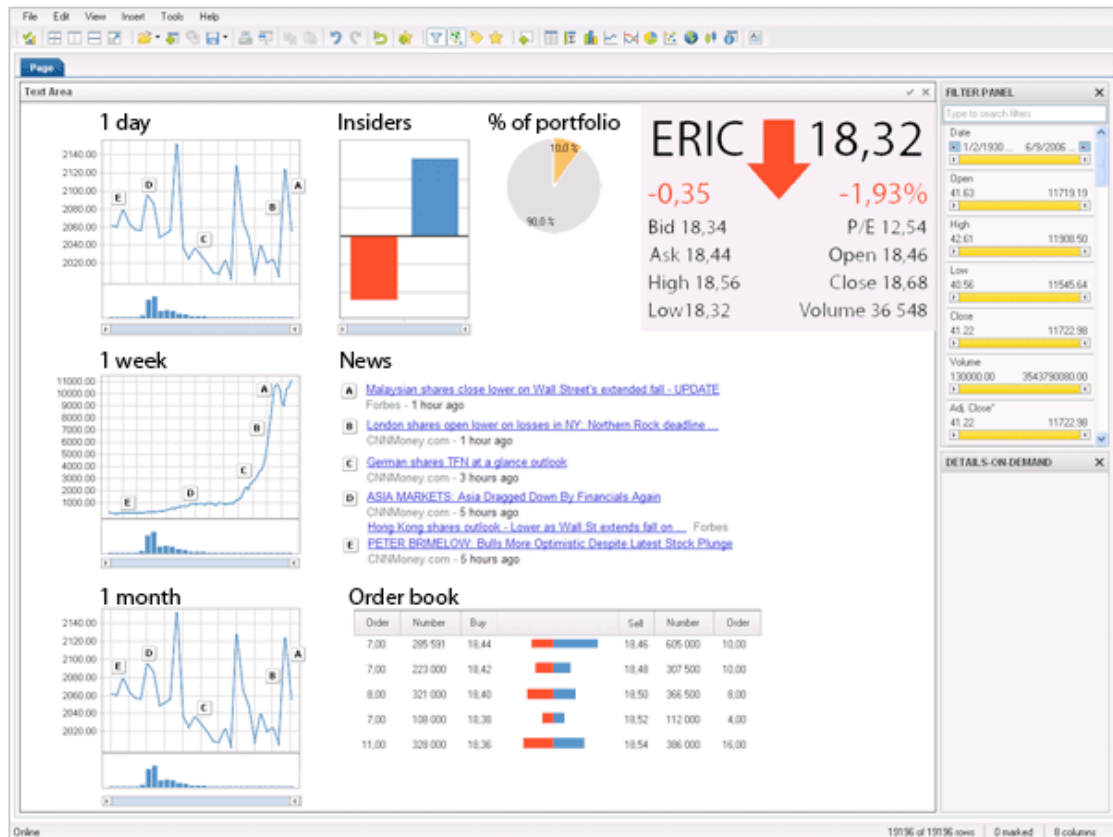


Figure 26. A sketch made in Photoshop of the layout in the stock view.

The market view and the portfolio view were a little more elaborate in their character than the stock view, since they needed to contain large amounts of hierarchical data. We studied literature and performed a brainstorming session to find techniques suitable for displaying data with a categorical and/or a hierarchical structure. The technique that seemed best suited was the *treemap*. It was difficult to find another technique alone that was as suitable so we decided to create an own solution of how the data could be displayed; a modification and combination of two techniques called *schema mapping* and *hyperbolic tree*. This alternative solution had some similar features to those of a *scatter plot* which inspired us to investigate that technique as well.

We created one treemap, one scatter plot, and one alternative design inspired by the schema map and the hyperbolic tree in Photoshop, see figure 27, 28 & 29.

REALIZATION

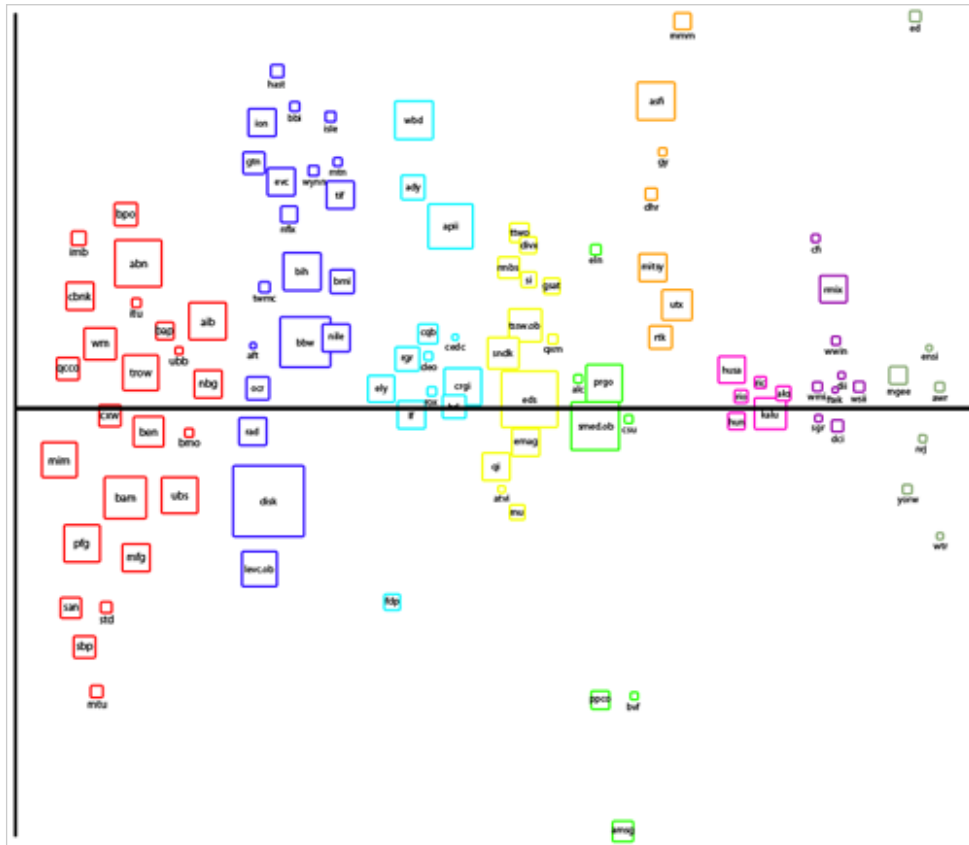


Figure 27. A scatter plot displaying hierarchical data.

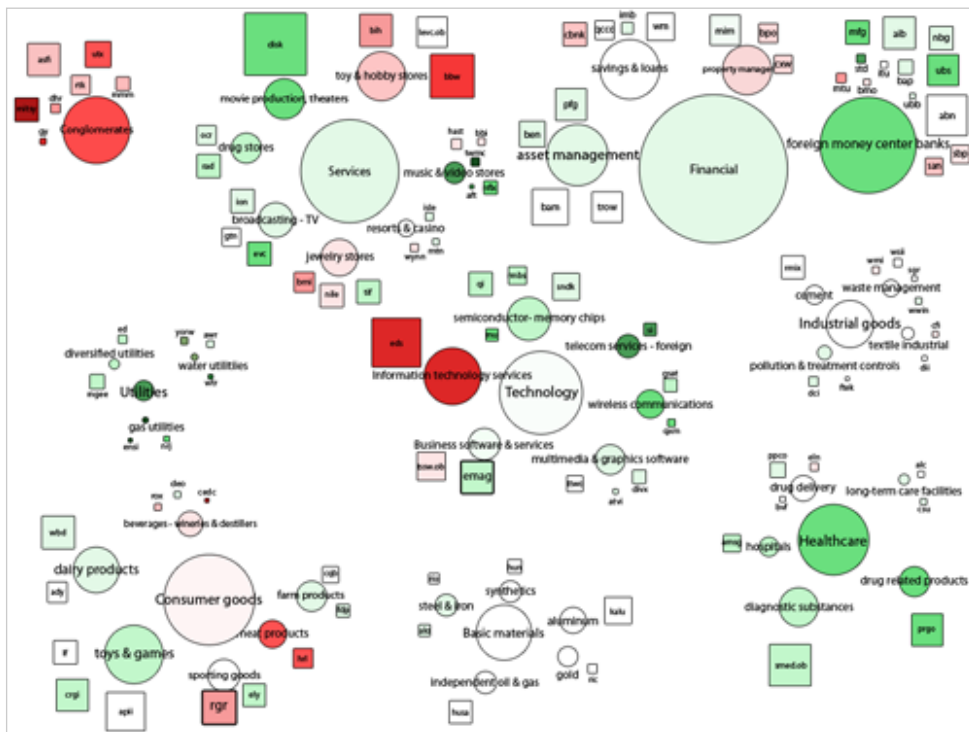


Figure 28. Our alternative design displaying hierarchical data.

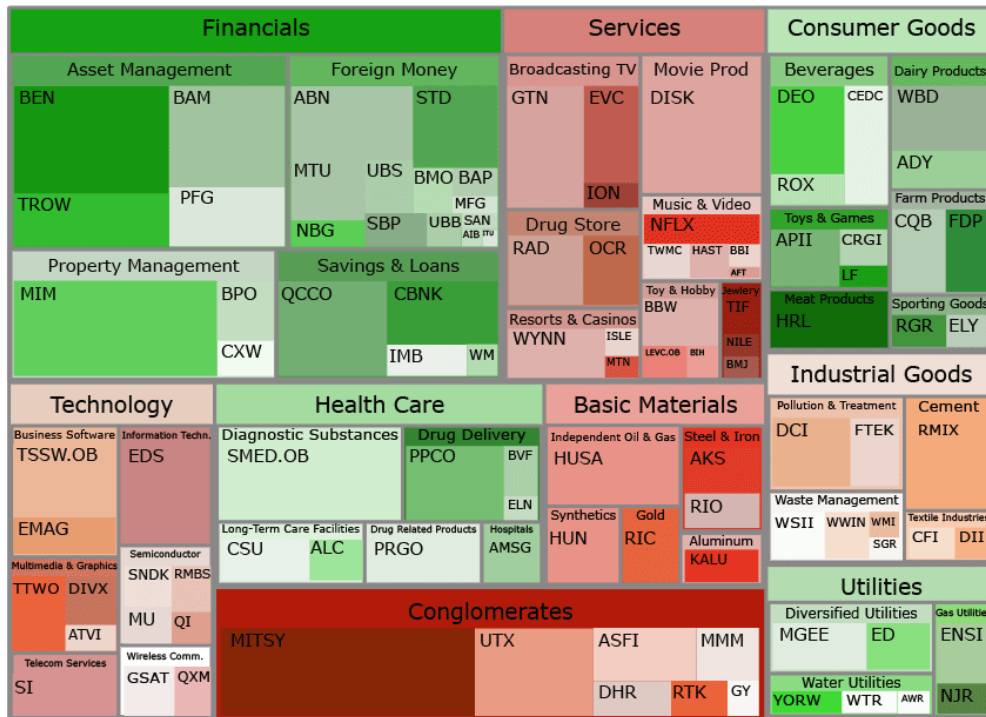


Figure 29. A treemap displaying hierarchical data.

After creating these designs we wanted to explore how the real-time aspect would affect each design and created each design in Microsoft Office PowerPoint because we wanted to, in a simple way, create a simulation of the real-time aspect in the plots. Doing this with the treemap and our alternative design proved to be easy, since real-time was represented by color and we therefore only needed to make changes to the colors in the plots. However, when it came to the scatter plot, real-time updates were represented by movement and PowerPoint was not the most suitable tool to use for animation. We therefore left the low-fidelity phase two weeks earlier than planned to start high-fidelity prototyping by developing and testing the treemap and the scatter plot using Adobe Flash. We also decided to abandon our alternative design at this stage, since several disadvantages were identified. The design did not take advantage of the available space, and it was therefore very difficult to display large amounts of data with the design. It was also rather difficult to interpret the visualization.

6.1.3.2 High-fidelity prototyping

We started out by developing a line chart and a bar chart in real-time using Adobe Flash since we believed that it would be easier to do these *plots in real-time* than it would be to develop the treemap and the scatter plot in real-time. We wanted to start out with easier parts since none of us had used Adobe Flash extensively before so we needed to learn the software. After developing the line and bar chart we also developed the scatter plot and treemap in real-time (see Figure 30 & 31). While making the plots in real-time we discovered that they would not move in the fast speed we thought they would. It rather appeared to be a problem to distinguish which plots were in real-time and which were static.

REALIZATION

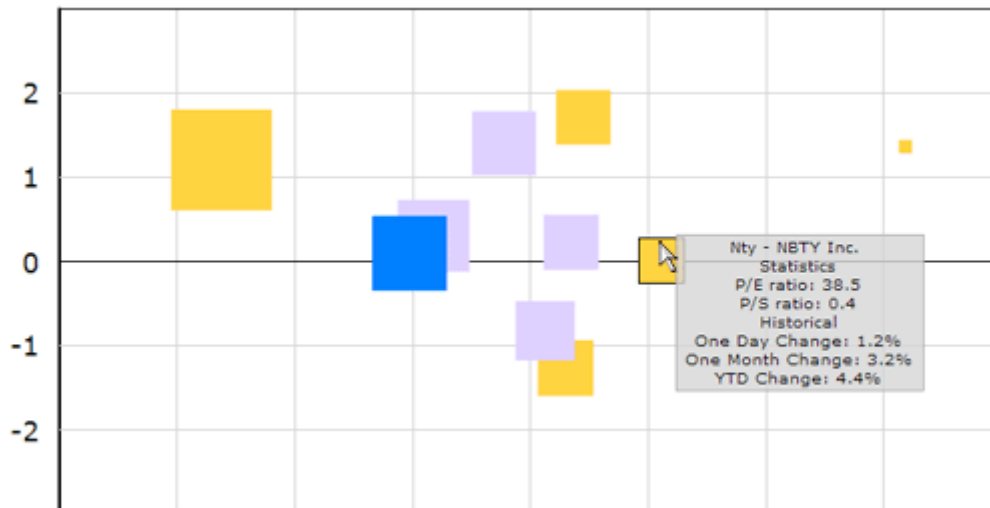


Figure 30. An early Flash animation of a scatter plot in real-time. One stock has been selected, which is shown by the blue color. The mouse cursor hovers over the stock "Nty", which makes the highlight frame appear around the stock and the tooltip is displayed.

The next step we took was to test how the *interaction* with the real-time plots was affected. Since we had discovered that the plots in real-time would not move rapidly we suspected that the current interaction would work rather well even for real-time data. We used the interaction from TIBCO Spotfire and applied it to our real-time plots in Flash to see if the interaction would work. We added interaction for highlighting on mouse-over, tooltip on mouse-over, left-click, ctrl-click, marking when selected, and drag rectangle to select items (see Figure 30 & 31).

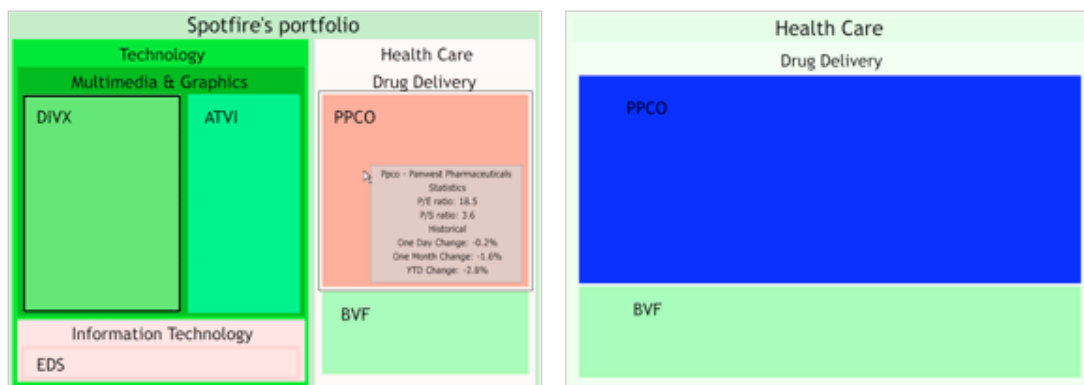


Figure 31. An early Flash animation of a treemap. The left image shows a portfolio where the stock DIVX stock price is being updated which can be seen from the update frame appearing. The pointer hovers over the stock PPCO causing the highlight frame to appear as well as the tooltip showing information about the stock. The right image shows a drill down on the healthcare sector where the stock PPCO has been selected.

6.1.4 Evaluation

We suspected that the interaction used for static data would also work for real-time data and tested the different kinds of interaction we had added by going through them one by one in each plot. Our suspicions were correct, the current interaction worked

well in the real-time plots, since the movements in the plots were so subtle. After making these discoveries we decided that we needed to see how these discoveries would affect our layout. We therefore went back to prototyping the layout, starting our second iteration.

6.2 Second iteration

We set out to review if our design conformed to the discoveries we had made concerning the interaction and real-time aspect of the plots; if not, a redesign could be required. The second iteration includes three of the four stages of Preece et al. (2007) iterative model, since we went directly to the second stage in the model, the (re)design stage. It also involves building an interactive prototype and the evaluation stage in the model was also revisited in this iteration.

6.2.1 (Re)design of the layout, work flow and functions

We started out with looking at and developing the layout of our design further. After this we looked at the workflow, how everything should be connected and how the specific functions should work. Within this (re)design phase we used both low-fidelity prototyping and high-fidelity prototyping.

6.2.1.1 Low-fidelity prototyping

We needed to distinguish real-time data from historic data and since we had gotten some ideas on how to develop our layouts further during our work we decided to once again draw the three views on paper and developed two alternatives per view (see Figure 32).

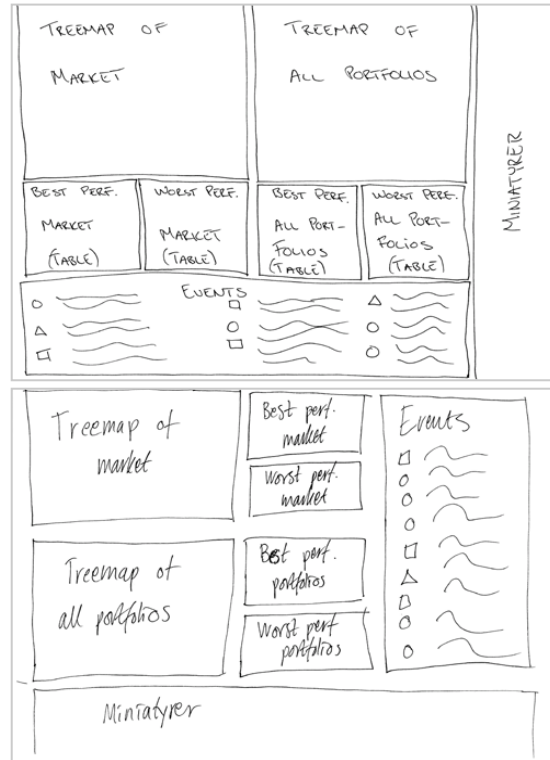


Figure 32. The second paper prototypes of the layout.

We then started working on how the views should be connected. Possible solutions for moving from one view to another were discussed, and a scenario containing the solutions we considered to be the most intuitive was created. We intended to test the different kinds of solutions in the upcoming expert evaluation. We also focused on which functions to include, based on the scenarios. The four functions we saw most important were: setting *alerts*, marking *minimum and maximum* values in the visualizations (extreme values), making *duplicates* of current plots in the visualization to be able to make comparisons and to be able to *go back and forth in time* to distinguish trends and patterns.

6.2.1.2 High-fidelity prototyping

It was now time to develop a more interactive prototype of our design so that we could test it and get feedback. We distinguished what was most important to test and decided that it was: *the different layouts, the solutions for the workflow, how the interaction works, and the design of the different functions*. We started to develop a scenario using Flash containing all of these different parts (see Figure 33). The prototype we developed was not fully interactive in the way that you could click on all different parts in the interface. Only parts of the interface were implemented in the prototype. With the help of the prototype you could however get an idea of how, for example, the work flow and the different features should look and work. The scenario was then evaluated to see what improvements that were needed to be done to our design.

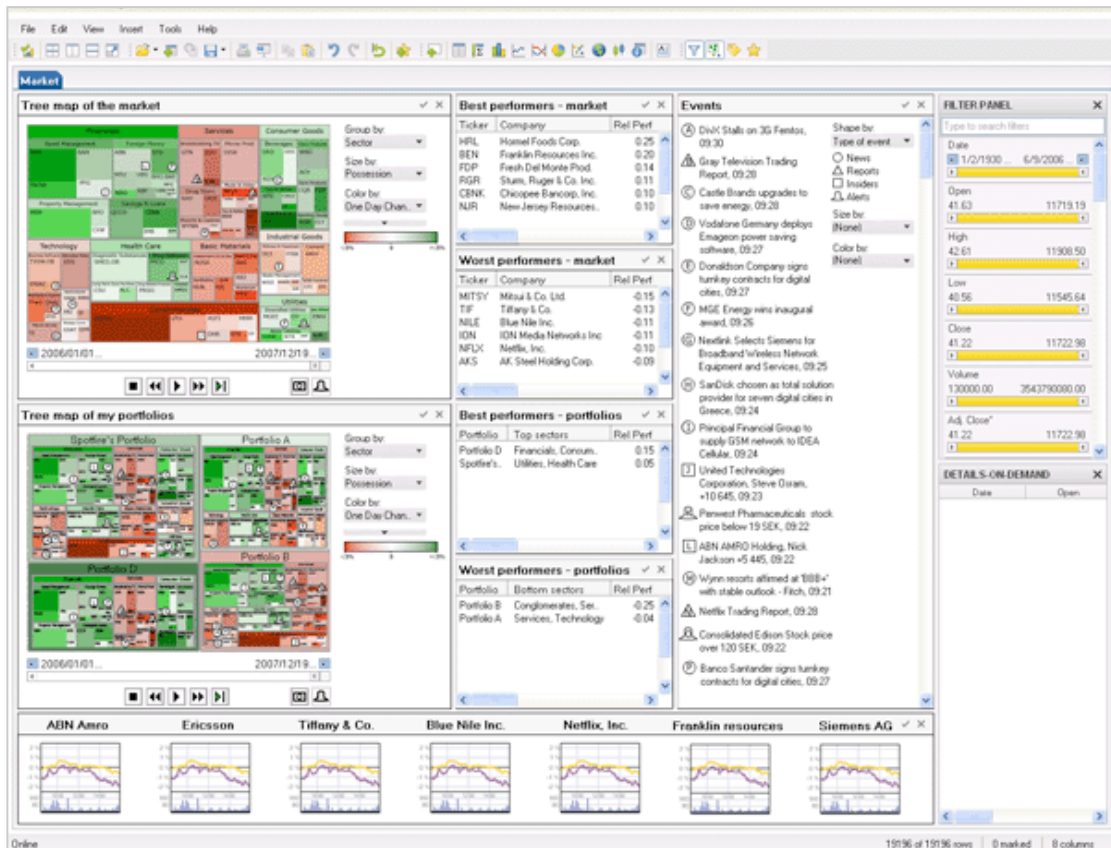


Figure 33. The market view from the first Flash prototype.

6.2.2 Expert evaluation

We conducted a formative expert evaluation to choose what layout our design should have, decide how the interaction should be made, and how the different functions should work. It was formative, since it was performed during the development process to find out what improvements and further developments that could be made to our design. It was an expert evaluation both because users were difficult to find, but also because we thought it would be very hard to perform an evaluation with the users since we did not have a fully interactive prototype, with which the users could test the prototype themselves.

The expert evaluation was performed with eight employees at the Company, five of them worked in the user experience team, and three were developers. Two of the attendees also had knowledge of the stock market from trading on the internet regularly. We wanted this mix of persons to get feedback from many different directions. The evaluation performed was informal in its nature since we wanted to be able to have a discussion with the participants. We went through the scenario step by step, discussing and evaluating each part of the design from an esthetic and a usability perspective as well as a more general perspective.

The discussions at the expert evaluation revolved around many areas, among others drill down, which is an important feature in TIBCO Spotfire today. The attendees agreed that double-click works for drill down and when drilling down in a treemap, miniatures should appear for each level to let the user step upwards at all times. It was suggested that *navigation* between views could be done by right-click, by marking an item and clicking a function key, or by using miniatures in the miniature field. Time controls and showing movement was other subjects we covered. It was suggested that if you move in time, the *last part of the graph could be kept in real-time*. History could be shown in scatter plots and bar charts using shadow or up and down arrows. Perhaps treemaps and bar charts do not necessarily need to show history since the plots are not suited for it. For the alert feature, some kind of animation was suggested; when the alert is triggered it could appear in an event window. They thought the minimum/maximum feature should not only be available to individual stocks but for sectors and other groups as well. A discussion revolved around the possibility to extend the interface to more than one screen since the users have at least two screens, but this idea was abandoned since it was likely that their information system must always be visible. Another issue concerning space was to possibly *hide the filter panel and time controls* to save space. The participants also saw the possibility that filtering could work for each plot separately to be able to compare different selections and you should then be able to search and mark on the plot instead of removing objects.

The expert evaluation gave us new ideas, made us decide on the layout we should continue with, what changes that needed to be done, how the interaction within plots and between views could work, and also how the different functions could be improved. With this knowledge we entered the third iteration in our process.

6.3 Third iteration

This iteration consisted of the three final stages of the iterative model; the (re)design stage, building an interactive prototype and evaluating the prototype (Preece et al., 2007). We brought all the pieces of our design together in this iteration; such as the layout, the plots, the functions, and the workflow; and built our final prototype. This prototype was then evaluated through a user test and an expert evaluation.

6.3.1 (Re)design of final prototype

After the expert evaluation we once again reviewed our concept and design to see if any redesign was needed. We then developed our final prototype starting with making low-fidelity prototypes, which were then realized into a high-fidelity prototype.

6.3.1.1 Low-fidelity prototyping

Based on the feedback from the expert evaluation we designed the three final layouts for the prototype on paper (see Figure 34, 35 & 36).

