ZEUS – Zoomable Explorative User Interface for Searching and Object Presentation

Fredrik Gundelsweiler, Thomas Memmel, and Harald Reiterer

Human-Computer Interaction Lab
University of Konstanz, Universitätsstraße 10,
Box D 73, D- 78457 Konstanz, Germany
{gundelsw,memmel,reiterer}@uni-konstanz.de

Abstract. In this paper we describe a first version of ZEUS, a web application that combines browsing, searching and object presentation. With the zooming and panning based navigation concept of ZEUS and a hierarchical organization of the information space we try to solve the problems of information overload. It has to be evaluated if categorization, zooming and a full text search can minimize that the user gets lost in hyperspace. The concept of ZEUS is based on some thesis about human cognition, navigation and exploration which we hope to prove with evaluation and user testing of our application in the future.

Keywords: Zoomable User Interface, Interactive System, Complex Information Space, Usability, Navigation, Search, Browse.

1 Introduction

In this paper we describe a first version of ZEUS, a software application based on [18] that combines browsing, searching and object presentation which are usually developed separately in software systems. Some systems realize a good search, a nice object presentation or a well designed browsing but most fail to combine them in context of users' tasks. Literature research brought us to the conclusion that users are swapping between search and exploration mode while performing their tasks in information spaces [2], [20], [21]. This change of mode can occur at any time and is hard to predict because user tasks may change while detecting new information.

Software applications, especially for the web, are getting more and more complex with the progression of technological innovations, broadband internet and application programming interfaces like Adobe Flash. Not only commercial internet sites have to provide usable, efficient and effective search, browse and presentation mechanisms. Presenting a product on the web means to give the user an interactive experience including all product data and multi-media support (e.g. audio, video). It would be very nice for users to explore data in task context and several levels of detail. The buzzword "Web 2.0" represents not only principles like user contributed value and co-creation; it also animates to think over the application world. One possible future scenario is to have the web browser as main application handling all different user tasks from the search for information to the creation of an office document.

M.J. Smith, G. Salvendy (Eds.): Human Interface, Part I, HCII 2007, LNCS 4557, pp. 288–297, 2007.

2 Related Work

Operating systems like Windows Vista, Mac OSX Leopard or Sun's Looking Glass project [22] bring some innovation by introducing new search techniques like Spotlight, virtual desktop environments to swap different desktop views (OSX Leopard - Spaces) or windows in a 3d view that can be turned aside to save place on the desktop. Related to our work there are search applications like grokker [11], Liveplasma [12], DateLens [3] and DesignClicks [8] using interesting zoom-, browse and search-techniques in combination. Other systems like MuSE [9], the Attribute Explorer [25] or applications created with Piccolo [4] or Pad++ [5] realize filter concepts with immediate user feedback following or expanding the dynamic query approach [1]. Our ambition is to design a user interface (UI) which picks up good ideas and presents multiple ways a user system view could look like. We think that conventional WIMP (windows, icons, menus, pointers) systems won't solve the problems of usability and user experience we are faced with today. That's why we propose zoomable user interfaces (ZUIs) which can be used to take advantage of the natural way users navigate and orientate in space supporting their physical abilities and habits.

Raskin describes the zooming interface paradigm in [19]. He considers it as superior in comparison with conventional paradigms. "*The zooming interface paradigm can replace the browser, the desktop metaphor, and the traditional operating system.*" [19]. We took this paradigm as initial point for the development of our application. His argument is the better match between zooming navigation concepts and human cognition. Colin Ware supports this theory in [26] where he stated that humans move their bodies and thus their view mostly forward and backward but seldom sideward. Therefore a simple UI has to support just three degrees of freedom. Two degrees of freedom are needed for rotation (direction and pitch) which matches the movement of the head and one degree is for moving forward or backward. Transferred to the design of an UI this argument could be interpreted to realize a panning and zooming navigation. Panning the virtual information space complies moving the head and thus changing the visible area or aspect of interest. The zoom in and out is like moving forward and backward in space controlling the degree of interest of an object.

3 Technical Constraints

There are a lot of technical constraints when developing web applications. The main criterion is that the application has to run in a web browser. Especially when visualizing large datasets the bandwidth and data transfer rates have to be kept in mind. The application should make use of intelligent preloading of data which will probably be explored next by the user. Dependant on the complexity of the visual representation and number of data items the performance maybe an issue because the browser or used browser plug-ins do not support hardware rendering. Another issue is the diversity of browser implementations (Mozilla Firefox, Internet Explorer or Safari may interpret e.g. a java script in different ways). We thought about four approaches to web application development that could meet our requirements of rapid feedback, interactivity and data transfer.