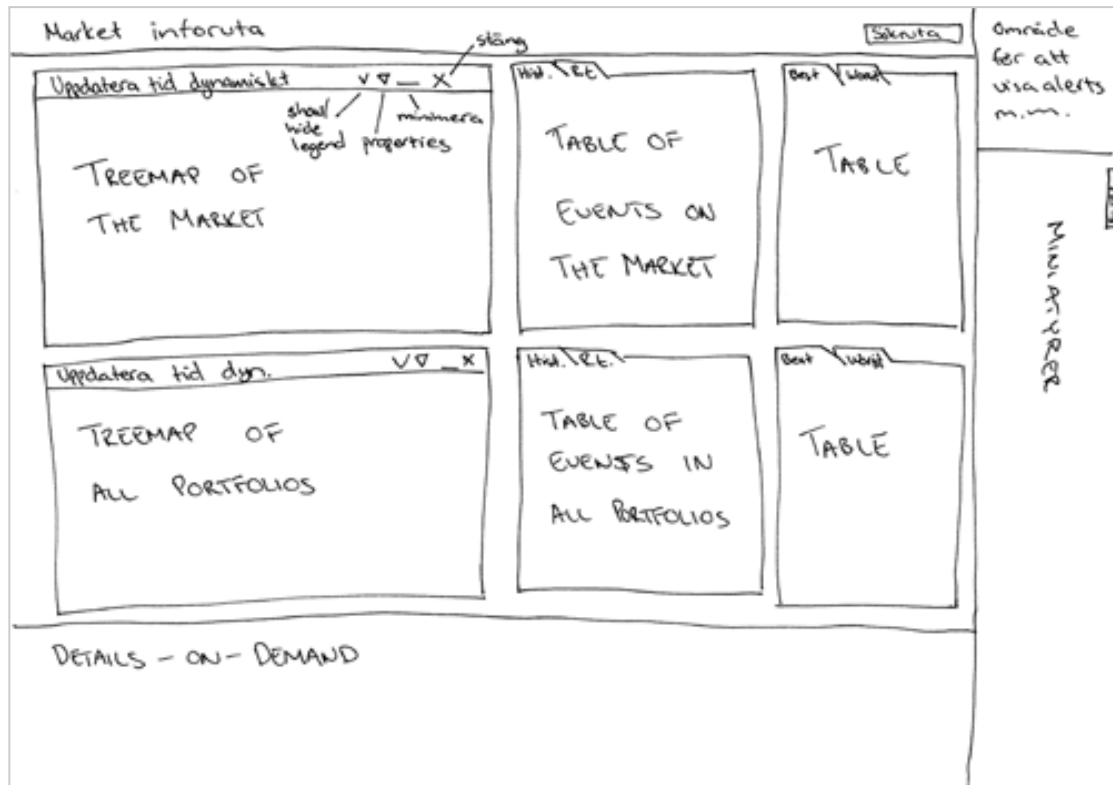


Figure 34. Paper prototype of the final layout of the market view in the prototype.

REALIZATION



Figure 35. Paper prototype of the final layout of the portfolio view in the prototype.



Figure 36. Paper prototype of the final layout of the stock view in the prototype.

6.3.1.2 High-fidelity prototyping

Considering the concept was very large and the limited time at our disposal we created a priority list of what we thought was most important to test later on. The list contained 36 points, which was far too many and we estimated that we would have time to develop about the first 15 points on the list in the prototype and that this would allow us to test the most vital parts of our concept. We considered some of the most important parts to be: *time controls, making duplicates, drill-down, interaction in plots* (marking, highlighting etc.), *navigation between views, setting an alert and noticing a triggered alert, the minimum/maximum feature, events and adding tickers to the line graph*. We also wanted to see how and if movement was noticed as well as if the users could distinguish between real-time and historical data. See Appendix 6 for the entire priority list.

To know which parts to develop in the prototype we created a test scenario that the users should follow during the tests later on. This way we knew exactly which parts of the prototype the user should be able to interact with. The test scenario contains specific tasks from the priority list which the user should perform with the help of the prototype (see Appendix 7).

We started with creating the foundation of one view and used that view to create the others, since they all should have a similar layout as base. We then started at the top of the prioritizing list and worked our way down. We reached point 15, which was our goal, on the prioritizing list before it was time to evaluate the prototype. The prototype is described in detail in section 7.1.

6.3.2 User test and expert evaluation

The purpose of the final evaluation was to test the layout and features in the prototype, if the right information was displayed, and if the interaction with the plots and between the views worked. Since the demonstrator we had developed was not a finished product but a prototype, the evaluation was formative.

We chose to perform the evaluation with both users and experts. The main reason for this was that it had proven to be very hard to get hold of users in our user group, since they are very busy persons. Preece et al. (2007) say that many usability professionals, among these Dumas and Redish (1999), recommend between 5 and 12 respondents when performing an evaluation. This was an unreachable number for us; we could only get hold of three users. We therefore chose to perform an expert evaluation with three attendees, i.e. a total number of six respondents. The three users were asset managers working at two large Swedish banks. The three experts all work at the Company; two of them as usability experts, one as a developer.

The user test was a regular *usability test*, i.e. the users performed predefined tasks while being observed, and the think-aloud technique was used. Usually you would record the test sessions on video, but this was not possible due to secrecy at the banks, so we had to rely on written notes. Since analyzing video recordings is very time-consuming and note taking worked well, we decided to settle for this in the later expert evaluations as well. The tests were performed in controlled environments, which is common when performing usability tests. The tests were performed at each user's work place, since they did not have the time to come to us to perform the test. We thereby brought and set up an environment containing a 21 inch flat screen, a

keyboard and mouse, all connected to a laptop. The reason why we wanted the users to perform the test on a regular sized computer screen instead of on a laptop was to make the situation as realistic as possible; asset managers are used to working on large screens and seldom use laptops in their work.

The evaluation performed with the experts was a modified *cognitive walkthrough*, and the setup was similar to the user tests. The experts got to perform a number of predefined tasks with the help of our prototype. They were also asked to think-aloud when performing the tasks. The main reason why we chose not to perform a more traditional expert evaluation with the help of a set of heuristics is that we thought that this setup would be more rewarding for us. Our prototype was not fully interactive, which made it hard for the experts to test the prototype by themselves. We also thought it would be rewarding to be able to have a discussion with the experts while they tested the prototype.

Before performing the first test with a user we performed a *pilot study* to test our setup for the evaluation. The respondent was an employee at the Company with knowledge of the Company's products, interface design, and the stock market, since he trades with stocks on the Internet. The pilot study made us aware of several different things. We noticed that the test took longer time than we had thought it would. We consequently asked the users if they could spare more time to perform the test, and surprisingly enough this was not a problem. We also noticed that it was better if we described the tasks orally, instead of letting the respondent read from a paper. It was difficult to concentrate on the tasks when having to read at the same time, so in the real study the users got a paper to look at if they wanted but we always explained the task to them as well.

The mix of users and experts was beneficial since they had knowledge about different domains and we could discuss different issues with them. Some general thoughts from the expert evaluators were that there could be much to be gained from *toning down the interface* color-wise, i.e. lighter grey color overall and removing the colors from the event icons. The event icons could also be smaller and less dominant, and the letters on the icon were confusing. The users thought the connection between events in the plots and the list was very useful. One respondent wanted to be able to see all the events during one day. General conclusions of which events to stress in the interface were impossible to make since the response varied per individual. We found throughout the tests that the respondents had *problems finding the time controls* and they expressed a wish to make them visible at all times. One user said it was much too time-consuming as it was now. The most preferred navigation for going back in time was dragging the line chart directly. All the users said it was an important feature, to be able go back in time and make comparisons. In general, the duplicate and minimum/maximum buttons were also difficult to find for most of the respondents, but once the minimum/maximum dialogue was opened none of the respondents had any problems entering the values. Most of them did not however notice the symbols that were displayed in the treemap after entering the values. The users thought this was very valuable though, since they look at extreme values a lot, and they would use it at all times. Some respondents did not know what to expect from the duplicate button. The users thought it was good however, that it happened instantly and without having to confirm the action. There was a suggestion to *put the buttons in a toolbar* to make them easier to find. The alert function was difficult mostly for the users to find,

while the experts found it rather quickly. None of the respondents had any problems entering values when the dialogue was opened. However, when it comes to the alert being triggered, only one of the six respondents noticed it at all. The users said that they would probably use this function since it was easy to use, compared to the one they currently had in their system.

The treemap was understood by all of the respondents after a short explanation prior to conducting the test. The users expressed a wish to be able to sort the treemap in different ways. The scatter plot was the least successful visualization. None of the respondents fully understood it or the history function for the visualization, but one expert suggested to *add sparklines to give extra information* about history. All respondents understood the line chart, but one user would like to see candlesticks or bar charts instead. All respondents understood that the tabs would open another view.

The users said they rarely wanted to exclude anything from the screen, instead one of them even said he still kept adding information to his screen, and therefore they meant that with less information the system would not be useful to them. They did not want to filter the entire views, but rather one plot at a time. They expressed a wish to connect a system like ours with the bank's own database of information, with for example reports, as well as for example Reuters to get the functionality of both of these systems. They believed that such a system would have a great potential since it would make their work more efficient.

In general, the experts had fewer problems finding controls in the interface, which is not surprising since they are familiar with TIBCO Spotfire as well as interface design in general. Another general finding was that the three users all had different wishes about what to look at and what information should be presented and in which way. How an asset manager chooses to set up the information is therefore varying and it is important to provide alternatives.

6.4 Fourth iteration

After performing the evaluation of our final prototype we went back to looking at the whole concept for the design. This included parts we had not been able to develop in our prototype as well as parts in the prototype that needed to be developed further.

6.4.1 (Re)design of concept

We started by discussing the result from the evaluation, section by section to identify the most important findings. In this way we could identify the parts we needed to focus on and develop further in the concept.

6.4.1.1 Low-fidelity prototyping

We started out by sketching the different parts of the concept, which we wanted to develop further, on paper. We created some different solutions of each part of the concept before deciding on the final design, see figure 37 for an example. After doing this we developed more advanced prototypes using Photoshop. The prototypes were however still low-fidelity prototypes since they lack in functionality and interactivity. These prototypes can be found in section 7.2 where the final concept is described and motivated.

REALIZATION

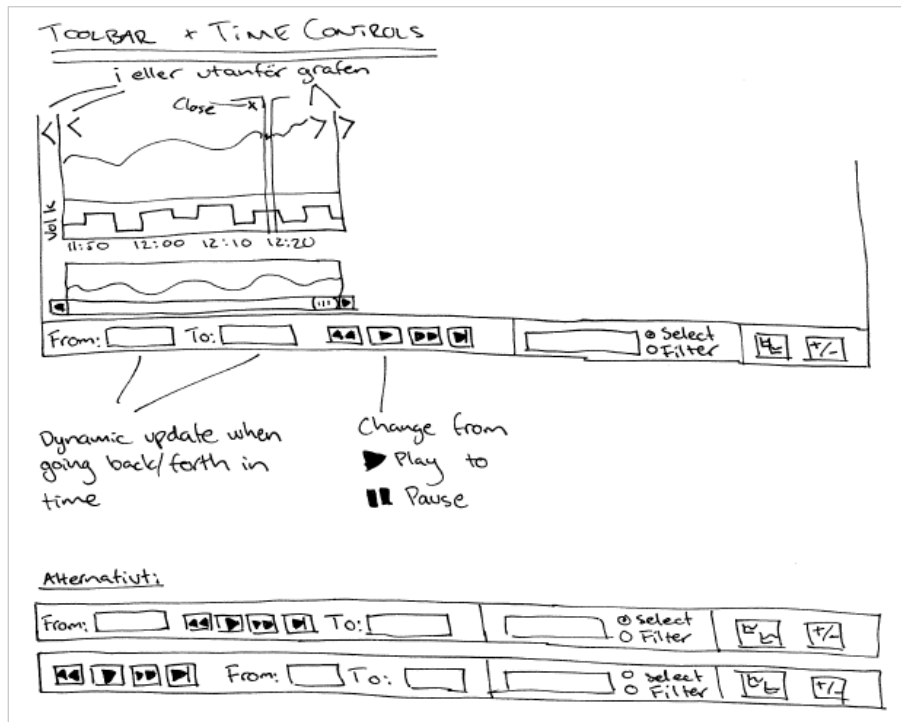


Figure 37. Paper prototype of the design of the toolbar in the concept.

7 Result

The results our work has led up to will be described here. A prototype, which was developed according to the gathered requirements, will be explained. We will also describe and motivate the whole design concept, which we have developed throughout the work.

7.1 Demonstrator

In this section a description of the demonstrator that was developed in Adobe Flash will be given. The overall layout of the design in the demonstrator will initially be explained. The three different views in the demonstrator: *the market view*, *the portfolio view* and *the stock view*, will then be described. Each display medium used in the concept is also described. Finally the features in the demonstrator, *alerts*, *duplicate*, *minimum/maximum* and *time controls*, are described.

7.1.1 Layout

The demonstrator consists of three main views; *the market view*, *the portfolio view* and *the stock view*. There is also a framework which is the same for all these three views. We will discuss the views separately in section 7.1.1.1-7.1.1.3 and concentrate on the overall *layout and framework* in this section. To get an overview, we will look at the market view and go through the layout of the interface, see figure 38. The major part of the interface consists of the plots and data for the specific views, and the layout within each view varies somewhat, which we shall see later. The *visualization area* is the dominant area in the interface (1). The upper section of the interface contains *menus and a toolbar* (2) which have not been changed from the TIBCO Spotfire software and since there is no functionality implemented in them, we will not discuss them further. Directly below the toolbar the three *tabs* are located (3). Clicking any of the tabs will open the selected view. Below the tabs is an *information area* containing unique information about each view (4). On the right hand side of the information area there is a *search field* (5) which has not been implemented but was intended to allow searches for stocks, portfolios etc.

In the upper right corner of the design, the *alert area* which we will explain further below, is located (6). Along the right hand side of the interface is an area with a number of *miniature plots* (7). The purpose with these miniatures is that the user should be able to keep track of specific data that is of interest at the time, but may not be visible in the current view. In the demonstrator many different kinds of plots have been chosen for the miniature plot area and the idea is that the user should be able to keep track of any kind of plot. However, the miniature plots have not been implemented in the demonstrator, which means that it is not possible to interact with the miniatures, to add new miniatures, or to move or delete the ones that are currently visible etc. Even if the miniature plots have not been implemented fully, there is movement in the plots, as they are synced and updated along with the other plots in the demonstrator. So, for example, when a specific stock is updated in for example a line chart, that same stock will also update in the miniature area.

RESULT

The *filter panel* has been hidden and its presence is visible through a tab located on the right hand side of the miniature area (8). There is also a tab labeled *search* above the filter tab. The search tab was a part of an early idea that the target users probably would not want to exclude data from the view, but rather locate specific data in a rather data intense view. Exploring how this kind of search could be done was a much too broad area to dig into and both tabs are therefore left unimplemented.

The *details-on-demand area* (9) which can be found at the bottom of the interface, below the plot area, is also left unimplemented, because it would have been very time-consuming to add fake data to the details-on-demand field and in addition to this make it dynamic. To change from one view to another you can either click the tab to open the wanted tab, or you can right-click. The latter will open a menu where you can choose which view you would like to go to.

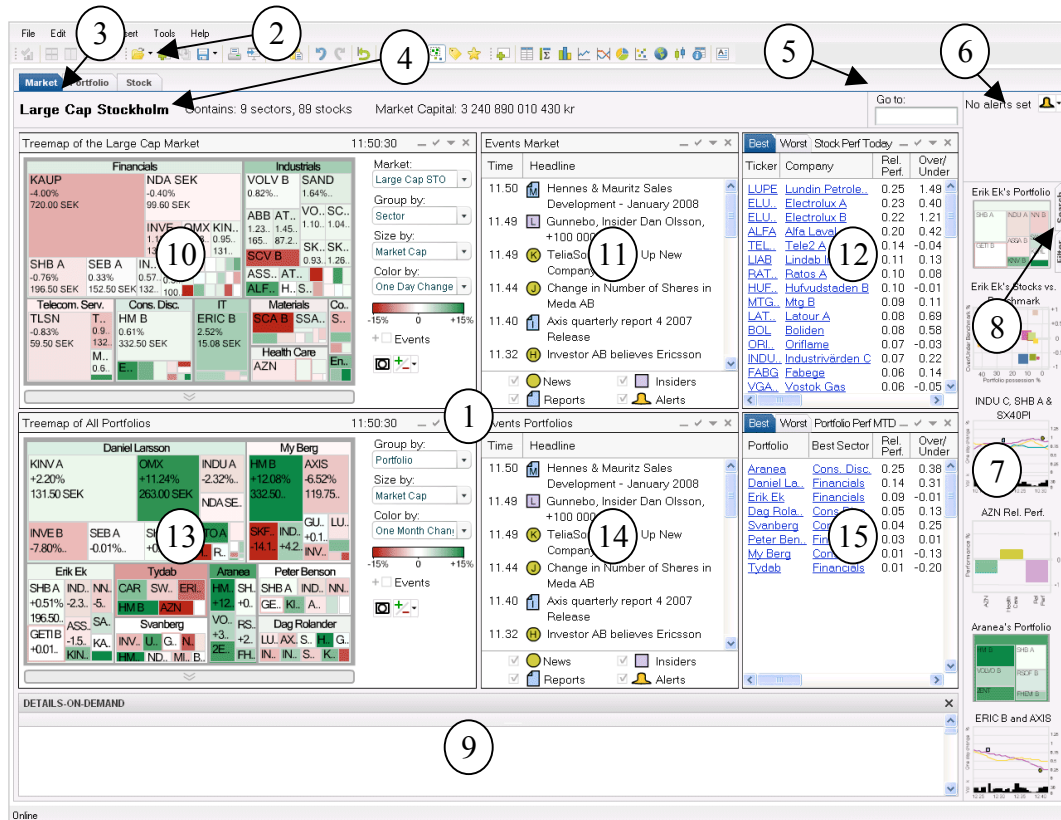


Figure 38. The market view in the demonstrator.

7.1.1.1 Market view

In the demonstrator the market view has six plots (see figure 38). The upper left plot is a treemap showing the sectors and stocks at the Stockholm Large Cap Market (10). Treemaps will be further described in 7.1.2.1. The two plots to the right of the treemap contain information connected to the Large Cap. The plot next to the treemap is a table showing events that are related to the Large Cap (11). The events plot will be described in 7.1.2.5. The rightmost of the three upper plots is a table showing the best and worst performers on the Large Cap market during the current day (12). The Best/Worst plot will be described in 7.1.2.6.

The lower part of the market view contains plots related to the portfolios managed by the user. The layout of the lower part of the view is the same as the upper. The leftmost plot is a treemap with an overview of all the portfolios managed by the user (13). To the right of the treemap is the events plot (14), this time showing events related to the portfolios instead of the Large Cap Market. The rightmost plot displays information about the best/worst performers the last month among the portfolios (15).

7.1.1.2 Portfolio view

The portfolio view contains plots showing information about one specific portfolio, in the demo this view shows My Berg’s portfolio, see figure 39. The upper plot row has the same layout as in the market view. That is, the leftmost plot is a treemap with an overview, in this case the overview shows the sectors and stocks in My Berg’s portfolio (1). The upper middle plot shows, in a table, events related to the stocks in My Berg’s portfolio (2). The plot to the right is a table, as in the market view, and shows the Best/Worst performers among the stocks in the portfolio (3).

The lower plot row contains five plots, two with data related to the portfolio and two providing the user with an overview of the market and portfolios. The plot in the lower left part of the portfolio view is a scatter plot showing the stocks in the portfolio compared to a benchmark (4), the index OMXSPI. Next to the scatter plot is a bar chart showing the portfolio performance compared to benchmark in the last month (5). The scatter plot and the bar chart will be described in more detail in 7.1.2.3-7.1.2.4. The remaining two plots in the portfolio view are two treemaps. The treemap next to the bar chart is an overview showing all portfolios’ performance the last month (6), where My Berg’s portfolio is highlighted by a blue frame. The plot to the right is an overview of the market showing the stocks performance during the day (7).

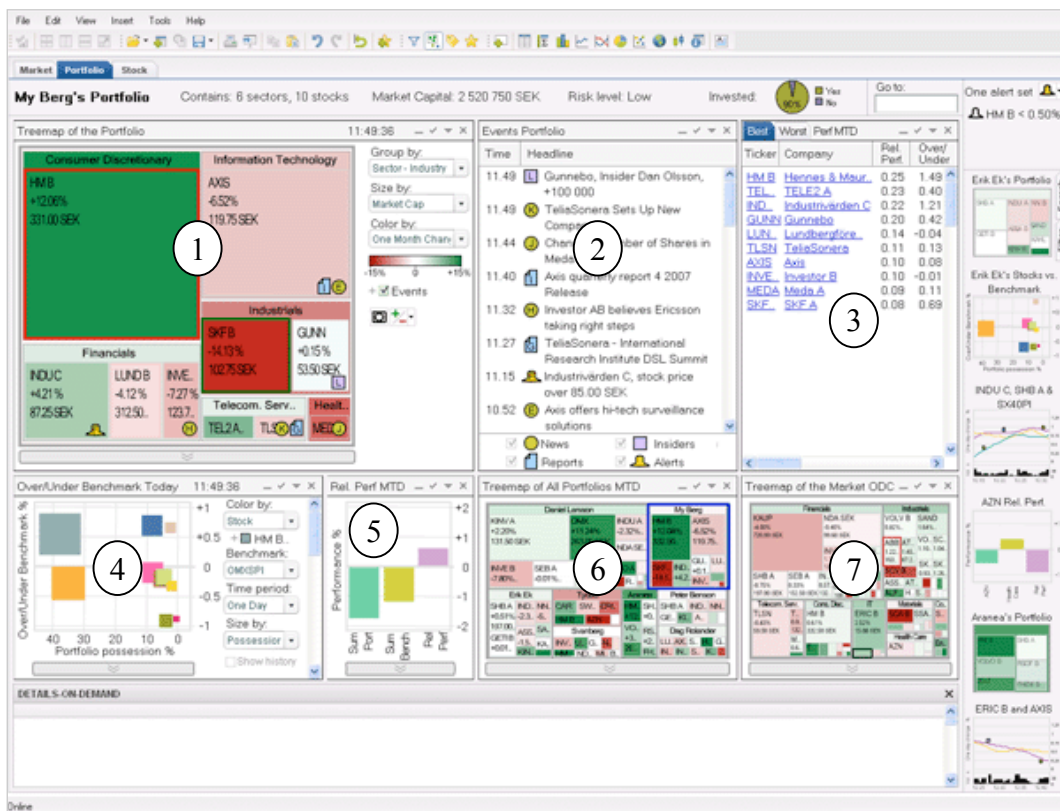


Figure 39. The portfolio view in the demonstrator.

RESULT

7.1.1.3 Stock view

The stock view contains information about one specific stock, in the demonstrator Hennes & Mauritz B, HM B stock. It is similar to the portfolio view in that it has the same layout, see figure 40. The upper left plot in the stock view is a line chart showing one day change for HM B, purple line, compared to the OMXS index, yellow line (1). The time period selected for the line chart is 30 minutes. Volume for the stock is also displayed. Line charts will be further described in 7.1.2.2. The two other plots in the upper row are tables. The middle one shows events related to the stock and the index (2), and the other table shows best/worst performers for the sector that HM B belongs to over the last month (3).

The leftmost plot in the lower plot row is a scatter plot showing the stocks in the consumer discretionary sector, which HM B belongs to, relative the index OMXSPI (4). The plot to the right of the scatter plot is a bar chart showing the HM B stock's performance the last month compared to the consumer discretionary sector as well as the stock's performance relative that sector (5). The plot to the right is a treemap showing all the user's portfolios with the HM B stock highlighted by a surrounding blue frame (6). The rightmost plot in the stock view is a treemap showing performance of the stocks in the Large Cap Market over one day (7). The HM B stock is highlighted by a blue frame in this treemap as well.

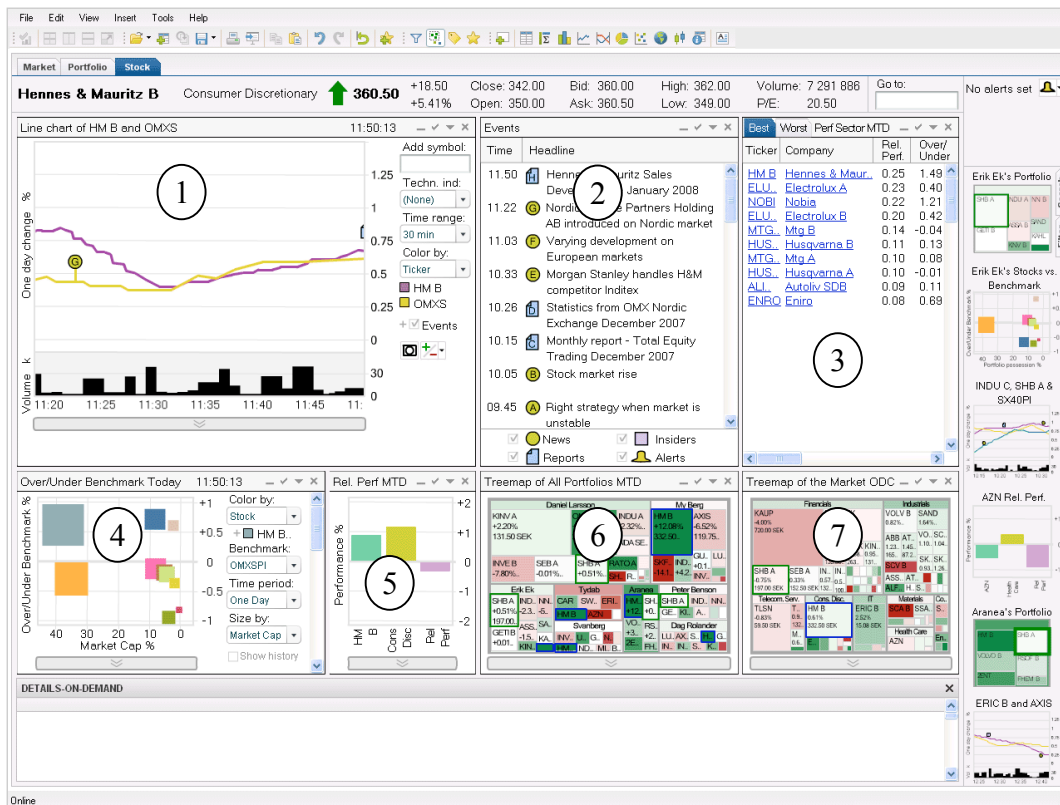


Figure 40. The stock view in the demonstrator.

7.1.2 Display mediums

The demonstrator primarily contains five different types of display mediums: *treemaps*, *line charts*, *scatter plots*, *bar charts*, and *tables*. Each of these display

mediums is used for visualizing different kinds of information on the work surface in the demonstrator. Each display medium will be described further in this section.

7.1.2.1 Treemaps

The demonstrator contains a number of treemaps, showing different hierarchical data, but they all work the same way. Figure 41 shows a treemap as it can be seen in the portfolio view. It shows all the stocks that the specific portfolio contains.

The *legend* (1) in the treemap provides the user with information about what is shown in the treemap. In the demonstrator, the legend dropdowns have not been implemented with functionality, which means that it is not possible to change what is shown in the treemap. The legend tells us that the treemap is grouped by Sector. This means that the treemap is divided into rectangles symbolizing sectors. The treemap in figure 41 contains six sectors: Consumer discretionary, Information technology, Financials, Industrials, Telecom, and Healthcare. Each of these sector rectangles contain the stocks in the portfolio that belong to the specific sector. The stocks INDU C, LUND B and INVE B belong to the financial sector which can be seen from the fact that these stock rectangles are placed inside the rectangle that represents the financial sector (2).

The legend also tells us that the treemap is sized by Market Cap, which means that the sizes of the rectangles depend on the market cap. It is evident that the HM B stock has a larger market cap than for example the SKF B stock. The legend also tells us that the treemap is colored by One Month Change. This means that the rectangles' colors represent the change in price for the stocks, and the nuance of the color is decided by how much the price has gone up or down during that month. In the legend there is also a color legend (3) showing the full color range of the treemap. In the demonstrator the colors in the treemap range from dark green via white to dark red. The darkest red symbolizes a bad trend, i.e. in the last month the stock price has gone down. The darkest green nuance symbolizes a good trend, i.e. in the last month the price for the stock has gone up. We can see that the HM B stock price has increased during the last month. This can be seen both by the dark green color, but also in the stock rectangle, where the exact change measured in percent is shown (i.e. +12,08%). In the same way, it can be seen that the SKF B stock has had a negative trend over the last month. We can also see that the stock price for GUNN has not changed very much over the last month by its white color.

Figure 41 shows how updates in the treemaps are visualized in the demonstrator. When the stock price change, a frame surrounding the stock rectangle appears. Initially the frame is black (4), it then changes to red (5) or green (6) depending on whether the price went up or down. The SKF B stock is surrounded by a black frame, i.e. the frame just appeared and we do not yet know if it has increased or decreased. The HM B stock, on the other hand, is surrounded by a red rectangle, i.e. symbolizing that the price went down. LUND B is surrounded by a green frame, letting us know that the stock price has increased.

RESULT

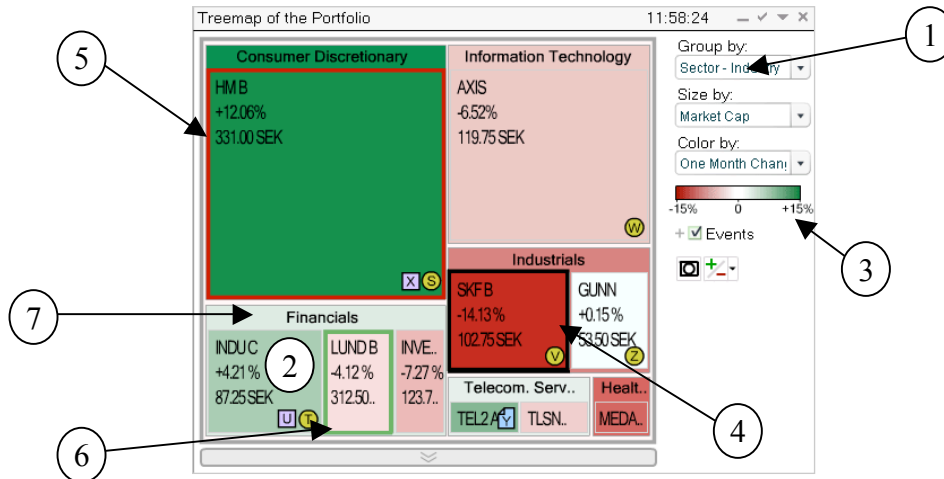


Figure 41. The treemap in the demonstrator.

In the treemap it is possible to *drill down* on a sector to take a closer look at its contents. This is done by double-clicking the sector title bar (7). For example, to drill down on the financials sector, double-click on the title bar, i.e. where it says “Financials”. The financial sector will then be enlarged to cover the entire treemap area, as seen in figure 42. When drilled down on a sector, a miniature of the original treemap is visible below the legend (8). This treemap makes it possible for the user to keep track of changes on the higher level as well. To return from a drill down to a higher level, you can either double-click again on the sector title bar or, click on the miniature treemap.

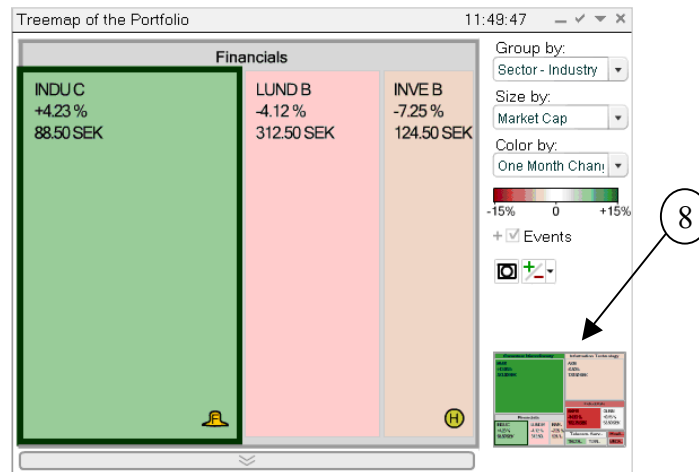


Figure 42. A drill downed treemap in the demonstrator.

7.1.2.2 Line chart

The line chart as it can be seen in the stock view in the demonstrator is shown in figure 43. In this line chart the HM B stock (1) is compared to the index OMXS (2). The line chart shows the one day change in percent (3). To the right of the chart we can see the range so far during the day, i.e. from just under 0 to 1.50 %, where both the stock and the index is currently around 0.75 % above 0. The lower part of the chart shows volumes for the HM B stock in thousands (4), and in this case ranges from 0 to 60 000. The area is in a grey shade to separate it from the line chart.

RESULT

To the right of the line chart is a *legend* which shows us that the time range for this line chart is currently set to 30 minutes, which can also be seen under the line chart (5). The legend also lets us know that the lines in the chart are colored by Ticker. This means that each line has a separate color, which color belongs to which line can also be seen in the legend (6), in this case HM B is purple and OMXS is yellow.

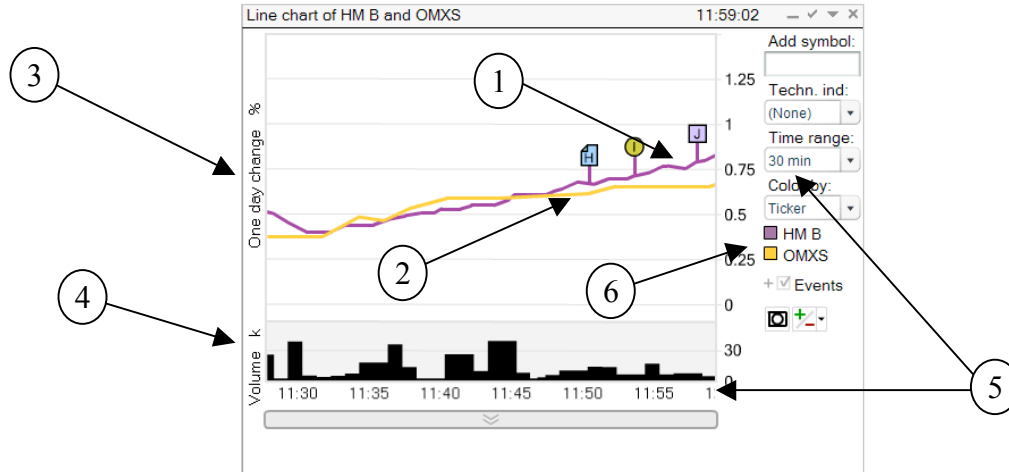


Figure 43. The line chart in the demonstrator.

To be able to compare more stocks or indices to each other it is useful to add them to the line chart. Above the legend in the line chart, there is a text field labeled *Add symbol* where it is possible to add more lines to the chart by entering the wanted ticker and then pressing enter, see figure 44.

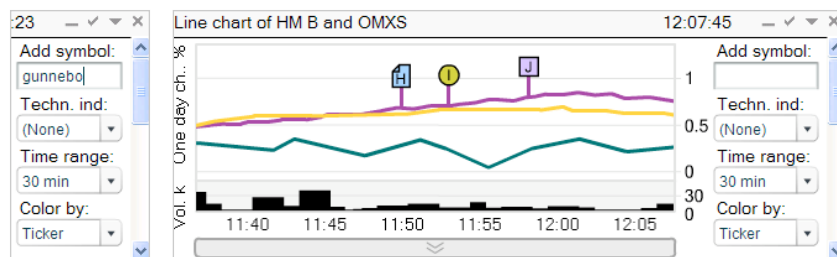


Figure 44. Add symbol in the demonstrator.
The Gunnebo stock is entered into the text field and added to the chart.

7.1.2.3 Scatter plot

The scatter plot is used in the demonstrator to show how well stocks perform relative to a benchmark. In figure 45 a scatter plot can be seen as it is used in the portfolio view. The X axis shows the percentage that each stock represents of the portfolio. The Y axis shows how much (expressed as percentage) over or under the benchmark the stocks have performed during the day. The *legend* tells us that the stock rectangles are sized by possession. In the legend you can see which stocks are represented by which colors by opening the color legend (1) located under the first dropdown. To be able to see specific stocks' movement over time you can select the stock(s) you want to see *history* for by clicking the stock rectangle(s) in the scatter plot and then check the box called history located below the legend (2). The stock rectangles will now leave traces

RESULT

(3) when they move over time. When you select one or more stocks in the scatter plot the rest of the rectangles will become transparent to make the selected ones stand out.

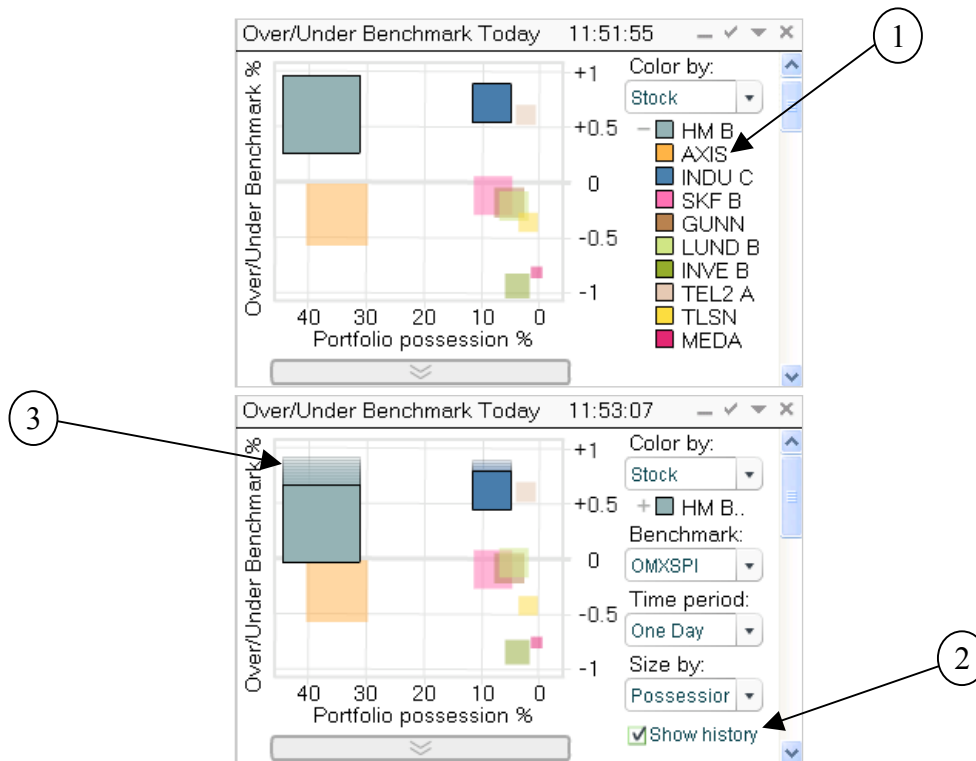


Figure 45. The scatter plot in the demonstrator.
The upper scatter plot with the color legend visible and the lower with history traces activated.

7.1.2.4 Bar chart

Bar charts are used in two of the views as well as among the miniature plots. The bar charts in the demonstrator are static and have not been implemented with any functionality. This is mainly because the movement would be very small and therefore hardly noticeable. The bar chart in figure 46 is the bar chart from the stock view and it shows the performance of the HM B stock and the consumer discretionary sector relative a benchmark.

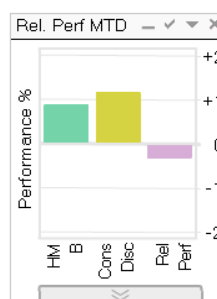


Figure 46. The bar chart in the demonstrator.

7.1.2.5 Events

In all three views there are one or two events tables which show different kinds of events related to the plots in the view. There are four kinds of events in the demonstrator, *news*, *insiders*, *reports* and *alerts*, which all have their own icon and color as can be seen in figure 47 (1). When an event appears in the events table, its corresponding icon will also appear in the plot. Each event also has a unique letter to make a connection between the event in the events list and its corresponding icon in the plot (2). In figure 47 (3), the news events are hidden in the treemap.

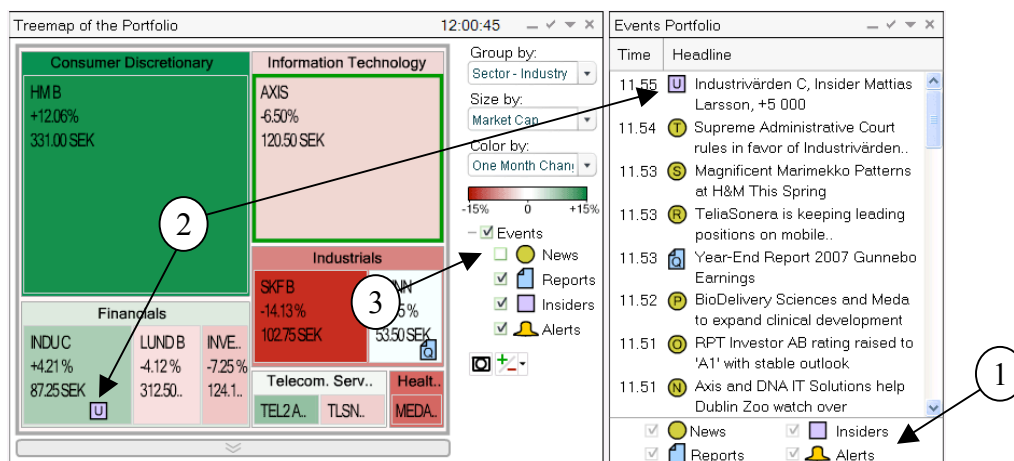


Figure 47. Events in the demonstrator.

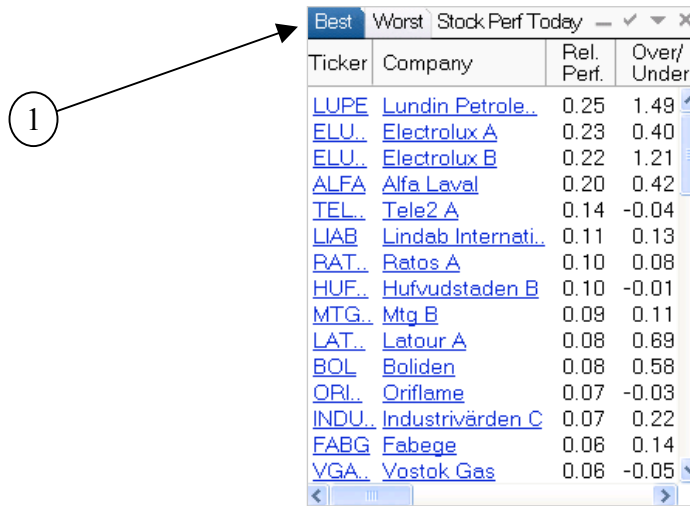
To the left is the events list which displays all events related to the portfolio.

If you do not want a specific type of event to appear in the treemap you can hide the events of that type in the plot but still keeping them visible in the events list. You can do this by clicking the plus (+) sign next to the events checkbox below the legend. This will expand a hierarchy where you can uncheck any event(s) that you wish to hide in the treemap. You can also uncheck the events checkbox without expanding it, which will hide all events. In figure 47 events of the type news are hidden (3).

7.1.2.6 Best/Worst

The best/worst table is used in all three views in the demonstrator. It contains a list of the best and worst performers over some specified time period. In figure 48 we see the stocks which have performed best over the current day. The list has two tabs (1), one for best performers and one for worst performers, which means that the best and worst performers are displayed in separate lists. The list that is currently visible has a highlighted (blue) tab. However, in the demonstrator the best/worst table is static and you cannot change between the two tabs.

RESULT



The image shows a window titled 'Best Worst Stock Perf Today'. A circled '1' with an arrow points to the 'Best' tab. The table below lists various stock tickers and their performance metrics.

Ticker	Company	Rel. Perf.	Over/Under
LUPE	Lundin Petrole..	0.25	1.49
ELU.	Electrolux A	0.23	0.40
ELU.	Electrolux B	0.22	1.21
ALFA	Alfa Laval	0.20	0.42
TEL.	Tele2 A	0.14	-0.04
LJAB	Lindab Internati..	0.11	0.13
RAT.	Ratos A	0.10	0.08
HUF.	Hufvudstaden B	0.10	-0.01
MTG.	Mtg B	0.09	0.11
LAT.	Latour A	0.08	0.69
BOL	Boliden	0.08	0.58
ORI.	Oriflame	0.07	-0.03
INDU.	Industrivärden C	0.07	0.22
FABG	Fabege	0.06	0.14
VGA.	Vostok Gas	0.06	-0.05

Figure 48. The Best/Worst table.

7.1.3 Features

In this section the four features that can be found in the demonstrator are described. These are: *alert*, *time controls*, *duplicate*, and *minimum/maximum*.

7.1.3.1 Alert

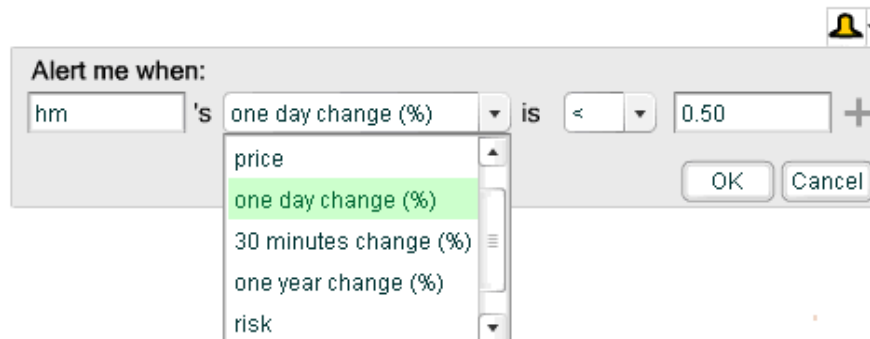
As mentioned earlier, the alert area is located in the upper right corner and is visible in all three views. A text indicates if any alerts have been set, see figure 49.



Figure 49. The alert area in the demonstrator.

To the left no alerts have been set and to the right an alert concerning HM B has been set.

To *set an alert*, you click the alert button with an icon of a bell. A dialogue box will then open in which you can enter the ticker you would like to set an alert for, as well as the values that should trigger the alert, see figure 50.



The image shows a dialog box titled 'Alert me when:'. It contains a text input field with 'hm', a dropdown menu with 'one day change (%)', a comparison operator 'is' with a '<' dropdown, and a numerical input field with '0.50'. A list of options is shown below the dropdown menu, including 'price', 'one day change (%)', '30 minutes change (%)', 'one year change (%)', and 'risk'. The 'one day change (%)' option is highlighted. There are 'OK' and 'Cancel' buttons at the bottom right.

Figure 50. The dialogue box for the alert.

When an alert has been set, the text next to the alert symbol will change to let you know how many alerts have been set. You will also see which ticker and to which value you have set the alert below the alert symbol. When an alert is *triggered*, the symbol becomes yellow and starts swinging from left to right, a frame will also start flashing around the alert, see figure 51. This will continue until the alert is deactivated, which is done by clicking on the alert. An information box will then appear showing information about the alert. By clicking the X in the upper right corner of the information box, the box will close and the alert will be deactivated.

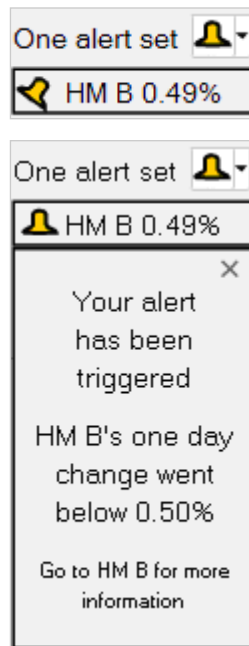


Figure 51. A triggered alert in the demonstrator.

The upper image shows a triggered alert, the lower image when the alert information box is opened.

7.1.3.2 Time controls

To go back in time and see what has happened earlier with a stock in a line chart, you can use the time controls. In the demonstrator they are hidden in the button below the charts to take up less space. In figure 52 the time controls have been made visible (1). The dialogue box for the time controls contains a *miniature* of the line chart showing the stock over a long period of time (2), in this case five years. The miniature is intended to provide the user with an additional way to move in time as well as being an overview. However, the miniature line chart was not implemented in the demonstrator due to lack of time, and therefore only work as an overview.

The five time control buttons (3) are located at the bottom of the dialogue box: *stop*, *rewind*, *play*, *forward* and *real-time*. If you click on the stop button or the rewind button, the line graph will enter static mode. If you click on the stop button, the chart will freeze immediately, but if you click and hold on the rewind button the line chart will go backwards in time until you release the button, upon which it will freeze at the point in time where the button was released. To be able to keep track of what is going on with the stock in real-time, the rightmost part of the line chart will continue moving on in real-time. To be able to separate the rewinded part of the chart from the

RESULT

real-time part, a frame will appear around each of the two parts of the chart, see figure 53. The rightmost part, showing real-time data is greyed out to further indicate that the two parts of the chart differ from each other (4). The play button can be used to play a historical sequence at normal speed. The forward button only works when the chart has been rewinded or stopped, for obvious reasons. If the chart has been rewinded or stopped, you can click and hold down the forward button can to fast forward the chart until the button is released, or until the chart has reached present time. It will then start playing at normal real-time pace. It is also possible to use the rightmost button to return to present time. Pressing that button will cause the chart to return immediately to present time.

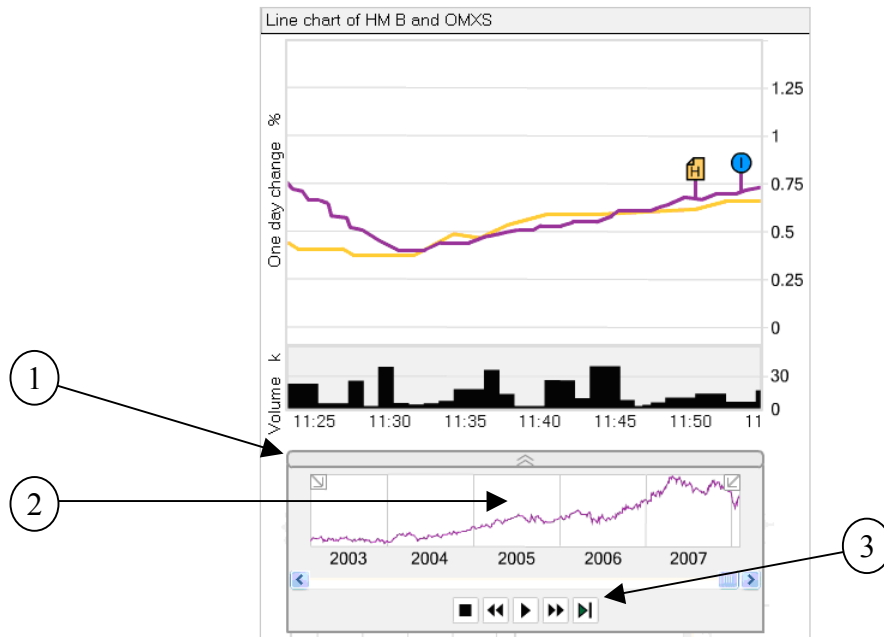


Figure 52. The time controls in the demonstrator.

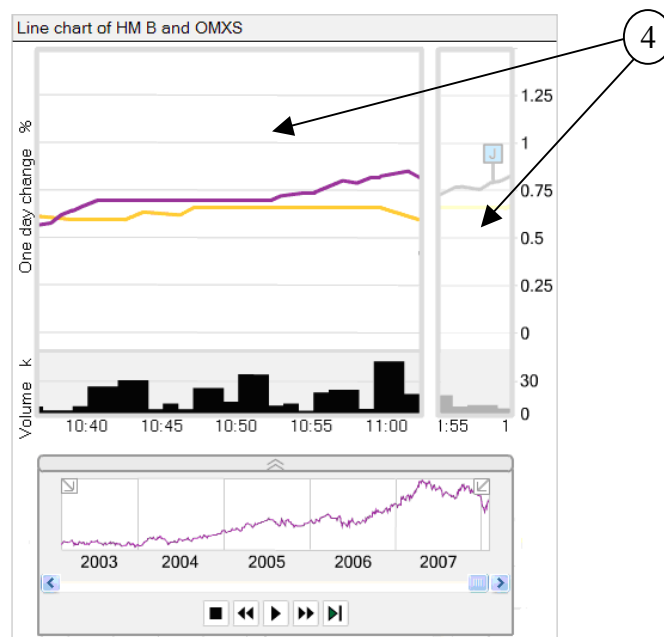


Figure 53. A rewinded line chart in the demonstrator.

RESULT

7.1.3.3 Duplicate

To be able to, for example, look at the performance of one specific stock at different times, it could be useful to make a duplicate of the plot and then rewind one of the plots. To make a duplicate of a plot you can click the duplicate button which is located below the legend, see (1) in figure 54.

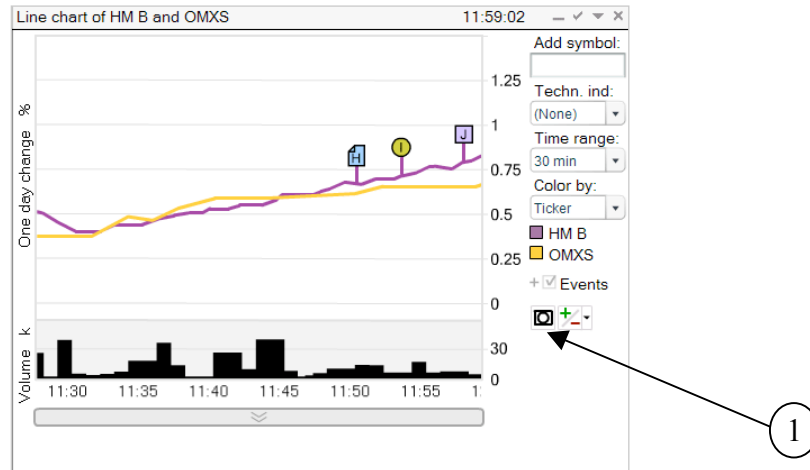


Figure 54. Before a duplicate is made.

The duplicate button is the left button located below the legend and the events check box.

Another identical line chart will then appear, see figure 55. As the figure shows, the space that the plot occupied prior to the duplicate was made is now split in two between the two plots to fit the new plot in the same space.

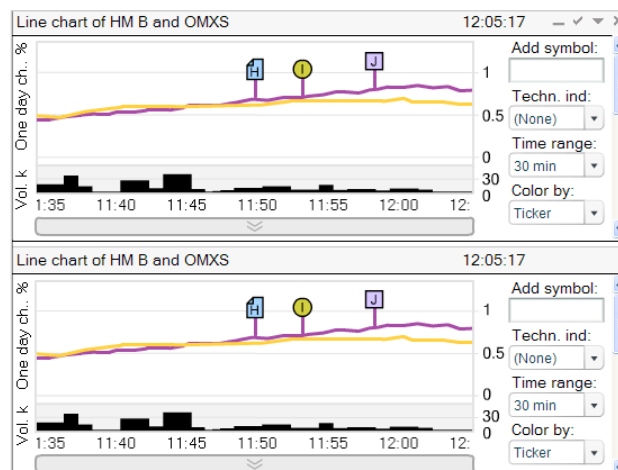


Figure 55. After a duplicate has been made.