One possible development approach is using DHTML in combination with AJAX [10]. This is working fine for most web browsers and no plug-in is needed. The disadvantage is that there are a lot of AJAX frameworks like Macao [13] or Mochi Kit [15] but no standard development environment which makes coding more comfortable. Additionally the UI and its interactive behaviour are not as easily implemented as with Adobe Flash/Flex 2 or Java. In contrary to the AJAX thin client solution, browser plug-ins can be used. Sun's java applets can be developed fast but are in need of the java plug-in and have an initial loading time on start-up. Adobe Flash/Flex 2 need the installation of the Flashplayer browser plug-in but have the advantage of easy-to-use development environments and standard support for vector graphics. Another interesting possibility comes from Microsoft with the Windows Presentation Foundation (WPF) [14]. Development of web applications is possible with the .net 3 environment and works with the Internet Explorer 7. WPF works with vector graphics and even supports 2d and 3d hardware rendering.

For development of the ZEUS prototype we have chosen Adobe Flash, the server side script language PHP as middleware and a free SQL database. With these components a comfortable development and prototyping of web applications is possible.

4 ZEUS - Theoretical Background

As we have seen a lot of technical constraints exist. Other problems appear when developing a concept for such a web application. One big problem is that the target user group is enormous and different users prefer different kinds of navigation, browsing, searching and object representation. One could reason that we need a system that supports all different navigation, interaction and visualization techniques. The user should be able to configure his preferred view. Following this approach arises two more problems. It is hard to decide which parts of the UI should be configurable. The second issue is the quantity of configurable options. Users are easily overextended if too many options are available. Apart from that it is essential for all options to be easy usable and accessible.

On the first glimpse the “Map View“-visualization of the grokker search engine [11] looks like our iNShOP application (see figures 1 and 2). That's why we want to point out the differences between both applications. Grokker builds up a hierarchy of the search results automatically (it can be searched in Yahoo, Wikipedia and Amazon Books). In addition to the outline view (an expandable list tree view) the search results can be explored in a hierarchical, zoomable “Map View”. With a slider the user is able to narrow search results to the newest documents by date. In fact there are several important differences between the applications. A first technical and maybe negligible point is that grokker is written in the java programming language and therefore needs the java web plug-in to be installed while ZEUS is realized with Adobe Flash and needs the Flashplayer browser plug-in which is widespread among internet users. ZEUS doesn't create its categories automatically. They have to be part of the database connected to ZEUS. With the web 2.0 one could think about to involve user in the categorization process by enabling them to create their own categories and change the assigned categories of data items. This approach generates a more comprehensible categorization than a computer system could calculate automatically. Apart from that the main interaction concept of grokker is like the one of ZEUS. A problem with grokker could be the depth of the hierarchy. While ZEUS makes the depth visible to the user by the category combo boxes for each level, the grokker-hierarchy is not visible to the user. ZEUS is more flexible with the categorization because users can change and select a proper category for each level.

Another difference is the main concept. Grokker is a web search engine that tries to get away from the conventional list presentation of search results. In contrast the focus of ZEUS lies on browsing the information space and on the presentation of simple and complex multimedia data which has to be supported by a full text search. A stringent semantic zoom concept realized in ZEUS is the visualization of search results in different detail depths. Grokker doesn't visualize the details of a result in the "Map View" but in a detail area on the right side of the application. In ZEUS the user can zoom in to an item to view all of the available information directly in the visualization. If the item consists of a lot of information a further zoom in on the item is possible to explore all of its information areas.

A feature of ZEUS that grokker does not have is the change of visualizations for the category tiles. The user can chose between round or square category representation using grokker but no other kind of visualization like a scatterplot or a treemap. A change of the category tiles enables the user to use different visualizations for different data item collections. One concept of Zeus that is not yet implemented is the collection of important data items. The user then can create own category tiles to store individual collections of relevant information.