7.1.3.4 Minimum/Maximum

To keep track of extreme values, such as which stock or portfolio has performed best or worst during some time period you can click the min/max button (1) located below the legend, next to the duplicate button, see figure 56.

RESULT

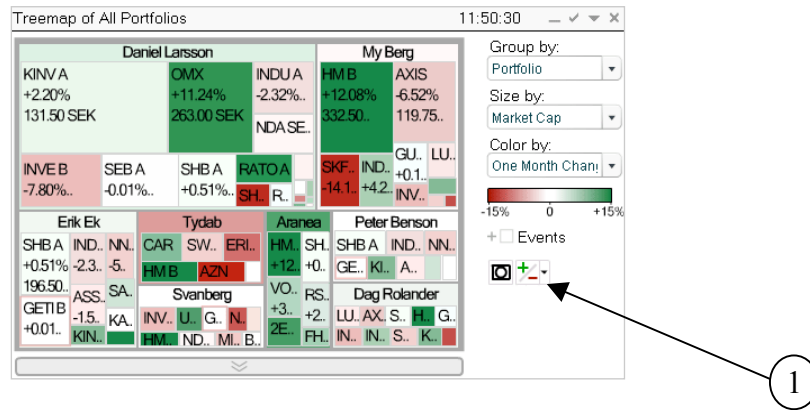


Figure 56. Before displaying minimum/maximum values.
The min/max button is the right button located below legend and the events check box.

A dialogue box will then open letting you choose what extreme values to keep track of. The first row in figure 57 says that you have chosen to show the stock with the largest increase in the past month. An additional row has been added, which is done by clicking the plus (+) sign. After you click the ok button the corresponding stocks can be identified by the symbols appearing in the treemap, see figure 58.

The dialogue box is titled "Mark:" and contains two rows of settings. The first row is: "The 1 stock(s) with the highest/largest increase (%) in the past month". The second row is: "The 1 stock(s) with the highest/largest decrease (%) in the past month". Below these rows are two checkboxes, one for "stock(s)" and one for "portfolio(s)". At the bottom right are "OK" and "Cancel" buttons.

Figure 57. The minimum/maximum dialogue box.

In the treemap all the rectangles representing HM B have a symbol showing that this is the stock that has increased the most during the last month. Tydab also have a symbol, but this symbol has an arrow pointing downward saying that this is the portfolio with the worst performance the last month. To the right of the treemap it is also possible to see exactly how much the stock and portfolio has gone up and down respectively in the specified time ranges, i.e. HM B has gone up 12.08% the last month and the Tydab portfolio has gone down 8.55% the last month.

Since it can become cluttered in the treemap if you enter many different values to keep track of, you can choose to hide the symbols and keep only the information to the right of the treemap visible. This is done by checking the checkbox next to the symbol. Figure 58 shows the treemap when the HM B check box has been unchecked. The figures to the right of the treemap are also updated along with the other updates in the plot so that you can keep track of the extreme values easily and without having to scan through the treemap.

RESULT

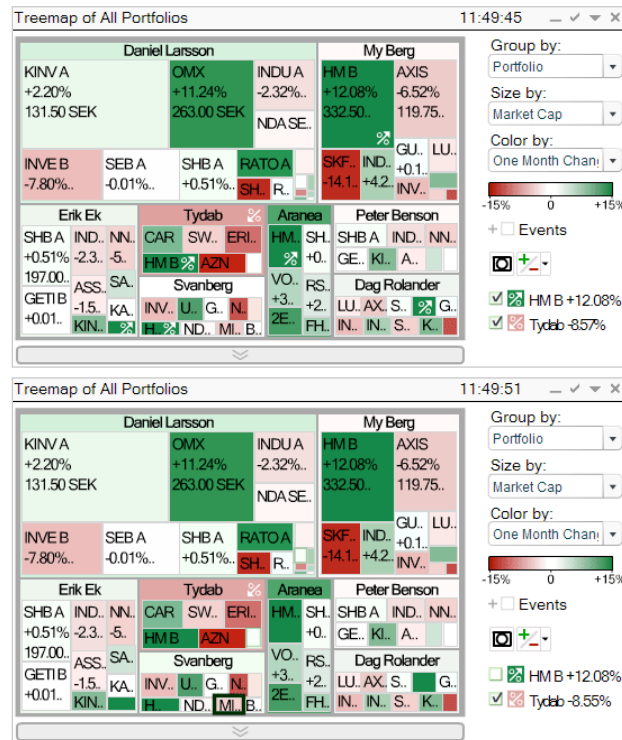


Figure 58. Minimum/maximum values displayed on a treemap.
In the lower image the min values are hidden.

7.2 Concept

In this section the whole concept for our design will be described and motivated. This final concept is a result from a further development of the demonstrator after performing the final evaluation. The concept is thereby larger than the demonstrator, since it contains both parts we were not able to develop in our prototype as well as some features and functionality, which have been developed further. All parts of the concept will be described and motivated and new designs will be shown for the parts that have been changed from the demonstrator.

The overall layout of the design will first be described. The four different views in the concept: *the market overview*, *the portfolio overview*, *the portfolio view* and *the stock view*, will then be described in further detail. Thereafter the graphic design, navigation and the interaction in the concept will be explained. Each display medium used in the concept is also described. Finally the different features in the concept are explained.

7.2.1 Layout

The layout of our concept consists of *four different views*; one market view, one view containing all portfolios, one view of a specific portfolio, and the last view over one specific stock. These four views are discussed more in detail in chapter 7.2.1.1 and forward. We have chosen to divide the information in these four views since we have identified that the users need to get information on different levels of detail. They foremost need to get an *overview*, both of the market and over all the portfolios they manage. The first view, the market view, therefore provides an overview and information about the whole market, while the second view contains information about all portfolios and gives an overview of the portfolios the user manages. The users also

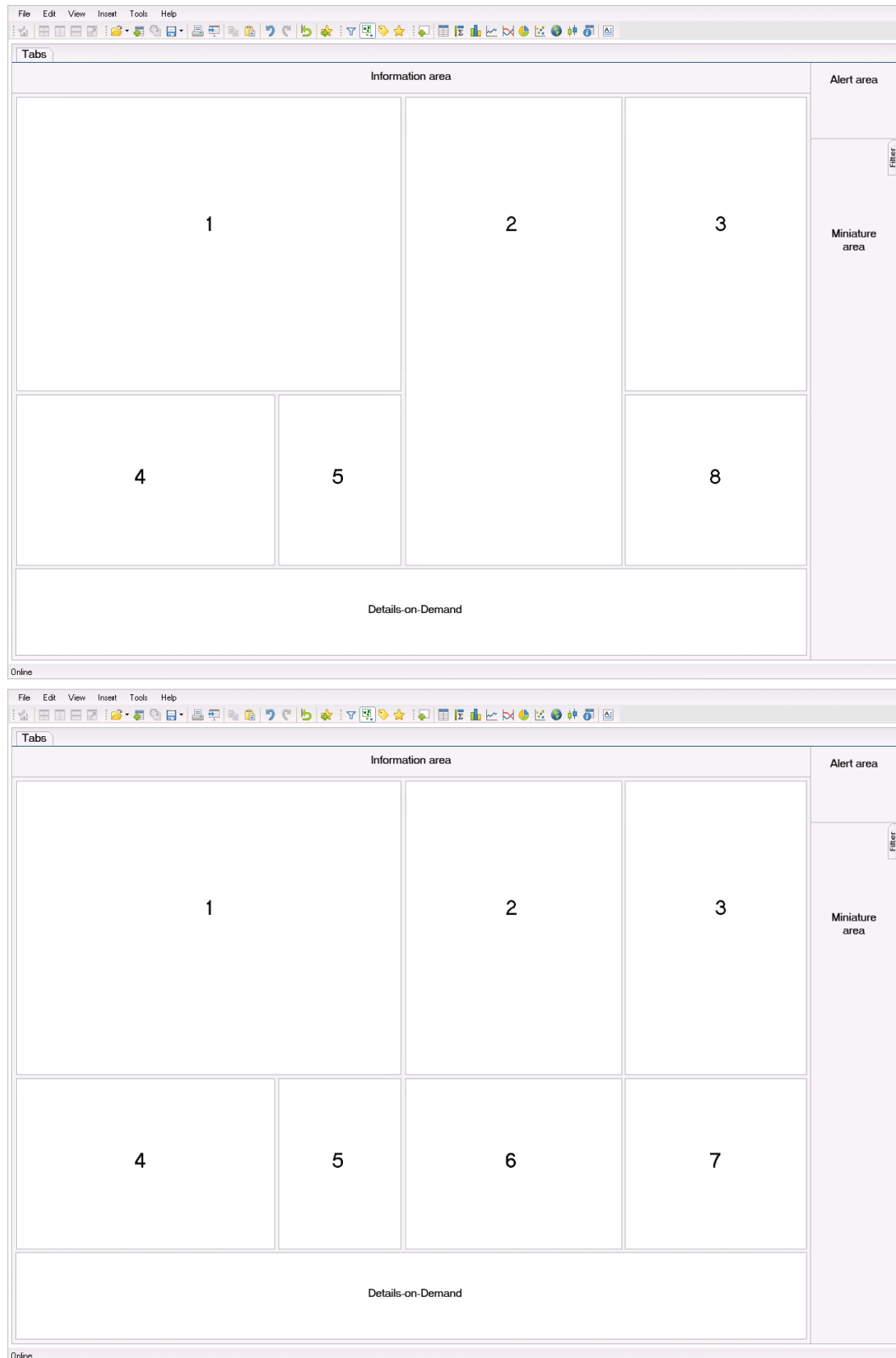
need to get more *detailed information* about both the portfolios and the stocks. Therefore we have developed the third view containing information about one specific portfolio and the fourth view containing information about one stock. Here you can perform deeper more detailed analysis of the data.

The layout in each view consists of a framework and a visualization area. The *framework* is always displayed and contains the same sort of information, independent of which view the user is in. You can therefore display general information not connected to one specific view, and also important information that always need to be visible in the framework area. The framework consists of the upper toolbar, which is the same toolbar that can be found in TIBCO Spotfire today and has not been explored in our thesis work. Below the toolbar are four *tabs*, which are used to navigate between pages. The idea is that four tabs are displayed by default, containing each of our four views, but the user should be able to add more tabs if he/she likes. The user might for example want to display information about two different portfolios and thereby needs to have two portfolio tabs open. This can be done either by clicking the empty fifth tab located after the four default tabs or in the menu that appears when you right-click in the work area. Below the tabs an *information area* is located, where real-time information about each view is displayed. In the upper far right corner in the design there is an area dedicated to important messages to the user; an *alert area*. Below this area *miniatures* in real-time are displayed. At the bottom of the design the *details-on-demand* window is displayed, which works in the same manner as in TIBCO Spotfire today. More detailed information about the different visualizations on the visualization area can be found in this window.

The framework is separated from the visualization area both by the use of color and borders. According to Few (2006) we tend to connect objects that are similar to one another in for example color and we also group objects together which are enclosed by any form of border. A light grey fill color has been used to connect the different parts of the framework. Borders have been used to enclose the objects in the framework, but also to distinguish between the different parts in the framework.

By arranging the concept in four views we are also able to maintain a *consistent structure of the visualization area* in all of the views. If a similar structure is maintained in the design it is a lot easier for the user to recognize, understand and navigate in the design and also to find the information he/she is seeking in each view. Nielsen means that being consistent in the design is very important [11]. Each view consists of a number of visualizations. The two views giving an overview of the market and of all portfolios contains six different visualizations, while the two views providing more detailed information about a certain portfolio or stock consists of seven different visualizations (see figure 59). In all the four views the most prominent information has been given the most space and is displayed in the top left corner (1). This is because it is very important to arrange the data in a dashboard so that it facilitates understanding (Few, 2006). The most important information shall be prominent and positioned in the upper left corner since the users in western countries scan the screen from left to right and top to bottom.

RESULT



*Figure 59. The layout in the concept.
The first structure is used for the views displaying overview, while the second structure is used for the views displaying more detailed information.*

RESULT

To the right of the largest and most prominent visualization a table containing events is displayed (2). This table is located as close as possible to the largest visualization, since they are the most related to each other. By using proximity Ware (2004) means that objects can be perceived to be related and linked to each other. The second table (3) is also related to the largest visualization and is therefore placed beside the table of events. The second table (3) contains a list of best/worst performers. We needed to be able to display large amounts of text both for events and best/worst performers. We therefore chose to display the events and the best/worst performers in tables. Tables are good at giving more detailed information and structure text in a good way (Wallgren et al., 1996). The users are also accustomed to having this information displayed in tables in the software they use today.

The fourth and fifth visualization provides a comparison of relative performance of the data in the view (4,5). A scatter plot is used to show performance compared to a benchmark, while a bar chart is used to show relative performance. Since both of these visualizations show more detailed information and also the same type of information they are placed close together. A scatter plot was chosen as display medium for showing performance compared to a benchmark, since it is very good at displaying relationship between two variables. The bar chart was chosen since this visualization needed to display summative values which bar charts are very suitable for (Few, 2007; Wallgren et al., 1996).

Two mini-treemaps are displayed to provide an overview of the data (6,7). Since this is not the most prominent information in the views we decided to place these visualizations in the lower right corner. As well as they are to provide an overview, they should also provide a mean for navigation between the views. The first mini-treemap (6) is an overview of all portfolios. With the help of this treemap you can navigate to the second view and tab containing information about all of the manager's portfolios. The second mini-treemap (7) provides an overview of the market and is connected to the first view and tab containing information about the market. Treemaps were chosen as display medium for these two overviews, since they needed to contain a lot of hierarchically structured information on a very small area of space, which treemaps present in a very space-efficient manner (Bederson et al., 2002). When the market view or the view of all portfolios is displayed, it is superfluous to provide two mini-treemaps, since one of them will be a duplicate of the largest visualization already displayed in the view. Therefore, only one mini-treemap is displayed in the lower right corner in these two views (8). The table of events is in these views extended and stretches from the top of the visualization area to the bottom.

To be able to make use of the limited space as well as possible the legend is by default hidden for the visualizations where the user does not need to make changes often. By hiding the legend the most important information, the visualizations, are given more place and the design also feels cleaner.

The *content on the visualization area* depends on which view you are displaying at the moment. We have created a default content for each view, which will be described in 7.2.1.1 and forward. During our pre-study we discovered that the users seldom have time to alter any settings or make changes in the software they use. We therefore thought it was very important to offer default views to the users and provide them with the most important information. Our concept is thereby very inspired by *dashboard*

design. Dashboards have become very popular within business intelligence and are to produce a simply and instantly understandable visual display, while they at the same time need to be able to fit large amounts of data into a small area of space (Few, 2006). Since this was what we needed to do in our design to provide an overview of the large amounts of data, the concept we developed is very much inspired by theories within dashboard design.

Few (2006) means that the information provided by the dashboard to give an overview, should fit in one single screen. However, when providing more detailed information Few means that separate screens can be used. Our design consists of four different views, all fitted in four separate screens. Since the different views display different levels of detail and the information does not need to be displayed together, the use of several separate screens is not an issue.

The pre-study also showed that the users all had their own style and way to perform their work; they looked at very different kinds of information. It therefore, besides providing the users with default views, also became very important to provide the users with the possibility to display the information they want to display. The design is thereby built up in the same manner as in TIBCO Spotfire, using different windows for each visualization. By using windows the user can in a simple way *rearrange*, *remove*, *minimize*, and *add new windows*. We have added a minimize-button for each window in the title bar. This gives the user freedom and control, which is very important according to Nielsen [11].

While conducting our pre-study we also noticed that the users often work with several computer screens. Since they have access to several screens our concept does not have to be limited to fitting into one computer screen. It should therefore be possible to *detach visualizations* from the interface and drag them to another screen. This was a request that emerged during our final user test and it would make the interface less cluttered and it would also be possible to display a lot more information if a larger screen space can be made use of.

7.2.1.1 Market overview

The first default view contains information about the market as a whole. This view is to give an overview of what is going on in the market. It consists of six main visualizations (see figure 60): a large treemap, a scatter plot, a bar chart, a table of events, a table of best/worst performers, and one mini-treemap. In this view the most important information is displayed using a large treemap. Here you can see the development and performance of the stocks in the market and you can also see the distribution of the stocks. The treemap was chosen as a display medium for this visualization, since it presents hierarchically structured information in a very space-efficient manner (Bederson et al., 2002). This visualization needed to be able to present large amounts of data, since it should give an overview of an entire market.

The table containing events displays all events connected to the displayed market. The best/worst table in this view displays the stocks in the market that have performed the best and worst. The scatter plot displays how the sectors on the market are performing compared to a benchmark. The bar chart in this view displays the relative performance for each sector in the market. An overview of all portfolios is also given by the mini-treemap.

RESULT

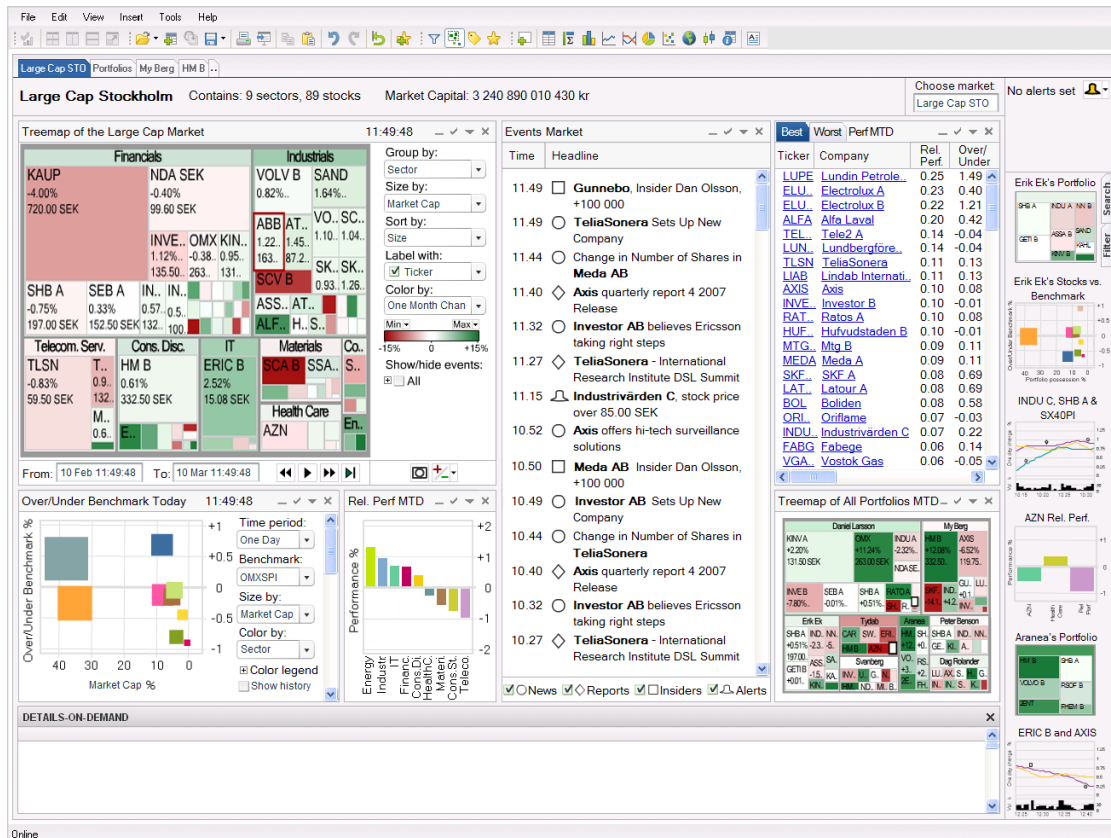


Figure 60. The market overview in the concept.

7.2.1.2 Portfolio overview

Our second default view also provides an overview but of the portfolios of one user instead of the market. In this view information connected to all the portfolios that one user manages is displayed. This view also consists of six main visualizations (see figure 61): a large treemap, a scatter plot, a bar chart, a table of events, a table of best/worst performers, and one mini-treemap.

The most important information is also here displayed using a large treemap, since this visualization also need to be able to present large amounts of hierarchical data to give an overview of all of the portfolios. The treemap provides information about the development and performance of all of the portfolios you manage. The table containing events displays events connected to the content of the portfolios you manage. The best/worst table in this view displays the portfolios that have performed best and worst. The scatter plot displays how the portfolios are performing compared to a benchmark. The bar chart in this view displays the relative performance for each portfolio. An overview of the market is also given by the mini-treemap.

RESULT

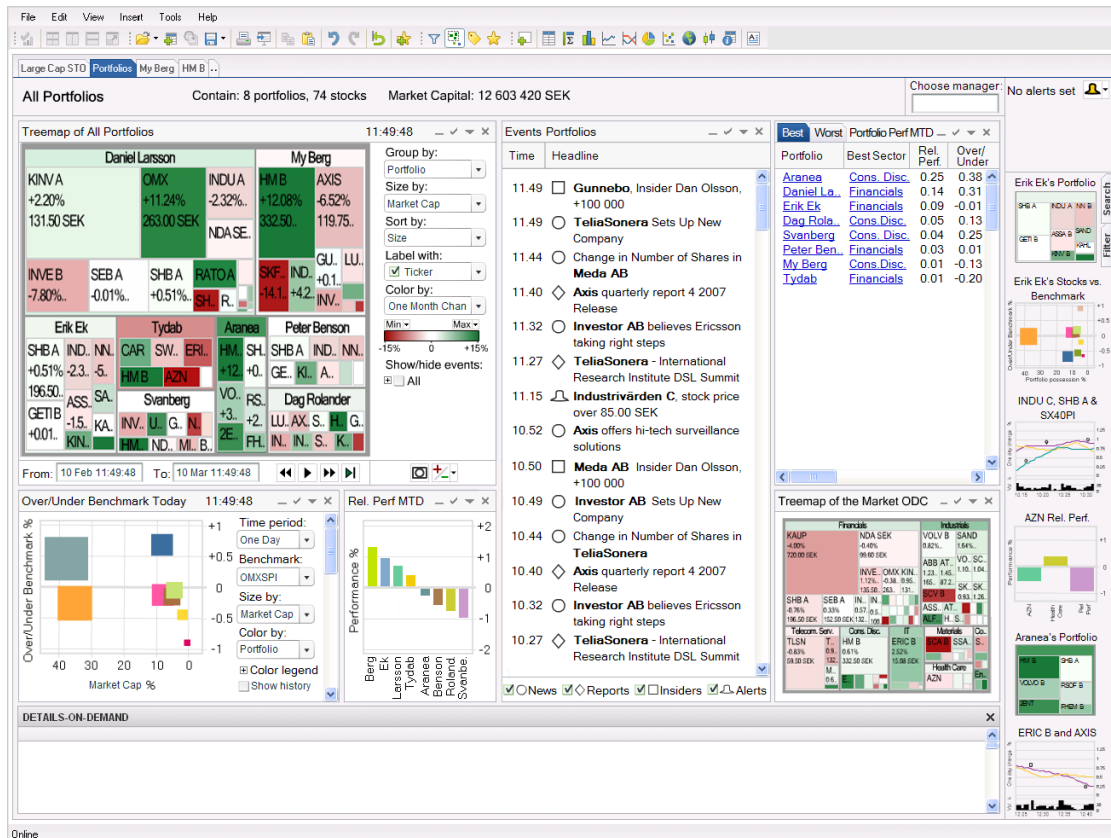


Figure 61. The portfolio overview in the concept.

7.2.1.3 Portfolio view

The third default view provides more detailed information about one specific portfolio. In this view information connected to one portfolio is displayed and can be analyzed. This view consists of seven main visualizations (see figure 62): a large treemap, a scatter plot, a bar chart, a table of events, a table of best/worst performers, and two mini-treemaps.

The most important information is once again displayed using a large treemap, since the visualization is to present hierarchical data. The treemap provides information about the development and performance of this specific portfolio. The table containing events displays events only connected to the portfolio. The best/worst table in this view displays the stocks in the portfolio that have performed the best and worst. The scatter plot displays how the stocks in the portfolio are performing compared to a benchmark. The bar chart in this view displays relative performance between the portfolio in this view and a benchmark. An overview of the market and of all portfolios are also given by two mini-treemaps. On the treemap displaying all portfolios, the portfolio that this view displays is marked by a blue frame. This is to show how the detailed information in this view is related to the other data.

RESULT

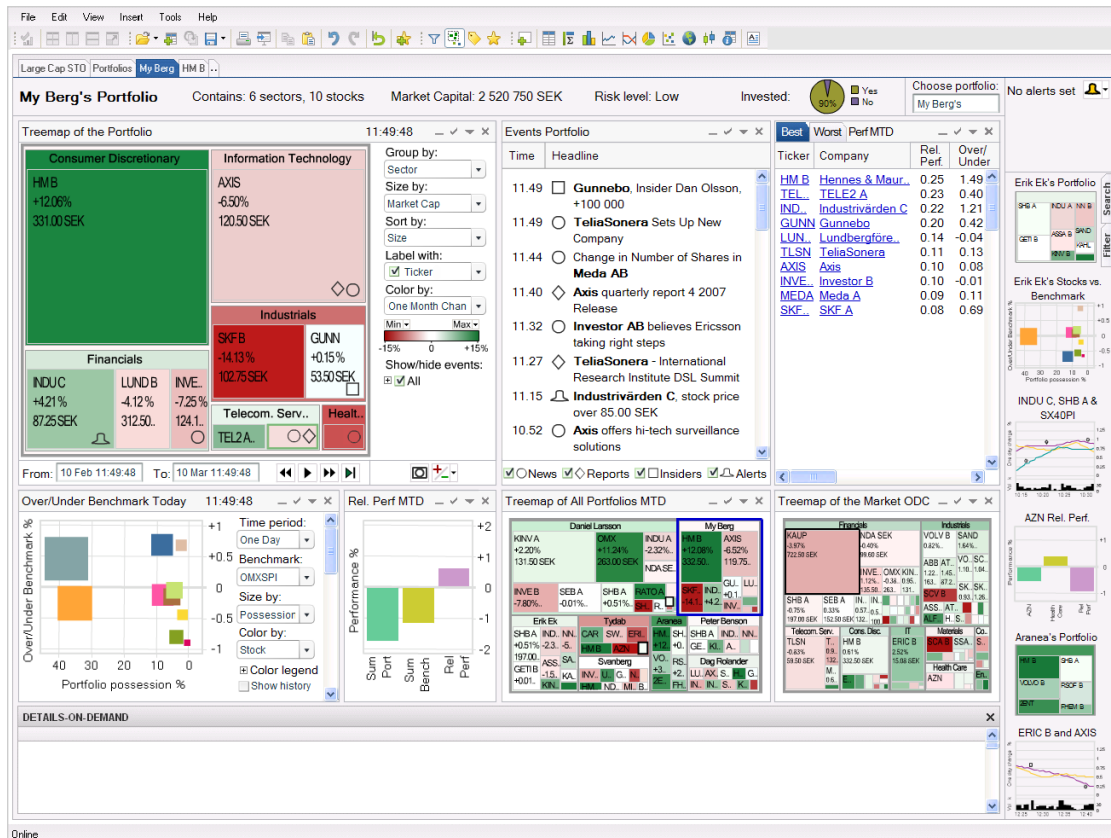


Figure 62. The portfolio view in the concept.

7.2.1.4 Stock view

The fourth default view provides more detailed information about one specific stock. In this view information connected to one stock is displayed and can be analyzed. This view also consists of seven main visualizations (see figure 63): a large line chart, a scatter plot, a bar chart, a table of events, a table of best/worst performers, and two mini-treemaps.

The most important information in this view is displayed using a line chart. This visualization is to display the stock's development over time and provide the possibility to compare its development with an index or another stock. A line chart was therefore most suitable for this visualization, since it is ideal for showing development over time and makes it possible for the user to distinguish patterns in the data (Wallgren et al., 1996). The users are also accustomed to using line charts to analyze stock development.

The table containing events displays events connected to the stock displayed in the view. If other stocks or indices have been added in the line chart, the list of events however also contains events related to these. The best/worst table in this view displays the best and worst performers of the stocks in the same sector as the stock that is being displayed. The performance of the stock in this sector compared to a benchmark is also displayed in the scatter plot. The bar chart in this view displays the performance for the stock and the sector the stock belongs to as well as the relative performance between these. An overview of the market and an overview of all portfolios are also given by two mini-treemaps. In these two treemaps the stock that this

RESULT

view displays is marked by a blue frame. This is to show how the detailed information in this view is related to the other data.

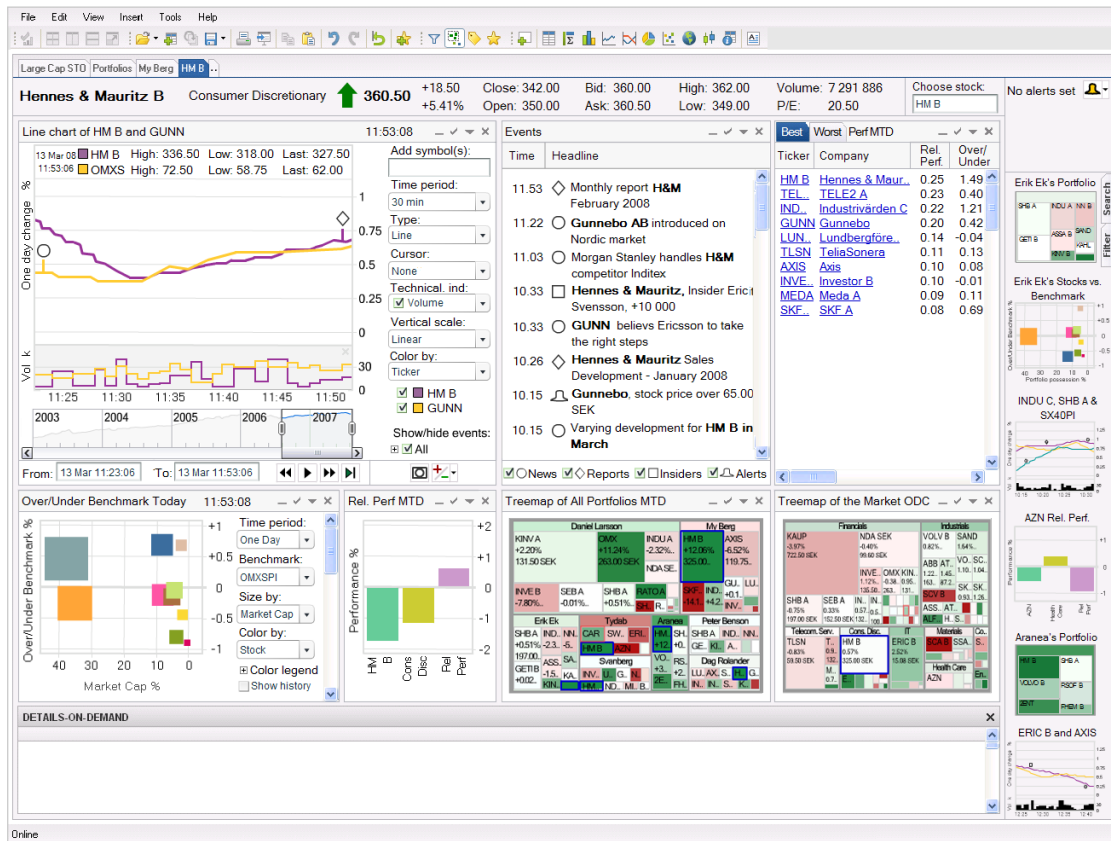


Figure 63. The stock view in the concept.

7.2.2 Graphic design

The graphic design of our concept is based on the design in TIBCO Spotfire. We used this design as a starting point and developed it further to fit our concept as well as possible. The visualizations in the concept all use *color*. Few (2006) means that color should not be overused in a design; it should only be used when it adds something to the visualization. The use of color in every visualization could however not be avoided in this design, since color is necessary to use to represent the information in the visualizations correctly. We tried to use as little color as possible in the rest of the design to tone down the interface. For example, we use light grey controls, borders, title bars and frames for the windows in the design. The backgrounds in the design are kept as neutral as possible by using white and light grey colors. It is also important to be consistent when it comes to colors and using the same colors to relate the data in the design (Few, 2006; Tufte, 1983). The colors for the different visualizations have thereby been separated in our design, since this data should not be related. The treemaps use a color range from red to green, and these colors are not used in any other place in the design. The visualizations in the design are also *toned down* as much as possible to not be too dominant. The axes and the gridlines, in the line chart, scatter plot and bar chart, as well as the borders in the treemap are all displayed in a light grey color, instead of black.

RESULT

We have also tried to keep the different *icons* in the design as simple and toned down as possible. The buttons consist of grey frames and outlines. This design was chosen to make the buttons fit the rest of the interface. The icons for the events are also kept as simple, colorless, and stylized as possible.

Each window in the design contains a *title bar* (see figure 64). The title bars in the concept contain dynamically updated headlines. These headlines describe the content of the plot (1) and are updated along with changes in the visualization. A time and date stamp (2) is also displayed in the title bar for the visualizations, to inform the user of which plots that display static and real-time data respectively. To the far right in the title bar for each window, four buttons are displayed. They are used to minimize the window (3), define settings for it (4), show/hide the legend (5) and to close the window (6).



Figure 64. Title bar.

A *toolbar* has also been added at the bottom of each window (see figure 65). Since the final evaluation proved it was very hard for both the users and the experts to find the buttons placed in each window, such as the duplicate and the min/max button, we wanted to distinguish the buttons from the rest of the functions in each window. The users also expressed a wish that the controls used for going back and forth in time should be more accessible and not placed in a dialogue box. We therefore created the toolbar to fit both the controls to go back and forth in time as well as the duplicate and min/max button. By adding a toolbar the most frequently used features for each visualization could be placed here. By placing a toolbar in each visualization it also becomes clear for which visualization the action is going to be performed. The user can choose whether to display the toolbar or not, exactly as with the legend.

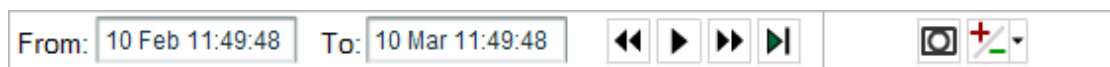


Figure 65. The toolbar.

The toolbar contains dynamically updated text fields which can be used to change the time period of the visualization. Buttons for going backward/forward in time, as well as a play-button and a button used for going back to real-time is placed here. The duplicate button and min/max button is placed to the right in the toolbar.

7.2.3 Navigation and interaction

The primary navigation in the concept consists of *movement between different views* and pages in the design. This can be done in two main ways; by using the tabs or by right-clicking. By using the tabs you can jump between the open views in the design. If you right-click for example in a portfolio in any visualization you can choose to go to the portfolio view to display details about this portfolio. You can also choose to

RESULT

open a new portfolio tab to display the portfolio in; if you for example want to compare this to a portfolio you already have open. We tested different alternatives for navigation between views in our evaluations and it proved to be very surprising for the user if a single-click or a double-click in a view navigated them away from the current view to another view. This was probably since the interaction within the views consists of single-clicks and double-clicks. It was therefore inconsistent and impossible for the user to know when a single- or double-click would navigate them away from the page and when it would not. We therefore separated the navigation between views much from the navigation within views, by using right-click, to avoid uncertainty for the user and inconsistency in the design.

During our final evaluation it proved to be difficult for some of the users to find for example the buttons to set an alert, make a duplicate or mark minimum and maximum values in the visualization. In the demonstrator *tooltips* were not implemented, but in the concept this is an important feature. They are very efficient when it comes to explaining the purpose of controls in a toolbar (Cooper & Reimann, 2003). Tooltips are also used throughout the concept for all visualizations in the design. They provide a lower level of detail of the information to the users. During the pre-study we noticed that the users need both overview of the data as well as details. Since parts of the design consist very much of information to give overview we also want to provide the users with an easy way to collect more detailed information. By inserting small charts in the tooltip we also believe that the information in the tooltip can act as a good complement for the information displayed in the visualization. For example, tooltips for a stock in the treemap consist of more detailed information about the stock as well as a line chart showing development over time for the stock (see Figure 66). The line chart in the tooltip can thereby show information which the treemap is not that good at presenting.

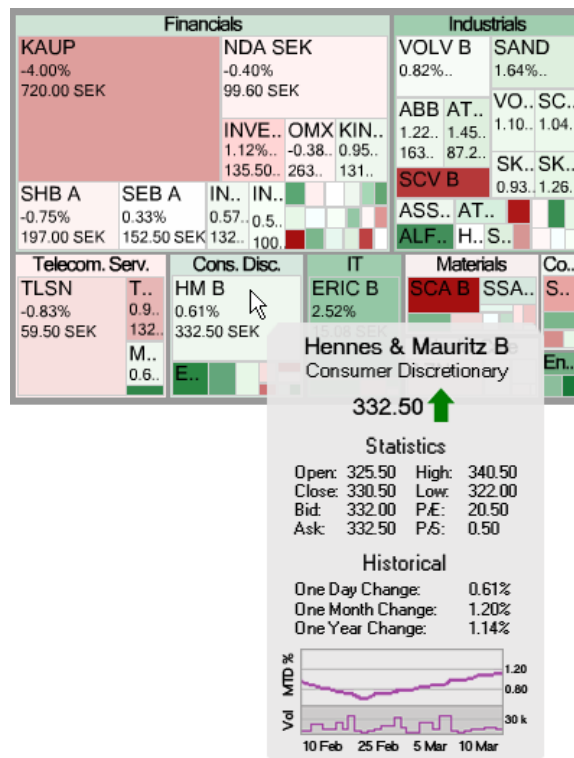


Figure 66. Tooltip displayed in a treemap.

RESULT

When it comes to *interaction* in the visualizations in the concept the same interaction techniques are used as in TIBCO Spotfire (see Appendix 1). During our process we evaluated if this current interaction also would fit real-time data and our concept. Since the real-time data proved not to move that much, the current interaction techniques proved to work very well for our design. All of the visualizations in the concept thereby consist of interaction such as highlighting of items on mouse-over, selection of one item on single-click, and selection of several items by Ctrl-click or drag rectangle.

Interaction specific for one display medium will be further described in the next section, 7.2.4.

7.2.4 Display mediums

The concept primarily contains six different types of display mediums. These are: *treemaps*, *line charts*, *scatter plots*, *bar charts*, *tables*, and *sparklines*. Each of these display mediums is used for visualizing different kinds of information on the visualization area in the concept. Each medium is described further in this section.

7.2.4.1 Treemaps

The treemaps used in this concept are updated in real-time and show hierarchically structured information on different levels of detail (see Figure 67). The treemaps in the concept can display this information in several different ways. We wanted to give the user the possibility to adjust in what way the information should be displayed in the visualization. By providing different ways to look at the information in the same visualization the user can perform deeper and more detailed analysis of the data. This is something that is very important within information visualization and also for a business analytics tool. In the user situation we identified, it is also important to provide the users with a mean to easily look at the data in several different ways, so that they can perform correct analysis of the data. This can be done by using the treemap legend.

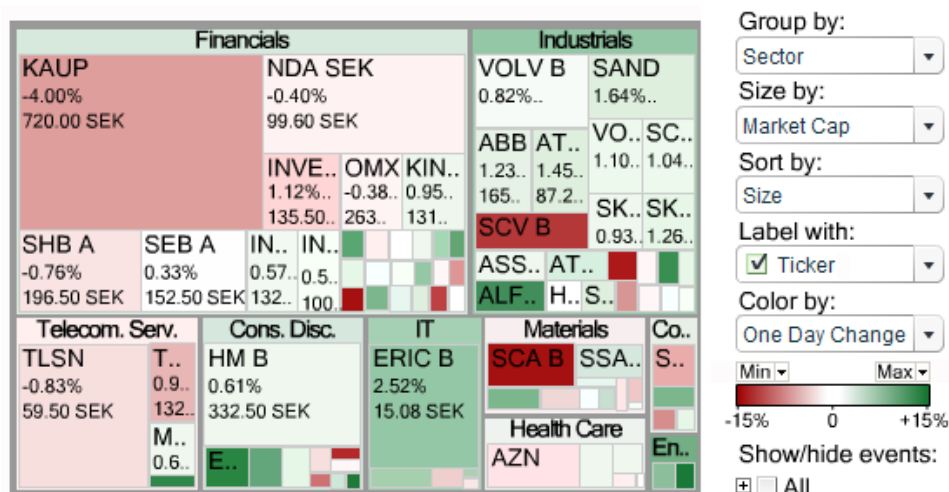


Figure 67. The treemap in the concept.

It displays the Large Cap Market, is grouped by sector, sized by market cap, sorted by size, labeled with ticker, one day change, price, and colored by one day change.

RESULT

The *legend* to the right of the treemap enables the user to make different settings concerning how the information is displayed in the treemap. The legend consists of five dropdowns, one color range, and a way to show and hide events on the treemap. In the first dropdown you can choose how to *group* the treemap, either by sectors, industry, sector and industry, or have no grouping at all. By using enclosure and proximity when grouping the data the user can easily distinguish how the data is categorized, see what data is related, and how the data affects one another (Ware, 2004).

You can also choose what the *size* of the rectangles in the treemap shall represent. This can be done in the second dropdown in the legend. Here you can for example choose to size by market capital, P/E ratio, P/S ratio, stock price, percental change in price, or choose to display all rectangles in the same size. By providing the user with the choice to change what the size in the treemap should represent they are able to rank the importance of the information according to their own desires (Few, 2006). Since we also, during the pre-study, identified that the user looks very much at different kinds of extremes, it can be good to let the size represent different kinds of importance. It is very easy to distinguish extremes in a treemap, for example the largest or the smallest rectangles in the visualization.

We also wanted to provide the users with the possibility to *sort* the information in the treemap according to their own wishes, since we had noticed that they looked at the information in very different ways. You can choose to sort the information by size, color, company name or ticker in the third dropdown in the legend.

The next dropdown in the legend is used to choose what to display in the *labels* in the treemap. Here you can choose to display for example ticker, company, price, market cap, percental change in price, P/E ratio, P/S ratio or no labels at all. If the treemap is to display very large amounts of data, for instance an entire market, it can become cluttered. It can then be good to provide the user with the possibility to decide how much information should be displayed in each label on the treemap. Checkboxes are placed inside the dropdown to make this possible. The users are accustomed to using tickers to identify each stock. We therefore choose to display the tickers in a larger size than the rest of the text. By using larger titles for each stock it makes it easier for the user to distinguish between the data displayed in the treemap (Few, 2006).

In the last dropdown, the user can choose what the *colors* in the treemap should represent. Each rectangle can be colored by for example one day change, one month change, one year change, and turn-over. The coloring also makes it easy for the user to identify extremes in the treemap, such as the most red or green rectangle in the treemap. A color range from red to green is used as a default, since the users are accustomed to this color range for this purpose. However, it should be possible to change the colors representing minimum and maximum values. It can be done in the color range displayed below the last dropdown in the legend. It informs the user of what the colors represent and between which values the data in the treemap is located. To change the colors in the range the user can click the Min- and/or the Max-button. To provide the user with an easy way to change the colors in the design is important considering color-blindness (Cooper & Reimann, 2003). The color range gives the user a possibility to select data in the treemap by drawing a rectangle on the color range.

RESULT

Below the color range, *miniatures* of the treemap can be displayed. These miniatures are displayed only if the user has performed a drill down in the treemap, see figure 68. Drill down means that you move between different levels of detail in the treemap. When a drill down has been performed miniatures are displayed of the higher levels in the treemap. The miniatures consequently have three different purposes; to provide the user with information about what level of detail he/she is looking at, an overview of the data in the other levels along with the real-time updates of this data, and also a mean for navigation between the different levels of details. The user can navigate to another level of detail by clicking the miniatures.

Drill down makes it possible to display more detailed information and it also enables analysis of lower levels of data in the treemap, see figure 68. Drill down is a feature that is very useful in the treemap since it otherwise has problems with displaying detailed information. To provide the user with the ability to drill down on the information and see details is also very important when designing dashboards (Few, 2006). Drill down can be performed in the treemap by double-clicking the sector or industry you want to look closer at. You can also right-click in the treemap and choose to drill down on the data with the help of the menu that appears when you right-click. To make the user aware of the fact that a drill down has been performed and that the treemap displays data on another level, an animation occurs when a drill down is performed. The rectangle containing the data that the user has performed drill down on, slides open to display this rectangle over the whole treemap. Motion is according to Few (2006) a powerful attention-getter and in this case the user will be informed that the displayed information in the treemap has changed.

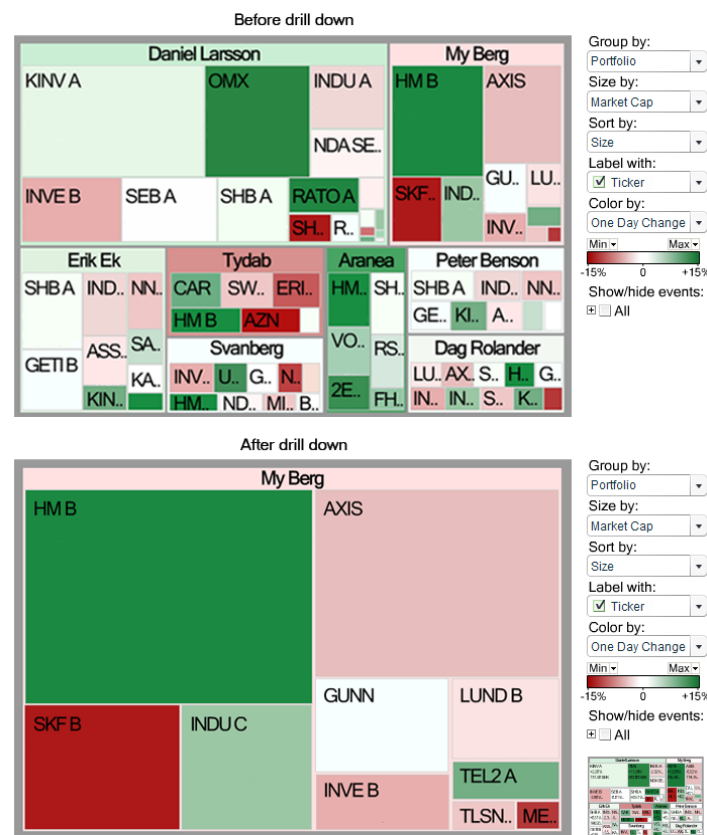


Figure 68. A drill downed treemap in the concept.
A drill down has been performed in the treemap of My Berg's portfolio.

Besides drill down, the interaction in the treemap works in the same way as it does in TIBCO Spotfire. That is, you highlight items on mouse-over, select items by clicking on them, all selections are colored in blue, you can also select multiple items by using drag-rectangle, etcetera.

All treemaps in the concept are *updated in real-time*. These updates are shown with the help of a frame surrounding the rectangle which the update concerns. When the update occurs a black frame is displayed. This frame catches the attention of the user. The user then gets more information about what happened, since the color of the frame changes. It becomes either red or green, as default, depending on if the update was positive (green) or negative (red). After the frame has changed color a fading effect is used to make the frame disappear (see Figure 69). The frame is visible for 5 seconds. This creates a form of flickering effect since the frame suddenly appears and draws the attention of the user. Few (2006) means that flickering is a very useful feature when handling and monitoring real-time data.

We also considered other ways to display real-time updates in the treemap, such as letting the frame become smaller and smaller and move to the middle of the rectangle or letting the frame move up or down to show change instead of changing the color of the frame. In the example where the frame would move towards the middle of the rectangle, the information in, and the color of, the rectangle would be obscured by the frame. Letting the frame move up or down to show change also proved to be very hard to perceive. Other suggestions were also discussed, such as dividing each rectangle into two triangles and letting one triangle show the real-time updates and the other triangle represent the color of the rectangle. This idea was discarded since the user would not instantly know what each triangle represented and there was a risk that the data thereby would be misinterpreted. Rectangles that are small and hard to see which color they have would also become even smaller since they should be divided in half.

When investigating the real-time aspect of the treemap we discovered that the updates occur very frequently, especially when a higher level of detail is displayed. The consequence of this will be that the frames representing real-time updates will be displayed in different places all the time in the treemaps. If the updates occur all the time and the frames are displayed very often their purpose will somewhat be lost. The idea is that the frame should draw the users' attention and inform the user that something has happen. If this happens all the time, there is a risk that the user cannot pay attention to all of these updates. It might also not be necessary to see all of the updates, and that the users are the most interested in larger changes. We therefore had an idea of letting the user decide what updates that should be displayed in the treemap. If the user wants to see everything that happens, which we have seen is the case sometimes; he/she should be able to do this. But if the user only want to display larger changes this should also be possible. The real-time updates could for example only be displayed if a stock has increased or decreased with more than 0.5% within a specified time period. In this way the user can filter out the updates that are not of importance. This feature has however not been developed further in this concept.

RESULT

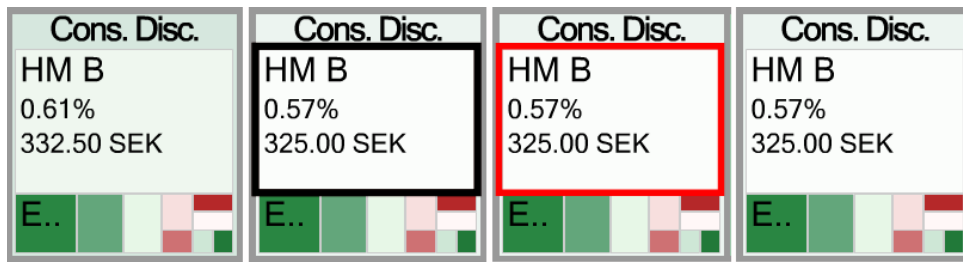


Figure 69. A real-time update in the treemap.

A black frame appears around HM B when it declines. The black frame change color from black to red after about 2 seconds to show that HM B has declined. The red color then fades out.

7.2.4.2 Line chart

The line chart shows development over time for one or several stocks and benchmarks, see figure 70. The x-axis in the line chart displays time, since the line chart is supposed to show the stocks development over time. The y-axis, by default, displays change in percentages. The volume for the stock is by default also displayed, since the users look very much at this information. In the concept, volume is displayed with a line instead of bars. This is because it showed to be very hard to display volumes in the same chart for several tickers, using bars. If lines are used instead, it is possible to display volume for all tickers, as lines are not as space-filling as bars, and they can also be colored to match the different tickers in the line chart. To separate the line chart from the chart displaying volume, colors and a grey border have been used. The use of different colors and also enclosure distinct objects from one another (Few, 2006). The information in the line chart can be displayed in many different ways, since the users need to be able to make different kinds of analysis of the data. The legend can be used to modify the content and appearance of the line chart.

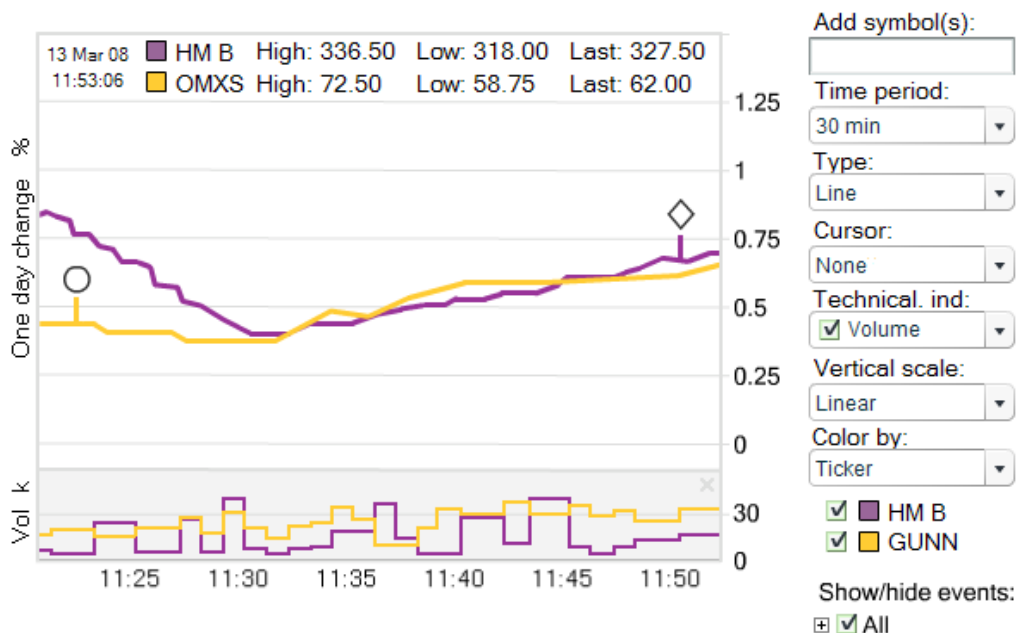


Figure 70. The line chart in the concept.

It shows the development in stock price for Hennes & Mauritz B and Gunnebo for a time period of the last 30 minutes. Volume is also displayed for the stocks.

RESULT

The *legend* to the right of the line chart enables the user to make different settings concerning how the information is displayed in the visualization. The legend for the line chart consists of one textbox, six dropdowns, and one explanation of the colors in the line chart. The first thing you can do in the legend is to *add a symbol* to the line chart. You can choose to add another stock or an index. This enables comparison of the data, something that is very important for the users. To add a new symbol to the line chart you type what you want to add in the text box. A list is then displayed of the alternatives that match your search. This list is updated dynamically when the user types in the letters in the text box, see figure 71. It is important to give this instant feedback to make the user aware that the system reacts to his/her operations (Cooper & Reimann, 2003).

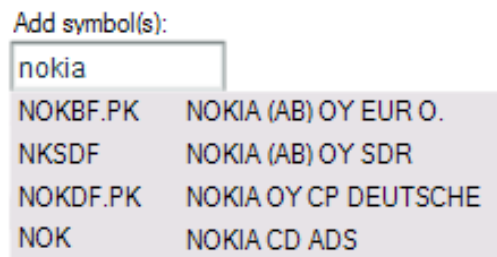


Figure 71. Add symbol in the concept.