4.1 Information and Navigation Overload

Information Overload or *Information Pollution* is the term expressing the flood of information rolling over today's internet users. The term *Information Overload* was coined 1970 by Alvin Toffler in [23]. Individuals that are confronted with too much information accomplish their tasks less efficient. The same problem may occur if users can't understand the organization of a presented information space in a search application. This can be the outcome of a bad information organization and presentation or *Information Overload* as described in [24]. If the UI presents too much information at once, users aren't able to process it correctly. They overlook information, categorize relevant information as irrelevant or even feel interfered. The same problem can occur with an "overloaded" navigation. In this case the user is confronted with too many navigation options. Difficulties with the logical organization of the information space have direct effect on the orientation and decision-making ability of the users in the digital world. Too many links and possible ways alienate users so that they can't decide how to reach their user goals. Our thesis is that we can solve these problems by a hierarchical information structure and zooming/panning enriched navigation supported by a full text search.

4.2 Navigation for Searching and Browsing

With ZEUS we develop a ZUI that supports searching and browsing in large information spaces with regard to user tasks, goals and activities. Therefore our system combines search and filter techniques with zooming interaction techniques to narrow search results. The wide range of internet application users with varying computer and domain experience makes it necessary to develop adaptable navigation and visualization concepts. Novices and experts shall be able to solve their tasks by using different strategies.

Users are mixing two basic approaches while trying to find the needed information. These two main strategies are on the one hand the directed search for known items and on the other hand the exploration (e.g. by browsing) of different items of interest. The used terms for searching and browsing are slightly different in literature. In [21] the term "teleporting" is used when a user directly jumps to the needed information. The browsing strategy is described by the term "orienteering". The user narrows the information space by "*a series of steps (e.g., selecting links) based on prior and contextual information to hone in on the target*" [21]. Our opinion is that we can support both strategies best with a UI that includes an always visible full text search and a flexible hierarchical visualization of the information space. In this way the user can change his strategy from searching (or teleporting) to browsing (or orienteering) or the other way round at any time. Concerning the browsing strategy we think that navigation by animated panning and zooming supports the collection of information and the orienteering tasks of users in the best way. It is easier for users to move visually through an information space to explore its contents than to navigate through a hyperlinked collection of sites which organization is invisible to the user. This animated exploration mode has an additional advantage regarding the search function. Is the search result (e.g. triggered by a full text search) one item or a collection of items, the user view can be focused on the relevant item or item group by automatically zooming and panning to its position.

5 ZEUS – User Interface Design and Concept

In the following we present the idea of ZEUS in form of a sample application called iNShOP that is connected to a music database. The application consists of a main area visualizing the objects and categories, and a filter area for search and choosing categories (see figure 1). Filter and category operations are triggered by selecting category attributes in the combo boxes. Selecting "Music type" as first category level organizes the results like in figure 1 with the categories "Drum'n'Base", "Classic", "Pop", "Dance/Electro" and the other eight. Our thesis is that this mix of hierarchical and flat data representation enables a quick switch between search and exploration mode and therefore a more efficient and effective work.

Tiles are the main visual components used to organize the information space and visualize the data items. There are two different types of tiles. Category tiles organize