In the second control in the legend you can change the *time range* for the chart. The chart can display the data for several different time ranges, such as 30 minutes, 1 hour, 1 day, 3 days, 5 days, 1 month, 1 year etc. During our pre-study we identified that the user often want to look at different time periods when analyzing the development over time for a stock.

The second dropdown in the legend is used to change the way the information is presented; the *type* of the chart. Most of the users like to display this kind of data with a line chart; however, we have also seen the some users prefer other techniques to display the same data. Within the stock market candlesticks and bars showing OHLC, open/high/ low/close, are also often used to display the same information as in the line chart. Since the users are accustomed to being able to choose how they want the information to be displayed, and also since the users prefer different ways to display the data, it is possible to choose what type of technique to use in the visualization. Here you can choose between displaying a line, candlesticks, or OHLC bars. It is, according to Nielsen [11], important to provide the user with concepts that he/she is familiar with.

The next choice you can make in the legend concerns the *cursor* in the visualization. When looking at a line chart it can be hard to distinguish the exact values displayed. The use of a cursor can help the user do this. You can choose to display a track ball, a crosshair, or no cursor at all. The track ball consists of a dot that follows the line when you move the cursor over the visualization. The crosshair is in the form of a cross which also follows the line (see figure 72). With the help of these two cursors the user can more easily see what value an exact point on the line has. To make this even clearer, dynamic text has also been added at the top of the line chart. This text is updated along with the movement of the cursor and thereby the exact values in time are

RESULT

displayed. The last value, the real-time value, is also displayed here. This provides the user with direct feedback, which is very important (Cooper & Reimann, 2003).

During our pre-study we identified that the users perform analysis using different *technical indicators*. In the legend it is therefore possible to add technical indicators to the chart. They are then added at the bottom of the chart and displayed in smaller graphs. Volume is a technical indicator that is added by default to the chart. You can choose what technical indicators to display in the dropdown menu. Since it should be possible to show several different indicators for the same chart, checkboxes are placed within the dropdown-list.

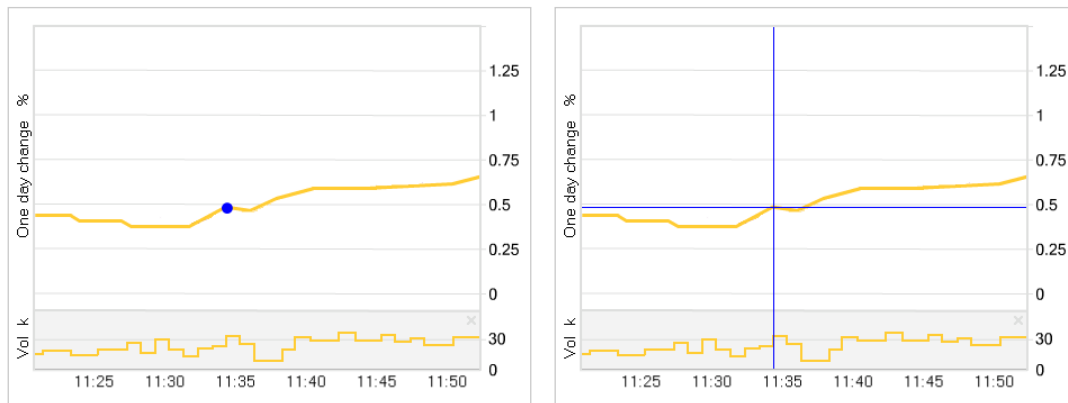


Figure 72. Cursors for the line chart.
The left image shows the track ball and the right the crosshair.

The fourth dropdown in the legend allows the user to change between a linear and logarithmic *scale* on the y-axis. Wallgren et al. (1996) mean that a logarithmic scale can be very suitable when studying percentage changes, since they display relative change. However, when looking at prices it is more suitable to use a linear scale, since it displays absolute changes.

The last dropdown in the legend is used to specify how the lines in the chart should be colored. It is most common to *color* the lines according to ticker, but the user can also choose to color the lines by for example sector or type. The lines displayed in the visualization are listed below the last dropdown. Color is used to relate the information displayed with the data in the visualization. The user can thereby easily see what is displayed in the chart. To provide the user with a simple mean to choose what information that is to be displayed in the chart, checkboxes have been placed in front of each color and ticker. Since the users are very busy, and work in a hectic environment, it becomes extra important to provide the users with the ability to manipulate the content displayed in each visualization in an easy and immediate way. If the user for example wants to remove a ticker altogether from the line chart you can right-click the line or the ticker and choose to remove the line.

7.2.4.3 Scatter plot

The scatter plot in the concept is to display performance compared to a benchmark. This visualization is displayed in real-time. Various settings can be made for the scatter plot in the legend to the right of the visualization.

The *legend* for the scatter plot consists of four dropdowns, one color legend and a checkbox for displaying trails in the plot. In the first dropdown in the legend you can choose the *time period* for the visualization. You can choose to display the performance for 1 day, 1 month, 1 year etc. In the second dropdown you can choose what *benchmark* the data should be compared to. The users within our user group all use different benchmarks and they sometimes also create their own benchmarks to make comparisons with. What the *size* of each object in the scatter plot should represent can also be selected in the legend. By letting the size represent a variable, a third dimension can be added to the scatter plot (Wallgren et al., 1996).

In the last dropdown in the legend, settings concerning the *coloring* of the scatter plot can be made. You can choose to color the objects in the scatter plot by for example stock, sector, industry, turnover, return, capital, or price. This can add yet another dimension to the visualization. Below the last dropdown, a color legend is presented. Color is used in this legend to relate the information displayed with the data in the visualization. The color legend informs the user about what is displayed in the chart. To provide the user with a simple mean to choose what information that is to be displayed in the chart checkboxes have also been placed in front of each color and ticker. With the help of these checkboxes the user can easily hide objects in the visualization. It is important to provide the users with the ability to manipulate the content displayed in each visualization in an easy and immediate way.

Since the users are very interested in development over time and the scatter plot is not good at displaying this as it is, we added a feature to the scatter plot. This feature enables the scatter plot to display change through time with the help of *trails* on the objects in the visualization. To enable/disable the feature the user only has to select/deselect the checkbox below the color legend. The trails show history from the beginning of the time period chosen for the plot. If the scatter plot displays over/under benchmark for the last month, the trails will consequently be displayed for a month back in time. By providing this feature the users can perform analysis which otherwise would not be possible, such as analysis of the magnitude, direction, and velocity of change for the values over time (Few, 2007).

The interaction in the scatter plot is the same as in the other visualizations in our concept and also as in TIBCO Spotfire. Objects are highlighted with a black frame on mouse-over, and colored blue when selected. You should also be able to select or highlight items in the visualization by clicking the color legend.

7.2.4.4 Bar chart

The bar chart is to display relative performance. The visualization is displayed in real-time, but since it displays summative values (information which does not fluctuate heavily) the movements of the bar charts are hardly visible. It could be questioned whether or not this information needs to be in real-time since it does not change heavily. It could as well be sufficient enough to update the information for example every 20 minutes. The user should be able to choose what information that shall be displayed in the bar chart. This can be done in the legend belonging to the bar chart. Here you can choose between which data the relative performance should be displayed. In the stock view you can for example choose to show the relative performance between the stock and several different kinds of indices, sectors and

other stocks. The user can also choose the time period for the chart, if it is to display the relative performance during one day, month, year, etc.

In our concept we have not explored the bar chart to a wider extent, since this is not one of the most important plots in our concept. We do believe it is important to explore the real-time aspect of the bar chart further. In this visualization we have noticed that it becomes more important to show how the bars have changed over a long time than during the most recent time period, since the movement is very subtle. We have a notion that this could be done with the help of some sort of shading showing the development of the bars.

7.2.4.5 Sparklines

In our concept sparklines are used frequently. Sparklines give a good overview of trends and development over time (Tufté, 2006). We believe that sparklines can be useful as a complement to the other visualizations in our design, mostly the scatter plot and the bar chart. These visualizations can have problems with displaying the data's development over time. By combining scatter plots and bar charts with sparklines, both overview and detail can be provided. The data can be put in a wider context, which is very important in a dashboard design (Few, 2006). See figure 73 for an example of how sparklines can be used in combination with a scatter plot.

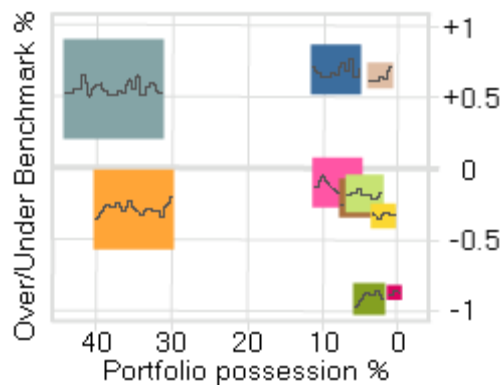


Figure 73. Sparklines displayed in a scatter plot.

Sparklines could also be displayed in the best/worst table to provide an overview of the otherwise detailed data in the table. To use sparklines in the miniature field would also be ideal, since the miniatures only are to provide an overview of the data and by using sparklines a large amount of data could be fitted in the miniature field. We believe that sparklines have great potential for dashboard design, since they are so space-efficient, and that this display medium need to be further explored to be able to find all the possibilities for this technique in our design.

7.2.4.6 Events

The events are updated in real-time. The table of events is sorted to display the most recent event at the top of the list, since this is the most important information. To facilitate for a more rapid perception for the user, the company names and tickers are displayed with bold text in the events table. The events are also displayed on the most prominent visualization in each view. The events can thereby be connected to the data, and it is easy to see how and if a certain event affects the data. During one of our user tests it emerged that it was important to be able to see all events that has

happened during the day. All events for the day are therefore displayed on the visualization.

The events are divided into different kinds of categories to make it easier to distinguish between the different events. Each type of event is represented by an icon. Different generic shapes are used for the *event icons* to differentiate the data from each other (Few, 2006). The icons are placed on the visualizations to relate the events to the data in the plots. These icons have been designed to be as simple and stylized as possible in the concept, by for example only using a grayish general outline for the icons and by avoiding color (see Figure 74). When designing a dashboard it is very important not to overuse color; color should only be used when it adds something to the visualization (Few, 2006). During the last expert test we got a lot of feedback on the design of the icons we had developed for the prototype. These icons were perceived to be too dominant and spacious. We therefore decided to tone down the icons by making them smaller and avoiding colors, since the concept otherwise is very dominated by visualizations and the use of color. The icons also needed to be small to be able to fit in the cluttered treemaps. Cooper and Reimann (2003) also mean that icons should be kept simple; you should avoid excessive visual detail, minimize the number of colors, and retain a modest size.

To be able to show relation between the information in the visualizations and the table of events, highlights and selections are used. If the user mouse-over one of the events in the table it and also the stock in the visualizations connected to this event will be highlighted with a black frame. If the user clicks on an event in the table it will be selected and a blue frame will be displayed around it. The stock and the icon connected to this event will also be colored blue in the visualizations (see Figure 74). By using the same interaction technique for highlighting and selecting items in the event table as in the other visualizations a consistency can be maintained. Through the use of the same shape and color for the highlights and selections in different parts of the design it also becomes easy for the user to relate the information with one another (Few, 2006).

It should also be possible to choose what events that should be displayed. This can be done in the *legend*. This legend is placed at the bottom of the event table since it was the most space efficient (see Figure 74). The pre-study showed that the users look at very different kinds of information. During our pre-study, we identified three main types of information that the users look at. These are: news, reports, and insider activity. Some of the users looked at all of these different kinds of information, while others only looked at for example news and reports, but did not take notice of insider activity at all. We therefore believe it to be very important to provide the users with the ability to filter out data in the events table and only display the information they primarily look at. The checkboxes in the legend can therefore be used to choose what news to display in the events table and in the visualizations. There is also a way to show and hide only the events displayed on each prominent visualization in the design. This option is given since the treemap can become very cluttered if it is to display all the news during the day. An expandable checkbox menu is therefore placed under the legend in the largest visualization in the design (see Figure 74). In this menu you can choose what kind of events you want to display in the visualization.

RESULT

During our final user test we got positive feedback when it came to displaying the events and linking them to the visualizations. However, the users also had a desire that the events displayed should not only contain the categories we had created. Today the users need to gather information about events from several different sources, such as websites, their information system, and the banks own system. In our prototype, the user was only provided with the information the users usually get from websites and their information system. If this analytics tool could also be connected to the bank's own system and retrieve the bank's internal reports and statements, this would save a lot of time for the users. To get these statements displayed together with the other events would facilitate the analysis process for the users.

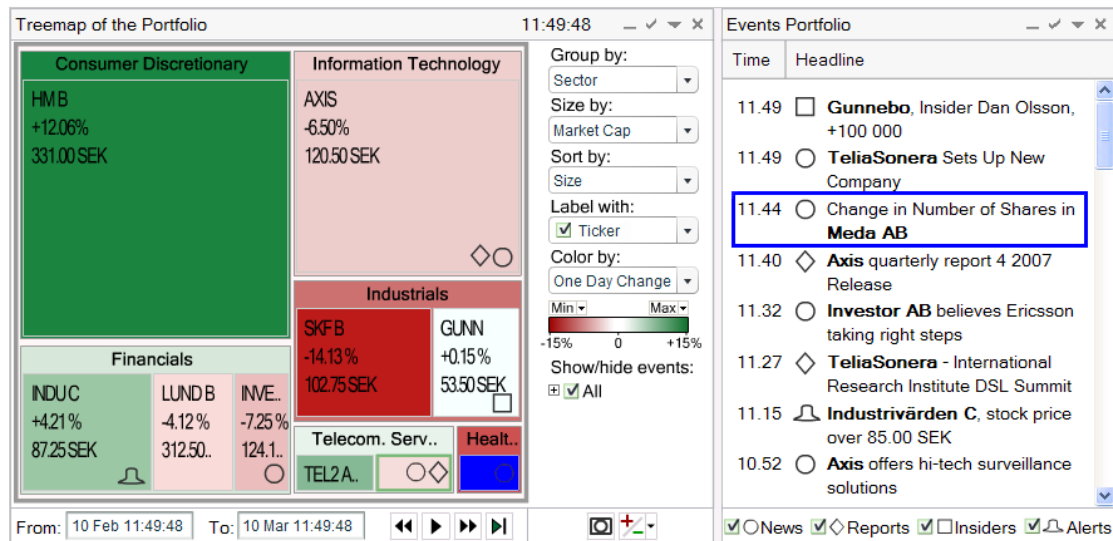


Figure 74. Events in the concept.

The figure displays the event table and the respective events placed on the treemap. The event concerning Meda A has been selected and is marked with a blue frame in the table and the stock has also been marked in the treemap.

7.2.4.7 Best/Worst

This table provides a list of the best and the worst performers. It is divided into two lists by using two different tabs within the window, displaying the best performers on one tab, and the worst performers on the other. The purpose for displaying this list is to provide the users with more detailed information, as well as an easy mean to *identify extremes*, since the pre-study showed that the users are very interested in extremes. The visualization also provides the user with a mean to navigate to other views. Tickers, company names, and portfolio names are all clickable in the list. If the user clicks either one of these they will navigate away from the current view, and open up another view containing information about the chosen data. This can also be done by right-clicking one of the links.

We have decided to change the appearance of the data that is clickable so that the user easily can distinguish between this and other data in the table, and also be aware of what will happen when they click on these places. The clickable texts are underlined and colored blue. This appearance was chosen since this is a commonly used appearance for links on the web. By using a concept that is familiar to the users Nielsen [11]

means that they can easily understand what will happen; it will appear logical to the users.

It should also be possible to choose the appearance and also the content of the best/worst table. This can be done in the legend. Here the user can choose which columns that are to be displayed, how the list should be sorted, and also for which time period the data in the table should be displayed.

7.2.5 Features

In this section the different features, which the concept contains, are described. These are: *alerts*, *minimum/maximum*, *time controls*, *duplicate*, *miniatures*, *a content control*, *details-on-demand*, and *filters*.

7.2.5.1 Alert

In the pre-study we identified that the users monitor large amounts of data, such as stock price, and need to be aware of when for example a stock price descends below a certain value. It therefore becomes very important and useful for this user group to be able to set alerts. By setting alerts the user does not have to monitor all activities, since the user will be alerted when something of interest happens.

Since it should be possible to set alerts in all of the views in the design, and also be visible at all time, independent of what the user is doing at the moment, the alert area is placed in the top right corner of the framework. The placement of this feature in the framework results in that it is always visible and accessible for the user. The alert area consists of a button, which is used to set alerts, and information concerning the alerts.

If no alerts are set, the information displayed in the alert area makes the user aware of this. To *set an alert*, you can either right-click in the visualization, click the alert button in the upper right corner, or take hold of the button and drag it and drop it on the visualization. During the final evaluation many of the respondents tried to set and wanted to set, an alert for a stock using right-click in the visualization. It was not possible in the prototype, but it should be possible in the concept. Providing the user with the possibility to also use drag-and-drop makes it very efficient to set an alert since drag-and-drop provides for a direct manipulation, and it also becomes very easy for the user to understand what his/her action will result in (Cooper & Reimann, 2003). A button for setting alerts is also given in the design since we wanted to provide a visible alternative for the user to do this. The icon on this button is designed as a bell. By using the bell as a metaphor it will make it easier for the user to understand what this button does. Cooper and Reimann (2003) mean that metaphors are understood intuitively and are useful for representing the purpose of for example a button in a toolbar. The button is also designed to let the user know what will happen when it is pushed. A small arrow is placed to the right to show that when the button is clicked a dialogue box will be opened, in which you are to set the alert. This dialogue box will also appear if you use right-click or drag-and-drop to set the alert.

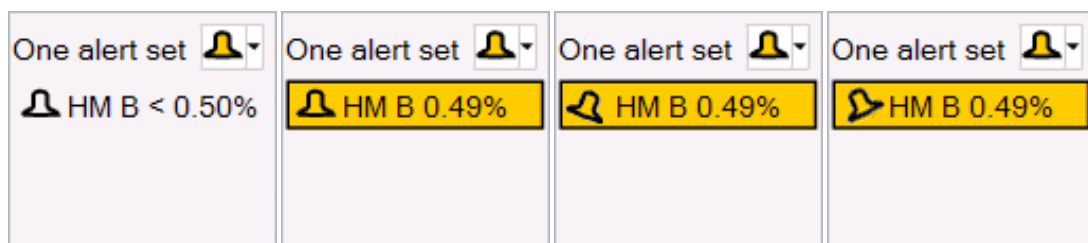
In the dialogue box the user can choose what to set an alert for. The dialogue box consists of four controls that are used to specify the values that should trigger the alert. At first you choose what you want to set the alert for by typing for example a ticker, company name, portfolio name, sector in the first text box. When the letters are typed in this text box a list will be displayed containing the items matching you

RESULT

search. This will make it easier for the user to find what he/she is looking for. In the second control, a dropdown, a choice of what the alert concerns is made. Here you can choose to set an alert for the price, change in percentages for different time periods, risk, P/E ratio etc. After making this choice the user chooses if the alert is to be triggered if the value go below, over, is equal to, go below or is equal to, as well as go over or is equal to. Finally the user types in the value which the alert is to be triggered for. If the user wants to set several alerts, another row can be added by clicking the button with a plus icon to the far right in the dialogue box.

When designing this dialogue box it was very important to make it as simple and easy to use as possible, since we had noticed that the users in our user group did not use features in their current software if they were to complicated and time-consuming to use. We therefore chose to design the settings for the alert by using sentences, since we thought it would make it easier for the user to understand what to choose and type in each control. Our final evaluation showed that neither the users nor the experts, who had very little domain knowledge, had any problems with setting an alert. Only minor changes were therefore made to the appearance of the dialogue box in the concept, such as adding headlines for the controls to make it easier for the user to know what to type in.

When an alert has been set, it will be visible in the alert area in the upper right corner of the interface to inform the user of how many and which alerts that are active at the moment, see figure 75. The alert is displayed in this way in this area until an alert is triggered. When an alert is *triggered* a frame will be displayed around the alert, the background in the frame becomes colored in the same yellow color as the icon on the button, and the bell starts to swing from left to right, see figure 75. All of these different ways to catch the users attention is used since it is very important that the user notices the alert when it is triggered. Color, motion and flickering are all powerful attention-getters according to Few (2006). The alert will continue to try and capture the user's attention until the alert is deactivated, which is done by clicking on the alert. The user can then choose to display information connected to the alert, to remove the alert, or reset the alert. When an alert is triggered this is also displayed as an event in the events table and on the visualizations. To display the alert as an event in the table and on the visualizations connects the alert with the information it concerns. This will make the user even more aware of the fact that the alert has been triggered.



*Figure 75. A triggered alert in the concept.
The alert that is set is triggered, which is shown with a black frame, a yellow background and a movement of the bell from left to right.*

7.2.5.2 Time controls

We identified a need for the user group to be able to move back and forth in time. They frequently compare different time periods to each other, trying to find trends and patterns in the data. The users also need to be able to see historic data up to 10 years back in time. By providing a feature which the users can use to move back and forth in time they can easily look for trends and patterns over time in the data. We believe that this feature is very useful for this particular user group when analyzing data.

Since we consider this feature to be one of the more useful and important ones, we wanted to provide the user with several different ways to go back and forth in time. You can go back and forth in time in six different ways in this concept. You can use *drag-and-drop* in the graph, you can use the *arrows* that appear in the graph when you mouse-over the far left or right area of the graph; the *miniature graph* (1), the *scrollbar* (2), the *date and time field* (3) and the *buttons* (4) can also be used to do this (see Figure 76).

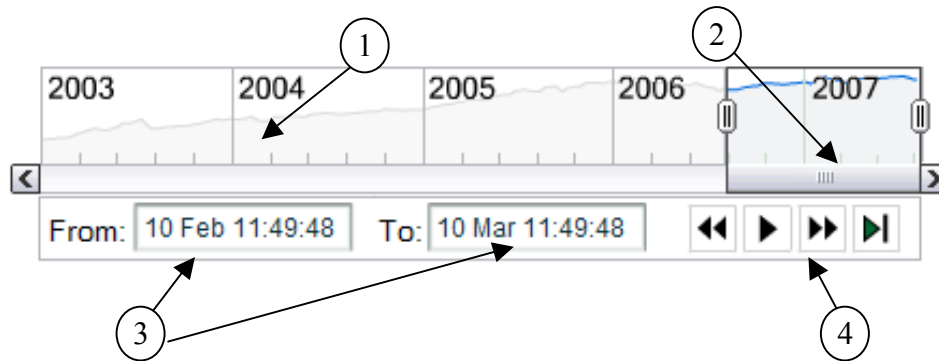


Figure 76. The time controls in the concept.

When we performed our final evaluation the most preferred way to go back and forth in time, both by the experts and the users, were to “grab” the graph and drag it to the desired place in time. This is very useful if you only are to move a minor distance in time. This feature is however not immediately visible for the user, but need to be tried out to know it is there. To provide the user with a more visible way to move shorter periods in time the design also comprise rewind and forward buttons along with play/pause and a button for going back to real-time.

To move longer distances in time the user can instead use the miniature chart, the scrollbar, or the date and time fields. The miniature graph provides both a way to move in time and a good overview of the development over a long time. This miniature graph is only displayed for line charts, since these charts are the only to show linear development over time. The miniature graph can be used to find a particular period in time of interest, for example where the line peaks or dips. The miniature graph is thereby very useful when looking for extreme values. A rectangle can be drawn in the miniature graph and the area within the drawn rectangle will then be displayed in the large line chart. The scrollbar is connected to the miniature graph and can also be used to move in time and change the displayed time period.

The text fields positioned in the toolbar display between which dates and times the data in the line chart are presented. These fields are updated dynamically as soon as the user moves back and forth in time to provide more detailed information about the

RESULT

time period in the graph. The text fields can be used if the user has a specific date and time which he/she wants to look at. To also provide a way to manipulate the graph directly to move longer distances in time, arrows are displayed in the graph when respective side of the graph is moused-over.

To go back and forth in time has proven to be the most important for the line chart, since it is very good at displaying change over time. This feature has therefore mostly been developed for the line chart in this concept. The feature would however be able to work in a similar way for the other visualizations, but display different points in time rather than a continuous development. When the user goes back and forth in time in the line chart the graph is divided into two different parts, see figure 77.

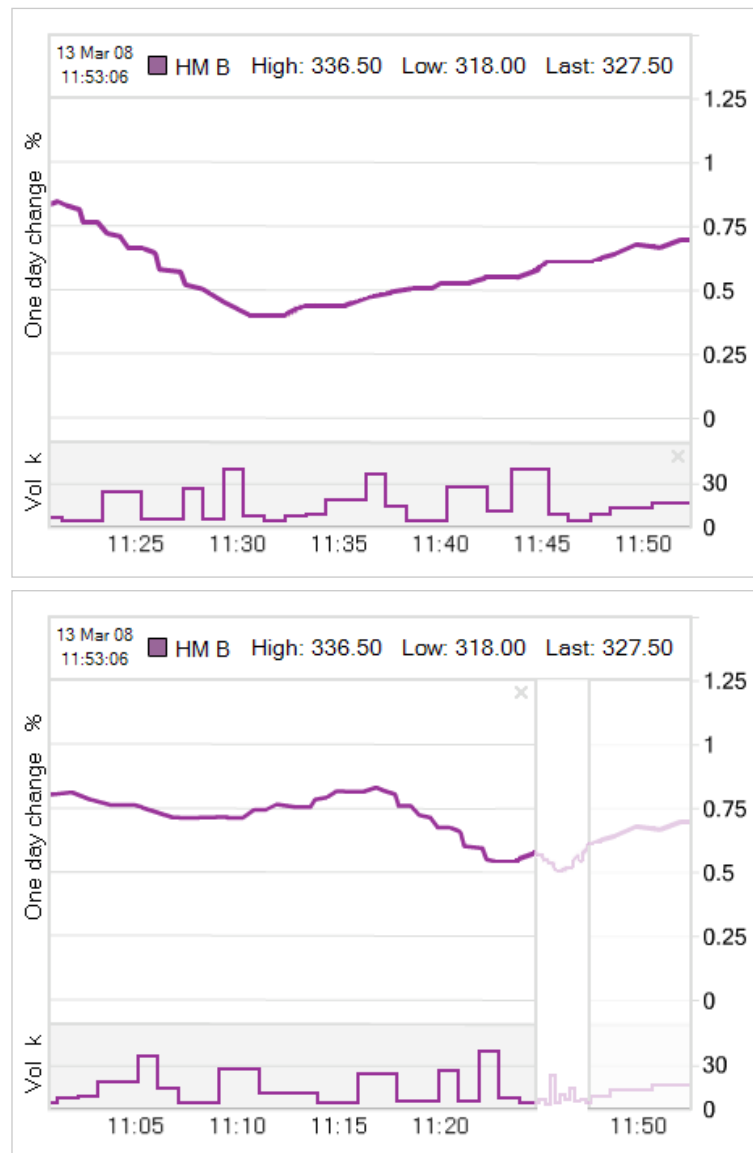


Figure 77. A rewinded line chart in the concept. The upper graph displays the line chart before going back in time. The lower graph displays the line chart when rewinded about 30 minutes back in time.