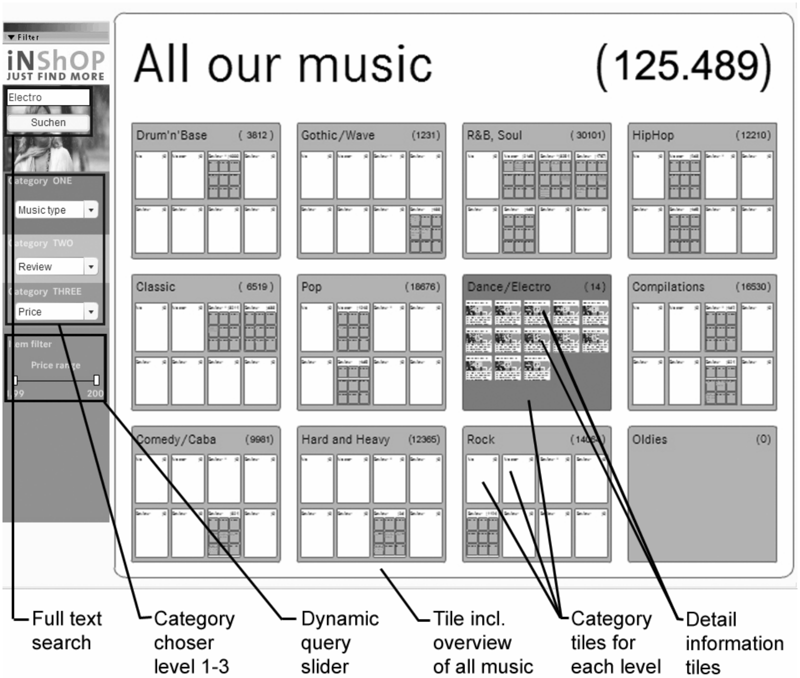

Fig. 1. ZEUS main overview by example of a virtual music store we called iNShOP

the information space in groups on different hierarchy levels. They can include further category tiles or information tiles to visualize the data items in that level. An information tile visualizes one item and can include text, images and multimedia objects like video and sound. Selecting a category in a combo box initiates a recalculation of the tile organization with a redraw of tiles.

For the system decision weather to further categorize the information space or visualize the included information tiles, we can set an item threshold. That means if e.g. less than fifty items (threshold=50) fall in the category "electro", they are shown in the "electro" tile and the information hierarchy is not expanded. If more than fifty items fall in the "electro" category the visual hierarchy is expanded and further category tiles (e.g. price or review points) are created within the "electro" tile. Additionally we included a price slider to narrow the price range of visualized items.

5.1 Main Navigation

The main navigation of ZEUS concerns all visible areas except the information tiles. Semantic and geometric zooming is used to show relevant information at different aspects and/or degrees of interest. Figure 2 shows a short scenario how a search for an artist called "Moby" could look like.