The lower graph still displays real-time data, since it has proven to be very important for the users to be able to monitor real-time data at all times. This part is however

grayed down to make it clear for the user that this area cannot be manipulated at the moment. By using another fill color the real-time data is distinguished from the other information in an effective way (Few, 2006). The left part of the graph displays static data for the time which the user has chosen to go back to. The period between the static and the real-time graph is displayed with the help of a sparkline. This is to make it clear for the user what the two parts of the graph are displaying and where in time they are placed. The user can always go back to displaying real-time data by either shutting down the static part of the graph by clicking on “X” sign in the upper right corner of the static part of the graph, or by going forward to real-time with any of the controls. Frames are used to separate the static part of the graph from the real-time part. This is since it should be obvious what the different parts of the graph are displaying. Enclosure is a good way of separating objects from the rest of the information (Few, 2006).

7.2.5.3 Duplicate

During the pre-study conducted we noticed that the users compare different time periods of the data to each other frequently. We thus wanted to offer the users a feature to help them do this in an easy way; the duplicate feature was therefore added. This feature allows the user to make an exact copy of a visualization. This allows the user to make a duplicate and then manipulate the new visualization and then be able to compare the two visualizations with each other.

A duplicate can be made in two ways, either by clicking the button in the toolbar for each visualization or by right-clicking in the visualization. The button used for making a duplicate contains an icon of a camera. This metaphor is used to make it easy for the users to understand the purpose and function of the button (Cooper & Reimann, 2003). We had some concerns that this feature would be surprising for the user since when the button is pushed the duplicate is made instantly. However, the final evaluation showed that this was not a problem, the users rather believed it to be good that the action occurred immediately since they in this way did not need to go through a time-consuming second step to perform the action.

The area of the visualization which the duplicate is made of is divided to fit the visualizations. This is to maintain the structure in each view even when a visualization is added using the duplicate feature. The users most often do not have the use of making more than two duplicates, whereas the area can fit the visualizations. The visualizations are placed in the same view beneath each other to facilitate comparison.

7.2.5.4 Minimum/ Maximum

This feature allows the users to distinguish information of greater importance in the visualizations. This becomes extra important in the cluttered treemaps since it can be hard to find the information you are seeking in these visualizations. The users look very much at extremes and make a lot of important decisions based on this information. This feature thereby lets the user mark the minimum and maximum values, that is, the extremes, in the visualizations.

You can either right-click in a visualization, or use the button in the toolbar to mark minimum/maximum values. The button is designed using a plus and a minus sign to make the user aware of that this feature is used to identify minimum/maximum

values. The button is also designed to let the user know what will happen when it is pushed. A little arrow is placed to the right to show that when the button is clicked a dialogue box will open, in which you are to make further settings for the feature. This dialogue box will also appear if you use right-click or drag-and-drop to set an alert.

When designing the dialogue box for this feature, it was very important to make it as simple and easy to use as possible, since we had noticed that the users in our user group did not use features in their current software if they were too complicated and time-consuming to use. We therefore chose to design the settings for this feature by using sentences, since we thought it would make it easier for the user to understand what to choose and type in each control. Our final evaluation showed that neither the users nor the experts, who had very little domain knowledge, had any problems with using this feature. Only minor changes were therefore made to the appearance of the dialogue box in the concept, such as adding headlines for the controls to make it easier for the user to know what to type in.

The dialogue box contains five different controls, in which you are to set different values. In the first control, a spinner, the user can choose how many values that are to be marked on the visualization, for example the 5 lowest values. In the next control, which is a dropdown the user can choose what type of data that is to be marked, such as stocks, portfolios, sectors etc. then you have to choose whether to show the highest or the lowest values. The next dropdown is used to choose what kinds of values are to be marked, for example those with the highest/lowest price, market cap, increase, decrease, volatility etc. The last dropdown is used to choose for which time period the values should be displayed. When these choices have been made markings are showed on the treemap according to the settings made by the user. The markings are updated dynamically along with the real-time updates. To make it easy for the user to hide the markings on the visualization if it becomes too cluttered, checkboxes are displayed in front of the explanations given below the legend.

7.2.5.5 Content control

When performing the final evaluation a request was given to have a more accessible way to change between for example different stocks. Therefore a control was added in the framework where the user can choose what to show in each view. The control is a text box where the user can type in what is to be shown. When the letters are typed in this text box, a list will display the items matching you search, see figure 78. This will make it easy for the user to find what he/she is looking for. In the market overview you can choose to display different markets, such as Large Cap Stockholm, Mid Cap Stockholm, and Small Cap Stockholm. In the portfolio overview you can choose to display different managers' portfolios. In the portfolio view different portfolios can be displayed and in the stock view different stocks.

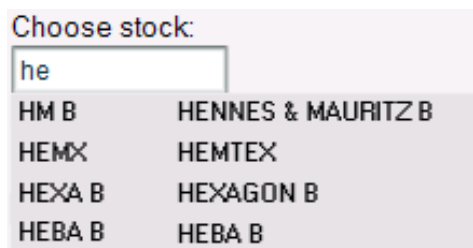


Figure 78. Control for changing the content in the view.

7.2.5.6 Miniatures

The miniature field is positioned to the right in the design. In this field the user can add different visualizations, which they want to be able to monitor all the time. This feature was found to be important since the users in our user group needs to be able to monitor several different things at the same time.

Miniatures can be added by grabbing a visualization on the visualization area and dragging it to the miniature field, or by right-clicking a visualization. By using drag-and-drop it becomes very easy and efficient for the user to add visualizations to the miniature field. Cooper and Reimann (2003) promote drag-and-drop since it is a very natural and easy to understand interaction technique.

The miniatures do not only provide a way to be able to monitor several different things at the same time, but also a means for navigation and a way to save the settings of a visualization and a view. When a miniature is added to the miniature field the settings for the visualization and the view the visualization was taken from are saved and can be returned to at a later stage by right-clicking the miniature.

7.2.6 Details-on-demand and filter panel

Both the details-on-demand window and the filter panel work in the same way in our design as in TIBCO Spotfire today, since we have not had time to explore these features very much as they are very extensive. The *details-on-demand* window provides detailed information that is associated with the selected items in the visualization. The filter panel is used to select what you want to look at in the data set. The filter panel is however hidden in our design using a tab to the far right in the interface, since we have identified that our user group does not want to filter out data in many situations. We do thereby believe that this feature does not need to be visible at all times, since it would not be used very often.

Instead of *filtering* out data the users seem to have more use for searching and finding specific information in the visualizations. The users need to see the data in its context to be able to make comparisons and identify correlations, and are consequently not interested in removing data from the visualizations. We have thereby had some thoughts of providing the users with a similar feature as the filter panel, but instead of filtering out data in the visualizations, the data should be marked in the visualizations to make it easier for the users to find the information they are seeking. This becomes extra important for the treemaps, where large amounts of data are displayed in a very small area of space, whereas it can become very hard to find the data you are seeking as the visualizations become very cluttered. The users in the user group we have studied do furthermore not seem to want to filter/search all the visualizations in the view at the same time, but rather be able to filter only one or some of the visualizations in the view simultaneously. However, we have not had the opportunity to explore these ideas in our thesis work and do thereby believe that they need to be further investigated.

8 Discussion

In this chapter the thesis work is discussed from two viewpoints: the realization and the results. We discuss what has worked well and what has not, what could have been done different and why we have been forced to deviate from our initial plan. A discussion of the user group and their situation, the demonstrator and the concept is also conducted. The discussion leads up to design recommendations for visualization of real-time data in general and for the specific domain and user group. We conclude this chapter with discussing possible future work.

8.1 Realization

The plan we made at the beginning of this thesis work turned out to be unfeasible in many ways for us to follow and we were forced to revise our plan several times during the work. We do not see this as a failure since it was partly due to circumstances out of our control, and partly it was due to discoveries along the way which made it necessary to rearrange our plans. Even if it is almost impossible to completely follow an initial plan when performing design work we have recognized the importance of making *plans* and of setting up *sub goals* during the work. By planning the work and setting up sub goals beforehand you can easily verify if the work is going according to plan or if you need to revise it. For example the requirement analysis we performed and the prioritizing list we used when designing the prototype was two successful examples of this during our work. We planned to follow an iterative model in our work. We were able to stay with an iterative work process even though our work parted from the initial plan. This model turned out to be well suited for our study, since it enabled us to go back and modify and redesign parts of the interface.

Our initial plan was to start with a *literature study* prior to meeting users. The literature study was however interrupted in advance since we got an opportunity to meet a user. At the time it sounded like a great opportunity but afterwards it proved to be too early in the process. We did not have enough domain knowledge and did thereby not know what questions to ask. Furthermore the participant also turned out not to be in our target user group. Consequently, we spent time on preparing for an interview that was not that rewarding for us in the end.

When preparing for the interviews performed throughout the work we have become very aware of that it takes a long time to develop *interview guides* or even to just adjust or rearrange them. When performing the interviews we also think that making recordings would have been very good, since it would have enabled us to concentrate more on the interview situation than having to focus both on interviewing and getting everything down on paper. All our interviews were semi-structured and we think that this was the right choice, since the respondents had no problems with answering the questions and we were able to collect the information needed.

For two of the interviews we used *mediating tools* to trigger the participants to talk during the interview (Karlsson, 2007), which turned out to be good in one case and not so good in the other. At the first bank we showed the participant images of dif-

ferent ways to visualize financial data and this did trigger our respondent to talk. At the second bank we wanted more specific information about the respondent's assignments and work flow and used different scenarios to get the respondent to discuss these issues. This was not as successful, it did not trigger the respondent to talk, but rather he had problems explaining what he would do. We think it is because the respondent would have needed to sit in front of a computer in the real work environment to be able to answer the questions we had about each scenarios. This confirms our thoughts concerning that we needed to conduct a contextual inquiry, which also was our initial plan. This was however not possible, due to secrecy issues at the banks. We think that had we been able to conduct contextual inquiries, or any kind of observation of the users' work situation, we would have gotten a much better understanding of the situation which would have made it easier to continue our work. We would also have been more certain of the validity of the results than we are now. Performing interviews instead of contextual inquiries we do however believe was the second best method to use.

During our design process we have gone back to *low-fidelity prototyping*, i.e. paper prototyping, several times. This is something we believe has been very rewarding since it allowed us to reflect and redesign. To develop paper prototypes of the design was less time-consuming than if we would have developed digital sketches or more advanced prototypes. By iterating in this way we believe we have thought through the different parts of the design better than we would have if we had passed the low-fidelity stages heading right for the high-fidelity stages.

Adobe Flash was used to design our *high-fidelity prototypes* and we are confident that this was the best tool, since we needed to be able to animate the data. One problem we encountered with Flash was however that it took longer time to learn the software than we had anticipated. It was therefore very good that we came to realize that we needed to start with Flash before we initially had planned to. Another problem we encountered was that we did not have access to any actual data when creating the prototype; we therefore were forced to fake the data which was very time-consuming. We wanted the prototype to be as realistic as possible and therefore needed to check all values and data we were to use in the prototype. For example, to create realistic treemaps of the stock market proved to be very difficult and time-consuming. If we would have had access to financial real-time data it would have made things easier.

By developing high-fidelity prototypes in *Flash*, it was easy to test how the real-time aspect would affect the visualizations. This led us to an important insight that we would not have been able to draw if continuing with low-fidelity prototypes, namely that real-time did not necessarily mean that things moved fast. The fact that it was difficult to see that the charts' content were moving was a very important insight at this point, since it affected both the visualization and interaction with the real-time data.

When designing we believe that spending so much time on developing the prototype caused the concept to suffer. We feel that we should have allocated more time for developing the concept during the process, especially after the final evaluation. We concentrated on building a prototype that would work. Since many things in our concept were not possible to implement in the prototype due to limited time and lack of knowledge in Flash, the prototype does not reflect our vision of the concept. We have had to make many compromises in the prototype just to be able to test some

functionality. This means that instead of focusing on how we really would want the concept to work and look we have focused on what we could do in the prototype.

We believe that the *methods* chosen for evaluation throughout the process were well-suited, since the evaluations all proved to be very successful. We had some thoughts of performing a heuristic evaluation instead of a cognitive walkthrough in the last evaluation. We do however believe that we made the right choice since it was very rewarding to be able to have a discussion with the participants during the tests, which the cognitive walkthrough enabled us to have.

One challenge throughout the thesis work has been the *limited access to users*, which has been problematic for many reasons. The users are very busy and have little time to participate in studies, and therefore it was not possible to find as many participants as we would have wanted. We have however tried to balance this by combining user tests with expert evaluations, which proved to be very rewarding. By combining techniques in this way, to triangulate, you can receive a higher validity to the results (Silverman, 2005). In our expert evaluations we have also tried to involve respondents with a widespread knowledge. We believe that using this combination of people was very successful, since they came with very varied insights.

One reflection from the evaluations with users and experts is that the experts were much more talkative and could express their thoughts much more freely and extensively than the users. This was not surprising considering the experts are used to discussing and evaluating interfaces in their work, which the users on the other hand have no experience of. Considering this, it was even more important that we involved both users and experts in the evaluations.

As with the interviews, we think that it would have been good to *record* the evaluations, preferable on video, to be able to go back and analyze the tests afterward. However, this was not possible because of confidentiality at the banks.

We have become aware that the conditions on the financial markets abroad may be somewhat different compared to the Swedish financial market, whereas the user groups and situations we have identified may be very specific for the Swedish financial market; e.g. there seem to be a *difference in the professional roles and the work situation* on markets in foreign countries compared to the Swedish financial market. Our results may thereby not be as applicable to financial markets abroad as it is for the Swedish financial market. The work situation might even differ within this country for the asset managers and stock brokers as a result of the size of the establishments. From our interviews in Göteborg we have learned that in the small office in Göteborg they need to manage on their own a bit more than the asset managers and stock brokers situated in Stockholm.

Since we have only had the possibility to explore users situated in Göteborg, an important user group, the analysts, might also have been overlooked. The analysts working at the major banks in Sweden are all situated in Stockholm, which made it unfeasible for us to explore this user group. As the analysts are the ones providing the asset managers and stock brokers with more thorough analysis of the financial data this user group might therefore have a huge advantage of having access to a visual analytics tool providing them with real-time data. However, this user group does most

likely not need to look at the same information as the asset managers and the stock brokers; for example, the analysts do not manage any portfolios and do thus not need to be supplied with aggregated information concerning portfolios, but rather only with information concerning the market.

We think that our process in this work has been too extensive and that we should have been better at making *delimitations*. Initially we thought it would be possible to deliver a fully interactive prototype but it turned out to be impossible in the time we had. Since this was our vision we had some problems with accepting changes that resulted in our final prototype becoming less interactive than we had initially wanted. By limiting the amount of scenarios to design for, the number of design alternatives developed throughout the work, and by focusing on parts of the design instead of an entire interface, we think that we would have been able to delimit our work to a more reasonable extent.

8.2 Result

We believe there is a great potential for a real-time analytics tool within the stock market, since we have noticed in our studies that the professionals in this market do not use or have access to any software of this kind today. They handle large amounts of data, data that is rarely visualized, but remains to be presented in tables; something which Merino et al. (2006) also confirms. This domain would therefore have a lot to gain from having access to visualizations of the large amounts of data. It would provide them with an overview of the data and easy means of comparing the data and find patterns and trends, which is very important within the stock market.

The data and information that proved to be the most important to display for the users consisted of both *overview and details*. They need to be able to analyze the data on different levels of detail. The layout we have chosen, consisting of different views, providing both overview and detail, therefore was a good setup. By having this setup the same structure could also be kept in all of the views, making it easier for the user to recognize and find the data in each view. To be consistent in the design is important according to Nielsen [11] and it also proved to be important for the users. They did not want any drastic changes in the position of the data as it made it hard to keep track of the data and find the information. To provide an overview of, and a context for, the data is something we believe the users can have very much use of. This is something they lack totally today since their systems provide information in tables, which makes it very hard to see the context for the data and get an overview of the data.

Providing the information in different views did not seem to be an issue since the users did not *compare information* on different levels of detail with each other. However, it could be useful to provide the user with an easier way to compare data on the same level with each other. If you want to compare for example two portfolios with each other there should be a way to do this in the same view, without having to change between pages since this makes it hard to compare the data. When it comes to correlation of the data some could therefore be gained from developing the concept further to contain a comparison view which could be used to compare data on the same level of detail more closely.

It is also important to provide the users with a way to spot *extremes* in the data. By using visualizations, such as the treemap, this proved to be a lot easier for the users to do using our prototype than in their current software where this information is displayed in tables. Few (2006) means that this is one of the strengths with the treemap, i.e. being able to spot conditions that can be of interest, most typically the extreme values. To provide the users with information both of the market, as we do when providing an overview of the market and in the stock view, as well as information concerning the portfolios they manage is very important. The users were very pleased with having this information in the same system, something they do not have today. This facilitates performing analysis and seeing patterns and trends in the data.

During our pre-study and also during the evaluations we have noticed that the professionals within the stock market are very fixed on how their current software works. They are accustomed to being provided with the information in tables and have some trouble with accepting that this might not be the best way to display this information. During the interviews the users often referred to how their current software work and when faced with the visualizations some of the users at first had some trouble with understanding and accepting this way of presenting the information. We therefore believe it to be very important not to deviate completely from the way they are used to looking at the information. A real-time visual analytics tool does not solely need to contain charts of different types, but also need to provide information in tables as these can provide more detailed information than can the visualizations (Wallgren et al., 1996). The users also need to be allowed to have a period of learning, since this concept differs a lot from how they are used to looking at the information. Some of the visualizations, such as treemaps and scatter plots, have proven to be harder to grasp, which can be explained by the fact that these are totally new concepts for the users. However, after studying the visualizations for some while all of the users have become very positive towards displaying the information in this way; they can easily see the benefits with visualizing the information.

The treemap and the line chart are the most explored visualizations for real-time in this thesis work; they have proven to be very good at displaying information in real-time and we do not believe that we could have chosen any other display medium which could have done this in a better way. To choose *display mediums* which can serve its purpose is very important when designing dashboards (Few, 2006). The scatter plot is very useful at displaying relations between different values (Few, 2007), something we believe the user group has great use of. However, the choice of using a scatter plot to display data over/under a benchmark in our concept might not have been the best. The users had problems with understanding this visualization, something we think can be explained with that they are used to comparing data to benchmarks and indices using a line chart. We are also somewhat skeptical at letting the bar chart in our concept be displayed in real-time; they often display summations and averages, which means the values will not fluctuate very heavily. Bar charts could consequently be updated on demand or with different time intervals instead of burdening the software with having them updated in real-time.

When designing for real-time data it is a risk that the interface becomes very intrusive as well as when designing for large amounts of data the interface can become much cluttered. To avoid this, we believe it is very important to try and *tone down the*

interface and keep it as simple as possible, with an uncomplicated layout and by using subtle colors, borders and controls. Few (2006) mean that simplicity often is the way to go when designing dashboards; overuse of colors and abundant decoration should be avoided. We have tried to tone down the interface in our concept by using for example grey borders, white and grey background colors etc. We do however believe that this could be done to a further extent if we have had the time to concentrate more on the graphic design of our concept. The colors are for example something we would like to make some changes to, to make the treemaps fit better with the rest of the visualizations.

The *time controls* are a part of the concept that we have looked at to a great extent, since we believe this function to be one of the most useful and important ones. There are six different ways to go back and forth in time. When we performed the user test some of the users were only interested in having the ability to go back and forth in time by manipulating the graph, with for example drag-and-drop. It is important to provide the user with this easy accessible way to go back and forth in time. But since this feature is most suitable for going backward or forward a short distance in time we also believe it is important to provide other possibilities for doing this. We believe this is something the users too would have noticed if they would have been able to go further back in time than a couple of hours, which was the case in our demonstrator. The users however all agreed that this was a very useful function, especially in combination with the duplicate function, which enabled them in a simple way to correlate different time periods of the data with each other.

When we studied the primary user group, consisting of advisory and discretionary asset managers and stock brokers, we noticed that they worked at a high pace; the work situation was very hectic. They did not have time to perform any elaborate *settings* and chose not to use the features in their current software, which were too time-consuming. To perform analysis and use different functions therefore need to be very easy, simple and possible to be performed quickly. However, at the same time as the users need to have fast access to analysis and functions and it should be easy and simple to use them, they also need to be able to analyze and look at the data in several different ways to be able to see patterns and trends in the data. It is a hard balancing act when the users want the software to be as easy, quick, and with as few settings as possible, but still want to be able to make elaborate analysis, which demand that the user spend some time on letting the system know what kind of analysis is to be performed. By keeping the dialogue boxes and settings in the software as simple as possible, something Cooper & Reimannn (2003) also believe is very important, we however hope that we have been able to both supply the users with the functions they need as well as with effectiveness and ease of use. The final evaluation also proved that the users had no problems with using the functions; they thought it was easy to use and understand the functions as well as they believed them to be less time-consuming than the functions in their current software.

A demand that was expressed from all the users and which they believed to be very important is that all data and information needs to be supplied in *real-time*. This is something we also think is of great importance; however, we believe that the term real-time can be discussed. Real-time data for this domain means as often as it is technically possible, down to under milliseconds. However, when it comes to a real-time visual analytics tool we do not believe it is important to supply real-time data in

this meaning, but rather in the meaning of continuously updated data to supply the data's development over time. All analysis performed in a real-time visual analytics tool is performed on static data, since the data becomes static as soon as it enters into the software, before the users have time to analyze it.

When it comes to *movement and interaction* our work has showed that the movement will not be so visible and that the interaction with real-time data therefore does not differ from the interaction with static data. Though this is something we believe to be very specific for this domain, since they look at no less than a time period of 30 minutes. Real-time data for other domains may very well move in a very fast speed and if this is the case we still have the belief that the interaction techniques used for static data may not be so suitable for real-time data since it will be hard to keep up with the data and be able to perceive what is happening if the data moves in a very fast speed. Here we believe it becomes even more important to supply functions for going back in time and displaying the data again and also to pause the data or slow it down to be able to perceive what is happening. Some visualizations, we believe scatter plots to be one of these, might also be less suitable for fast streaming of real-time data than others, something which needs to be further studied.

Being able to *distinguish between data* displayed in real-time and historical data becomes very important when the movement of the real-time data is very moderate. It then becomes hard to see which visualization is in real-time and which is displaying historical data. Since all visualizations are to display real-time data in our concept we did however not explore this aspect greatly. The most important information, i.e. the real-time updates in this case, should also be highlighted and immediately stand out (Few, 2006). In the treemaps we have tried to do this by displaying frames around the values that are updated. The users had no problems with understanding this during the tests and we believe this was a good way to display real-time updates in the treemap.

The users had some requests regarding displaying additional information to the information displayed in our concept. They wanted to display somewhat more detailed and specific information, for example the order book for each stock. This was information which we had in our prototype at an early stage of the development, but had discarded during our work since we believe it to be unnecessary to display information of not that high importance, which the users also already have access to in their information system. This is since we see the real-time visual analytics tool as a complement to their current information system, not as a replacement for these systems. If our concept was to replace their current system it would need to contain a whole lot of functions and information besides the information we have chosen to display and the functions we have explored. We therefore believe that the users input on displaying more information has to do with that they are used to having access to this information in their current system, rather than that they would actually have any use of displaying this information in our system.

8.3 Design recommendations

In this section, a number of design recommendations for visualizing large amounts of real-time data based on the result and discussion will be presented. These design recommendations are divided into more *general recommendations concerning*

designing for real-time data, as well as *design recommendations specific for the domain and user group* that we focused on in this thesis work.

8.3.1 General design recommendations for visualizing real-time data

1. *Provide a context for the real-time data with historic data.*

The real-time data is of no use in a visual analytics tool if you do not have anything to compare it with, i.e. historical data.

2. *Distinguish the most important information, i.e. the real-time updates, from the rest of the data.*

Highlight the important information, i.e. the values that are updated so the user is aware of what is happening in the system.

3. *Facilitate for analysis of data at different points in time.*

To be able to find correlations in the data you must be able to display data at different points in time and for different time periods. It is important to be able to look back in time as well as change the time period for the data displayed.

4. *Separate historical data from real-time data clearly.*

This becomes even more important if it is not easy to see the difference immediately, if the movement of the real-time data is very small.

5. *Facilitate for the user to sift through the large amounts of real-time information.*

Since large amounts of data are handled it is important to be able to distinguish the most important information from the rest, which is why for example search, alerts, and minimum/maximum functions become of uttermost importance.

6. *Apply the same interaction techniques for real-time data as for static data, before inventing new interaction techniques.*

The interaction techniques used for static data should be tested also for real-time data, since they may very well be suitable also for real-time data if the speed of the real-time data is not that fast.

7. *Strive for simplicity in the interface; use subtle colors and a straightforward layout.*

Interfaces handling large amounts of real-time data can very easily become cluttered. To try to avoid this, the interface should be kept simple and toned down.

8. *Display data that is to be compared on the same page.*

Data that is not to be compared can be displayed on separate pages, but display data that is to be compared on the same page to facilitate comparison and analysis of the data.

8.3.2 Design recommendations specific for the domain and user group

1. *Supply both overview and detailed information.*

The users need to see different levels of detail, ranging from overview of the market and the portfolios they manage, down to a specific portfolio or stock.

2. *Facilitate for different kinds of analysis of the data to find patterns and trends in the data.*

Analysis of different kinds must be possible, such as trend analysis using different technical indicators.

3. *Avoid time-consuming actions and settings by providing easily accessible features and ways to analyze the data.*

The users do not have time to go through several different dialogues and perform a lot of settings to be able to carry out an analysis of the data. All features should be easy to access and visible to the user to make it easy and quick to perform different actions.

4. *Facilitate for the user to discern extremes in the data.*

The users look very much at extremes, the lowest or highest values. Therefore it should be easy to spot the extreme values in the data.

5. *Provide easy means to compare the data to different benchmarks and indices.*

Comparing the data to a benchmark or an index is a common way to see how well a specific stock or portfolio is performing.

6. *Relate the incoming real-time data with other displayed information.*

By showing relations between the data in a simple way, by for example displaying incoming news on the visualization, the data becomes a lot easier to understand, analyze, and find correlations in.

7. *Choose display mediums appropriate for displaying the real-time data for this domain.*

Some display mediums are more suitable for showing overview while others are better at showing detailed information. The treemap is for example suitable for displaying large amounts of hierarchical data, while line charts are more suitable for displaying development over time for a stock price.

- 8. The system should provide the users with continuously updated real-time information at all times.*

The users are very dependent on receiving the information in real-time, since the smallest delay can make them lose large amounts of capital. They assert that all data must be supplied in real-time. Even when the user is looking at historical data, there should be a mean for the user to see real-time data simultaneously.

- 9. Display many different kinds of information, such as news, stock prices, indices, financial ratios, and reports.*

All of this information is needed for the users to be able to make accurate analysis of the data and to make correct decisions. The users shall also have access to all of these kinds of information in the same software.

- 10. Have access to historical data up to 10 years back in time.*

Since the users often search for patterns and trends in the data, they need to have access to historical data a long period back in time.

8.4 Future Work

If this work was to be developed further the most natural way to proceed would be to develop a new interactive prototype of the concept we have developed to be able to test it thoroughly. Another interesting approach would be to explore the second potential user group, the analysts, to see if they have the same needs and requirements as the first user group. Since there also have been suspicions that there are differences in the professional roles and the work situation on markets in foreign countries compared to the Swedish financial market it would be interesting to explore what parts of our concept that can be applicable to financial markets abroad.

It could also be interesting to look at other domains that are dependent on large amounts of real-time data, for example systems for controlling network activity, or security systems, and investigate if there are any differences in visualizing and interacting with real-time data for different domains. To explore other display mediums for real-time data than the ones we have concentrated on in this thesis is also a possible future work. It is also necessary to look more at how real-time data can be distinguished from static data.

9 Conclusions

In this section we present our final conclusions about visualization and interaction with real-time data for the stock market, and answer the research questions for this thesis.

In this thesis work we have explored the possibilities for visualizing and interacting with large amounts of real-time data for the stock market with the aid of a visual analytics tool. To be able to answer the primary research question of this thesis, we will first answer the two additional research questions.

Who are the potential users of a real-time visual analytics tool within the stock market?

The potential users within the stock market for a real-time visual analytics tool primarily consist of advisory and discretionary stock brokers and asset managers. It became clear in the interviews conducted in the early stages of the thesis work that both of these professional groups need to perform analysis of the large amounts of real-time data which they handle daily to be able to make decisions on how to invest. A potential secondary user group for real-time visual analytics tools has also been identified; the analysts, who perform and supply the primary user group with more thorough analysis of the real-time data.

What are the similarities and differences between visualizing and interacting with static versus real-time data?

Visualizing and interacting with real-time data is in many ways similar to visualization and interaction with static data. However, when it comes to visualizing real-time data it becomes very significant to provide a context for the real-time data with historic data, since you have no use of the real-time data if you have nothing to compare it with. When the prototypes were developed, it also became obvious that it was very important to visually separate historic data from real-time data in a clear and efficient way, so the users can differentiate between the different kinds of data. Distinguishing the values that are updated from the rest of the values in the visualization is vital. When you visualize large amounts of real-time data the interface risks to become cluttered. It is therefore important to tone down the graphic design for an interface handling real-time data.

When user tests were performed on the developed prototypes it was proven that the same interaction techniques can be used for interaction with real-time data as for static data. This is due to the fact that the movement in the plots is not that significant in real-time, which was discovered early on in the development of the prototypes. Since the primary user group does not look at data containing a shorter time span than 30 minutes, the movement of the data in the visualizations will not be so significant. It however remains unclear if the interaction techniques used for static data also will fit real-time data for other domains than the one we have studied.

CONCLUSIONS

After answering the additional research questions, we are now able to answer the primary research question of this thesis.

How can TIBCO Spotfire, a visual analytics tool designed for static data, be modified to provide real-time data for the stock market?

Research conducted during the thesis work show that a visual analytics tool providing real-time data for the stock market needs to contain and display very large amounts of data and therefore the design should be more similar to a dashboard than that of a visual analytics tool designed for static data.

The visual analytics tool providing real-time data for the stock market needs to supply the users with both overview and detailed information; to be able to see different levels of detail in the information is vital. It therefore becomes important to provide a display medium which can supply an overview of the large amounts of hierarchical data on the stock market. When prototyping, we discovered that a display medium that is well suited for this is the treemap. It is also vital to display more detailed information, such as the data's development over time to make it easy for the users to find patterns and trends in the data. Line charts have proven to be very suitable for displaying more detailed stock market data, such as development of stock prices and indices over time.

It is essential that the data is provided in real-time and that the real-time visual analytics tool facilitate for the user to distinguish extreme values in the large amounts of data. In addition to this, it should be possible to perform analysis on the data quickly and easily. Functions such as alerts, going back and forth in time, as well as searching and finding information in the visualizations become very important in order to make analysis more efficient.

References

- Ahlberg, C., Williamson, C., & Shneiderman, B. (1992). Dynamic Queries for Information Exploration: An Implementation and Evaluation. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 619 - 626.
- Andersson, L. (2006). *Värdepapper – En genomgång av kapitalmarknaden och skattereglerna*. Sjunde upplagan. Uddevalla: Björn Lundén Information.
- Andersson, L. & Svensson, U. (2001). *Aktieaffärer – praktisk handbok om hur du framgångsrikt handlar med aktier*. Första upplagan. Uddevalla: Björn Lundén Information AB.
- Bederson, B., Shneiderman, B., & Wattenberg, M. (2002). Ordered and Quantum Treemaps: Making Effective Use of 2D Space to Display Hierarchies. In *ACM Transactions on Graphics (TOG)*, Vol. 21, No. 4, pp. 833-854. New York: ACM Press.
- Bell, J. (2000). *Introduktion till Forskningsmetodik*. Tredje upplagan. Lund: Studentlitteratur.
- Björk, S. (2000). *Flip Zooming: The Development of an Information Visualization Technique*. Diss. Göteborgs University: Department of Informatics.
- Card, S.K., Mackinlay, J.D., & Shneiderman, B. (1999). *Readings in Information Visualization: Using Vision to Think*. San Francisco: Morgan Kaufmann Publishers.
- Cooper, A., & Reimann, R. (2003). *About Face 2.0. The Essentials of Interaction Design*. Indianapolis: Wiley Publishing.
- Dumas, J.S., & Redish, J.C. (1999). *A Practical Guide to Usability testing*. Exeter: Intellect.
- Few, S. (2007). Visualizing Change - An Innovation in Time-Series Analysis. *Visual Business Intelligence Newsletter*, September.
- Few, S. (2006). *Information Dashboard Design: The Effective Visual Communication of Data*. Sebastopol: O'Reilly & Associates.
- Frankfort-Nachmias, C. & Nachmias, D. (1996). *Research Methods in the Social Science*. 5th edition. London: Hodder Education.
- Gulliksen, J., & Göransson, B. (2002). *Användarcentrerad systemdesign*. Lund: Studentlitteratur.
- Hansson, S. (2005). *Aktier, Optioner, Obligationer: En Introduktion*. Tionde upplagan. Lund: Studentlitteratur.

- Jones, J.C. (1992). *Design methods*. Second edition. New York: John Wiley & Sons.
- Karlsson, M. A. (2007). *Kompendium och Metodappendix*. Course Material: Interaction Design, Analysmetoder 5p, Spring 2007, IT University, Göteborg.
- Löwgren, J., & Stolterman, E. (2004a). *Thoughtful Interaction Design. A Design Perspective on Information Technology*. Cambridge: MIT Press.
- Löwgren, J., & Stolterman, E. (2004b). *Design av informationsteknik: Materialet utan egenskaper*. Andra upplagan. Lund: Studentlitteratur.
- Mackay, W.E., & Fayard, A-L. (1997). HCI, Natural Science and Design: A Framework for Triangulation Across Disciplines. In *Proceedings of ACM Designing Interactive Systems '97 (DIS '97)*, pp. 223-234. New York: ACM Press.
- Merino, C.S., Sips, M., Keim, D.A., Panse, C., & Spence, R. (2006). Task-at-hand interface for change detection in stock market data. In *Proceedings of the working conference on Advanced visual interfaces*, pp. 420-427. New York: ACM Press.
- Nesbitt, K. V., & Barrass, S. (2004). Finding Trading Patterns in Stock Market Data. In *IEEE Computer Graphics and Applications*, Vol. 24, No. 5, pp. 45-55. New York: IEEE Press.
- Panopticon. (2007). *Information Visualization Technology: Conceptual White Paper*. Version 2.1.4.
- Preece, J., Rogers, Y., & Sharp, H. (2007). *Interaction Design: Beyond human-Computer Interaction*. 2nd edition. New York: John Wiley & Sons, Inc.
- Shneiderman, B., & Plaisant, C. (2005). *Designing the User Interface: strategies for effective human-computer interaction*. 4th edition. Reading: Addison-Wesley.
- Silverman, D. (2005). *Doing Qualitative Research: A Practical Handbook*. 2nd edition. London: Sage Publications.
- Simunic, K. (2003). Visualization of Stock Market Charts. In *The 11th International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision'2003*.
- Spence, R. (2007). *Information Visualization: Design for Interaction*. 2nd Edition. Harlow: Pearson Education Limited.
- Sveriges Riksbank (2007). *Den svenska finansmarknaden 2007*.
- Swedbank Markets, FöreningsSparbankens Investmentbank. (2003). *Aktieboken – Spara i aktier. Lättare än du tror*. Katrineholm: Allde & Skytt AB.
- TIBCO Software Inc. (2008). *TIBCO Spotfire 2.1 – User's Manual*.
- Trost, J. (1997). *Kvalitativa intervjuer*. Andra upplagan. Lund: Studentlitteratur.

Tufte, E.R. (2006). *Beautiful Evidence*. Connecticut: Graphics Press LLC.

Tufte, E.R. (1983). *The Visual Display of Quantitative Information*. Cambridge University Press.

Wallgren, A., Wallgren, B., Persson, R., Jorner, U., & Haaland, J-A. (1996). *Graphing Statistics & Data: Creating Better Charts*. Thousand Oaks: Sage Publications.

Ware. (2004). *Information Visualization Perception for Design*. 2nd edition. San Francisco: Morgan Kaufmann Publishers.

Internet references

[1] Nationalencyklopedin. Available at http://www.ne.se.proxy.lib.chalmers.se/jsp/search/article.jsp?i_art_id=221147&i_sect_id=22114706&i_word=&i_history=9 [2008-03-12]

[2] Aktiespararna. Rollerna på finansmarknaden. Available at <http://www.aktiespararna.se/lar-dig-mer/Fordjupningar/Finansyrken/Lar-dig-rollerna-pa-finansmarknaden/> [2008-03-12]

[3] Google Finance displaying the Coca Cola Company. Available at <http://finance.google.com/finance?q=NYSE%3AKO> [2008-03-12]

[4] Reuters First Brochure. Available at http://about.reuters.com/productinfo/reutersfirst/material/Reuters_First_brochure.pdf [2008-03-12]

[5] Reuters First Manual. Available at http://about.reuters.com/productinfo/reutersfirst/material/RF_training_manual.doc [2008-03-12]

[6] SIX Trader Manual. Available at http://www.six.se/upload/pdf-filer/sixtrader_manual.pdf [2008-03-12]

[7] Yahoo Finance. <http://finance.yahoo.com/> [2007-10-05]

[8] Demo of Panopticon's treemap. Available at: <http://217.13.226.112/demo/trademonitoring/> [2008-03-12]

[9] Demo of Hive Group's treemap. Available at: <http://www.hivegroup.com/solutions/demos/stocks.html> [2008-03-12]

[10] Demo of FusionGadgets dashboard. Available at: <http://www.fusioncharts.com/gadgets/Demos/SparkDashboard/Index.html> [2008-03-13]

[11] Nielsen, J. Ten Usability Heuristics. Available at: http://www.useit.com/papers/heuristic/heuristic_list.html [2008-02-22]

Appendix 1. Interaction techniques in TIBCO Spotfire

Action	Description
Mouseover item	Highlight item.
Click on item	Mark clicked item and unmarks all other items.
Ctrl + click on item	Toggle mark for clicked item. Leave other items as is.
Drag Rectangle	Mark items within rectangle and unmark all other items. When you start to drag, a transparent rectangle is shown indicating the area to be marked.
Ctrl + Drag Rectangle	Add items within the rectangle that are not already marked to marked items, and remove items that are already marked.
Shift + Ctrl + Drag Rectangle	Remove all items from marked items that are not within the rectangle. This allows you to Mark rows that satisfy multiple criteria in a line chart or parallel coordinate plot. For example, you may mark lines that are low in the beginning by dragging a rectangle around them and then Shift+Ctrl+drag a rectangle around high lines at the end to restrict marked rows to only those rows that satisfy both criteria.
Shift + click	Useful in table visualizations, where you mark all items between the previously marked item and the clicked item. All other items are unmarked.
Ctrl + Shift + click	Useful in table visualizations, where you mark all items between the clicked item and the previously marked item. Other items are left as is.

Up/Down/Left/Right Arrow key	Mark the first item above/below/left/right of the previously marked item and unmark all other items.
Shift +Up/Down/Left/Right Arrow key	Extend the set of marked items up/down/left/right, unless the next item is already marked. If the item is already marked, unmark the previously marked item.
CTRL + Up/Down/Left/Right	Moves highlight up/down/left/right.
Space	Toggle mark for highlighted item. Unmark other items.
CTRL + Space	Toggle mark for highlighted item. Keep other items as is.
Ctrl + Shift + Space	Mark the range of items between the highlighted item and the previously marked item.

Appendix 2. Interview guide

About Respondent	
1.	What is your occupation?
2.	How long have you had this occupation?
3.	What assignments does your work consist of? Is your work related to the stock market?
4.	Do you have any knowledge of the stock market through your work?
	<i>For a stock broker:</i>
5.	<i>At which stock exchange(s) do you trade? (NYSE/London/Tokyo etc?)</i>
6.	<i>Do you trade with stocks of a certain kind? (Industry sector/high risk/low risk etc?)</i>
7.	<i>How often do you perform stock trade?</i>

Software	
8.	Which tools/software do you use in your work?
a.	➤ How many software do you use? What different purposes do they have?
b.	➤ Are they linked together?
c.	➤ Within your company, who use software for stock trade? (occupation)
d.	➤ Which system connects your company to the stock exchange? (Infolect) Mandatory? What information is given?

Trade	
9.	<i>Describe a typical stock trade scenario.</i>
a.	➤ Who are your customers? (persons/companies?)
b.	➤ How many customers do you have? (a fixed number or does it vary?)
c.	➤ How do you communicate with your customers? (Phone/email/fax/software/in person?)

Real-time data	
10.	What data do you need to have in real-time in your work?
a.	➤ <i>Does it correspond to the data presented in stock tables? If not, which data is added/removed?</i>
b.	➤ What do you gain by having this data in real-time?
c.	➤ How can you tell that the data is real-time? What distinguishes it from other kinds of data?
11.	Is some real-time data more/less important in your work?
12.	When looking at real-time data, what time range is of interest? (days/a day/hours/minutes)
a.	➤ When are the different time ranges interesting?

Data/Information	
13.	What other kinds of data/information do you handle in your work, besides real-time? Static/dynamic data?
a.	➤ Which data/information is the most important?
b.	➤ For what purposes are the different kinds of data used?
14.	Are the different kinds of data used in combination with each other or separately? When? Why?
15.	How often is the data updated at present?
16.	How is the data updated? (on demand/automatically) How can you see if the data is updated automatically or need to be done manually?
17.	How often does the data need to be updated?

Visualization	
18.	<i>How frequently do you monitor rise and fall in stock prices?</i>
19.	<i>Approximately how many stocks do you monitor at the same time?</i>
a.	In what way do you monitor your data/information? (Graphs/Tables/News ticker?) Is it different depending on if the data is static/dynamic/real-time?
b.	➤ Why do you monitor the data/information in this way? On the same/different screen?
c.	➤ Can you think of any advantages/disadvantages with the current presentation?
20.	Do you have any automated system for stock trade? (Automatically perform trade at set values? Who sets these values? Based on what?) How do you monitor the automated system?
21.	Can you set your software to alert you when stock prices reach or go below a certain value?
a.	➤ How does the system alert you?
b.	➤ What is your opinion of the alert system?
c.	➤ Is this a commonly used feature?
	If no:
d.	➤ Could this be a useful feature?

Problems/possibilities	
22.	Do you experience any problems with the software you use?
23.	Can you think of any features which do not exist in the current software, but that you would like?

Appendix 3. Persona

This persona describes a typical user within our primary user group as well as his work situation. Our fictional user is called Niklas M. Bergmark.

Niklas is 34 years old and works as an advisory asset manager in Göteborg, at one of the largest banks in Sweden. This work includes making assessments of customer's needs and appropriate risk level. Based on this assessment Niklas analyze different investment possibilities and advice his customers on what to invest in. The customers take Niklas' recommendations under advisement and make a decision on how to invest. Niklas then executes this investment.

He has worked as an asset manager for seven years and prior to this he went to the School of Business, Economics and Law at Göteborg University and studied national economy for four years. Before beginning his studies, Niklas did his military service and also backpacked around the world. Today Niklas lives in a house in Örgryte with his wife since one year back and his six months old son. Niklas work long hours, at least 10 hours per day, sometimes more. He is dedicated to his work and could not imagine working with something else. Niklas likes sports and whenever he has some spare time he likes to meet his friends and watch a soccer game or work out.

Niklas work as an asset manager is diverse, since it contains several different kinds of tasks every day. He has about 40 different customers, which he has contact with and speaks with regularly. Niklas always needs to be available to his customers and thereby spends a lot of his time talking on the phone. When he is at work he never puts the phone down and when he is outside work he never turns his mobile phone off. A lot of the customer contact Niklas also performs via e-mail. The work as an asset manager involves a rapid work pace and a work environment that is hectic and noisy. Niklas likes that the asset managers at his company work as a team; they cooperate and share information with each other daily.

In his work Niklas handles and monitor huge amounts of real-time data and information on a daily basis. His workplace is dominated by three computer screens providing Niklas with constantly updated information about the stock market. Niklas look primarily at news, stock price, key figures and indices. He frequently uses several different kinds of software to gather and process all of this data. Internet is also a primary information source for Niklas.

He has a talent for finding and seeing patterns in the vast amount of information he handles and often concentrates on looking at extremes. To be analytic and be good at spotting trends is a necessity when working as an asset manager, since they need to be able to make quick decisions.

Appendix 4. Requirement specification

Data/information which the users need to have access to and monitor in real-time
<ul style="list-style-type: none"> • News headlines and full articles and news
<ul style="list-style-type: none"> • Stock prices and closures
<ul style="list-style-type: none"> • Financial ratios
<ul style="list-style-type: none"> • Indices
<ul style="list-style-type: none"> • Risk parameters
<ul style="list-style-type: none"> • Currency exchange fluctuation
<ul style="list-style-type: none"> • Stock volumes sold/bought (show supply/demand)
<ul style="list-style-type: none"> • Economical statistics
<ul style="list-style-type: none"> • Extremes (best/worst/highest/lowest)
<ul style="list-style-type: none"> • Commodity prices
<ul style="list-style-type: none"> • Interests
<ul style="list-style-type: none"> • GDP's
Data/information over time
<ul style="list-style-type: none"> • Find historical correlations between companies and sector/industry
<ul style="list-style-type: none"> • Find patterns in stock prices historically
<ul style="list-style-type: none"> • Look at stock prices several years back in time
<ul style="list-style-type: none"> • See indications of a new price trend
<ul style="list-style-type: none"> • Use different technical indicators to analyze the data over time, such as: <ul style="list-style-type: none"> • Relative Strength Index • Moving Average Convergence Divergence <ul style="list-style-type: none"> • Bollinger Bands
Overview of portfolio data
<ul style="list-style-type: none"> • Have an overview of the stocks in the portfolio/in the market
<ul style="list-style-type: none"> • Monitor the stocks in one portfolio, what sector they belong to etc.
<ul style="list-style-type: none"> • See an overview of how much is invested in each sector
<ul style="list-style-type: none"> • See current portfolio and how the portfolio has changed over time
<ul style="list-style-type: none"> • See overview of what kind of security a portfolio consists of
<ul style="list-style-type: none"> • To see aggregated data (i.e. summative data of stocks in a certain sector)
<ul style="list-style-type: none"> • To find abnormalities in the data through overview
Details of the data in the portfolio
<ul style="list-style-type: none"> • Customize and analyze the yield in a portfolio
<ul style="list-style-type: none"> • Make hypothesis tests
<ul style="list-style-type: none"> • Control risk level in portfolios
<ul style="list-style-type: none"> • See best stock performance in portfolio
<ul style="list-style-type: none"> • See worst stock performance in portfolio

Searches in data
<ul style="list-style-type: none"> • Search databases to extract information about portfolios
<ul style="list-style-type: none"> • Find all clients that has invested in a certain company
<ul style="list-style-type: none"> • Search in text for specific keys
Comparisons of data
<ul style="list-style-type: none"> • Compare stock prices with indices
<ul style="list-style-type: none"> • Compare information about stocks with peer group and indices
<ul style="list-style-type: none"> • See correlations between companies and industry
<ul style="list-style-type: none"> • Supervise traders, be able to compare them, their performance
<ul style="list-style-type: none"> • Compare portfolio performance with benchmarks
<ul style="list-style-type: none"> • Compare different analysis with each other (risk vs. fundamental)
Work situation
<ul style="list-style-type: none"> • Stressful and active environment, nothing should be time-consuming
<ul style="list-style-type: none"> • Information should be easy to share among co-workers
<ul style="list-style-type: none"> • Must be able to monitor many different kinds of data in one view
<ul style="list-style-type: none"> • Making customization settings manually should be kept to a minimum
General
<ul style="list-style-type: none"> • Information must be delivered in real-time (updates every millisecond)
<ul style="list-style-type: none"> • Need to be easy to use, intuitive
<ul style="list-style-type: none"> • Quality of data + real-time updates is essential
<ul style="list-style-type: none"> • Be able to make duplicates in time for comparisons
<ul style="list-style-type: none"> • Make presentations and reports for clients
<ul style="list-style-type: none"> • Drastic changes in position and sorting of objects should be avoided
<ul style="list-style-type: none"> • Must be able to see both real-time data and historical data

Appendix 5. Scenarios

Scenario 1

Have an overview of portfolio performances: An asset manager may want to identify all the portfolios that are affected by the profit warning and get an overview of them in order to monitor and have control over the portfolios' performances compared to the market and segments of the market. Potential questions he/she would want to analyze and answer: To what extent is each portfolio affected by events related to the specific stock? For example, how many percent of the portfolio is invested in Sony Ericsson? One way to visualize such an overview could be to compare the overall performance for each of the portfolios with the market and sector performance at present.

Scenario 2

Investigate different information sources: The asset manager needs to gather information from many different sources to be able to form an own opinion of the situation. For example, news and reports about Sony Ericsson's profit warning and statements from other investors, analysts, and asset managers become important. In general, browsing the news feed is one of the most important activities for asset managers. The asset manager keeps track of the news, statements, reports etc. as they come in and follows up on the news that are relevant at the moment. In a situation where a certain event is of special interest the asset manager would probably want to pay special attention to what is said as well as what has been said in the past about similar events about the company.

To facilitate this activity it could be useful if news etc. concerning Sony Ericsson and the profit warning could be extracted from all the incoming real-time data to get the most relevant information displayed separate from the rest of the news. This could for example be done in a graph where incoming information are marked on the stock price curve or in some other way, making it possible to find correlations between incoming information and the fluctuations in stock price. It can also be of interest to see correlations in the past.

Scenario 3

Correlate stock price and index: The asset manager may need to monitor and analyze how the stock's price and related indices are affected by the profit warning, and to see how the market reacts as a whole to the announcement. An asset manager may also want to look at extreme values for the specific stock. For example, look back in time to see the greatest changes in stock price in one single day. How does the events surrounding that particular day correlate with the events of today? Has something like this happened earlier, if so in what context? To make these kinds of comparisons it could be useful to make duplicates in time displaying stock price and index in several different graphs, a combination of graphs showing static and real-time data. For instance, display one duplicate of when Sony Ericsson so far had the greatest fall in one single day, one duplicate of the development for the Sony Ericsson stock for a similar profit warning made in the past as well as a graph of the present day with real-time data.

It may also be relevant to make hypothesis tests to see how potential fluctuation patterns can affect the portfolio as a whole if the stock is not sold. This can help the asset manager to decide when to sell the stock. The result of such a test could be combined with setting an alert which would then tell the asset manager if the stock price reaches a critical level.

Scenario 4

Compare volumes: The asset manager may want to monitor and analyze the volumes sold and bought. Who is selling? Who is buying? How much are they selling or buying? It is for instance interesting to look at how insiders have reacted on the news. How are they trading? For this activity it could be useful to display real-time and static data in the same graph. Displaying current buy and sell volumes compared to buy and sell volumes before the profit warning. Displaying how insiders are buying and selling the Sony Ericsson stock at the moment compared to right before the profit warning.

Scenario 5

Have an overview of a specific portfolio: The asset manager may want to take a closer look at and analyze a specific portfolio and to get an overview of the portfolio. Which types of securities does the portfolio consist of? What sectors do the stocks belong to? To get such an overview the portfolio composition could be displayed hierarchically. Another possible activity is to test how the portfolio would react according to different hypothesis to be able to decide when and if it is time to sell the stock. An alert system could then be set to warn the asset manager if the value is reaching a certain limit.

Appendix 6. Prioritizing list

1. Rewinding the line graph
2. Which way is preferred when rewinding (grabbing the miniature, grabbing the line in the chart, using the slider or using the buttons)
3. Is the miniature line graph useful (rewinding to a peak, i.e. to an event rather than a specified point in time)
4. Duplicate (and manipulate) a line chart
5. Is movement in the plots noticed
6. Can the users tell if the plots show real-time or static data
7. Drill down in a visualization
8. Interaction in the plot (mark a line, make drag rectangle, ctrl-click)
9. Jump between views (right-click/function key)
10. Set an alert
11. Are activated alerts noticed
12. Set min/max symbols
13. Is it useful to add news/events to the line graph
14. Connection between events and plots (which stock does the event relate to)
15. Add symbols to the line chart
16. Search and find information
17. Filter/mark views separately vs. all
18. Understand the difference between filtering and marking
19. Understand what happens when a “reversed” slider is used
20. Which way is preferred to zoom
21. Zooming in on an area in a graph (with the zoom tool, drag rectangle)
22. Find and go to a specific stock/portfolio
23. Change time range for the graph
24. Is there a need to see history in all plots (for example in treemaps/bar charts which are not so good at showing history)
25. How is the scatter plot with over/under benchmark understood (does it provide more information than the treemap)
26. Which portfolios contain a certain stock
27. Set tags/comments at different points in time in the plot
28. How do they prefer to find information (rewinding a graph, scrolling in a list etc.)
29. Minimize windows
30. What do the different plots show
31. Is the right information displayed
32. Is the information they want to see displayed together
33. Add/remove miniatures
34. Navigate by using the miniatures
35. Comparing two stock views/portfolio views
36. Add an indicator to the line graph

Appendix 7. Test scenario

In the market view

- Set an alert to notify you when the stock HM B's one day change goes below 0.50%Mark, in the visualization showing portfolios, the stock with the largest increase measured in percent the last month as well as the portfolio with the largest decrease measured in percent the last month
- In the treemap showing portfolios, hide the marking showing that Jesper's portfolio has performed worst the last month
- Open the portfolio with the largest possession of the Hennes & Mauritz stock in a new view, where we only see that particular portfolio.

In the portfolio view

- In the treemap, find the sector containing three stocks and open it
- Return to the level where the entire portfolio is shown in the treemap
- In the treemap, hide the events that are news
- Go to the view that shows the stock that the alert concerns

In the stock view

- Show history for the stocks Hennes & Mauritz and Industrivärden in the visualization showing the stocks in the portfolio relative to a benchmark
- Duplicate the line graph
- Add the stock Gunnebo to the upper line chart

The line graph

- Rewind to 11.10 in the line chart by using the buttons
- Go back to real-time