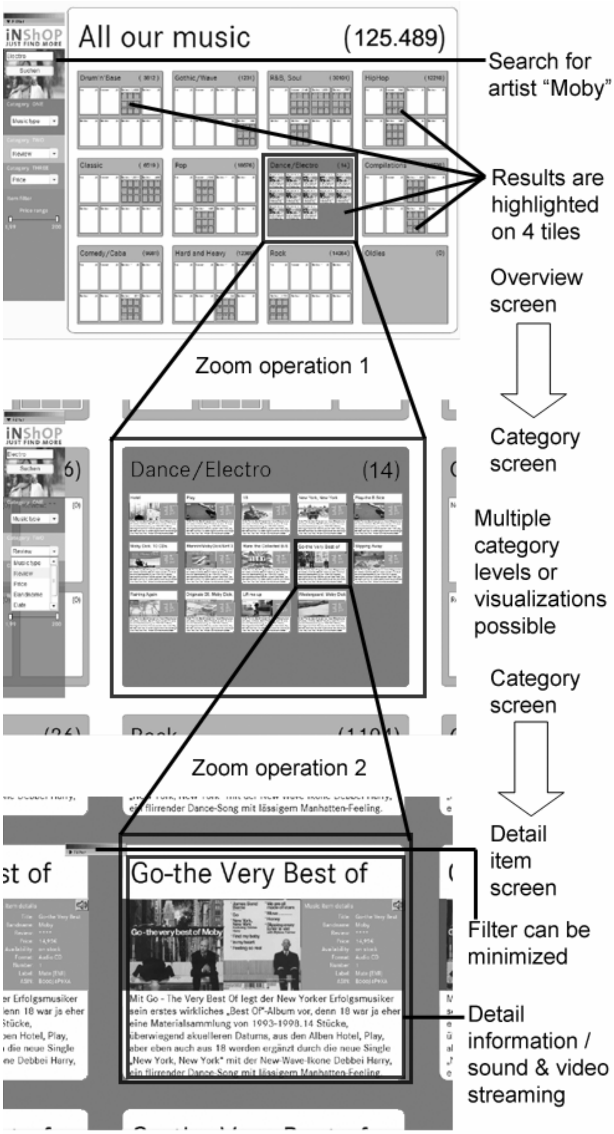

Fig. 2. ZEUS from overview to detail view by zoom in

First the user enters "Moby" as search query into the search field located in the movable and resizable filter area (best visible figure 1 left). The system processes the query and highlights the results. In this case there are results for "Moby" on four tiles ("Drum'n'Bass", "HipHop", "Compilations" and "Dance / Electro"). Now the user is able to change the category combo boxes to let the system build up a hierarchy matching his interests. Maybe on the first level he could be interested in the review rating, so he would chose "review" for the first level combo box. For simplicity we assume he is interested in the "Music type" categorization on the first level because he wants to see which albums are in the category "Dance/Electro". On the first glimpse the user sees that fourteen albums of the artist fall into this category. With a click on the "Dance/Electro" tile the user triggers zoom operation one and the tile enlarges to screen fitting size. Zoom and pan actions are clarified to the user by an animation [6]. Now that the category tile is larger, the included albums are better readable. In future there should be an area where the user can select different visualizations for the category tile view. Let's assume that the user is interested in the album with the title "GO – the very best of". A click on that tile triggers a further zoom into the detail information tile (figure 2 bottom and figure 3). Now the user can see all album details. The way back to the category overview is triggered by clicking on the category tile area visible between the detail information tiles. So it is possible to navigate back to the main overview at any time.

5.2 Navigation and Visualization Within Information and Category Tiles

Navigation within an information tile should follow some straightforward rules so that the main navigation concept still remains consistent.

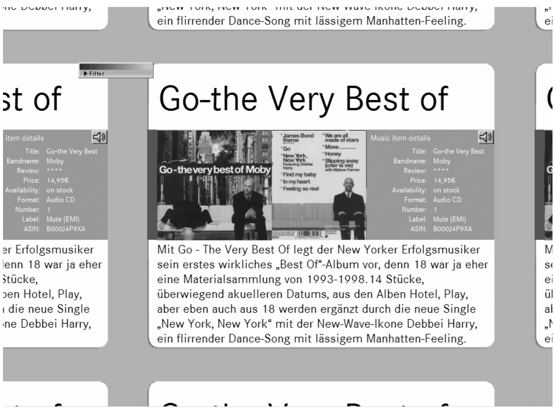

Fig. 3. ZEUS detail item view showing all data of an item e.g. an audio cd

In a detail view title, artist, price, format and so on are shown. Additionally there media streaming can be included. The category tiles shall be able to change their look and feel. In contrary to the systems in the related work section different visualizations can be integrated into our "meta"-system. It will be possible to integrate various visualizations like a table, scatterplot, treemap or others. Each category tile will be able to present its contents independently by other visualizations. Additionally different databases or XML sources can be connected to ZEUS. For each data type dependent on the kind of data other basic templates are available to visualize the data objects. Let's imagine we have music data objects like in iNShOP. The basic template can show title, label, band, other detail information, album image and play an example sound stream. An easy way to add video support would be to adapt a basic template. A play button can trigger the video stream to start and has to be included in our adapted template. Additionally the visual overall representation can be different from the basic template. We propose to use templates for each kind of data domain (e.g. sound compact discs, books, cars or computers).

6 Discussion of Results and Future Work

In summary, we have developed a first version of a nested ZUI that combines search and browse techniques to support users in their search modes. New in ZEUS are the combination of searching and browsing in one UI, the idea of switching the visualization for item groups and the connection of data sources whose results are visualized by different adaptable templates. Our assumption is that users are generally not very familiar with zoom navigation but it may be more effective and efficient than conventional navigation, especially coupled with a full text search and a hierarchical information organization. The main thesis that has to be verified is that human cognition is better supported by zooming and panning than by any other navigation style when exploring an information space visually. A technical issue is runtime rendering without hardware support. Therefore zooming operations with more than approximately 200 items (depending on hardware and complexity of items) are too slow and the system gets stuck. Maybe a change of the development environment to Adobe Flex or Microsoft WPF can avoid this. Future work will be the implementation of visualizations contained in the category tiles and further improvement of the UI. Additional search and exploration techniques like scent trails [16] or navigation by query [7] can be implemented. At the end we would like to test different application versions (which differ in database, search techniques, browsing techniques, with and without zooming and panning) against each other.

References

1. Ahlberg, C., Shneiderman, B.: Visual Information Seeking: Tight Coupling of Dynamic Query Filters With Starfield Displays. In: Ahlberg, C. (ed.) Proceedings of Human Factors in Computing Systems, pp. S313–S317. ACM Press, New York (1994)
2. Bates, M.J.: Toward an Integrated Model of Information Seeking and Searching. The Fourth International Conference on Information Needs, Seeking and Use in Different Contexts, Lisbon, Portugal (September 11-13, 2002)
3. Bederson, B.B., Clamage, A., Czerwinski, M.P., Robertson, G.G.: Datelens: A fisheye calendar interface for pdas. ACM Transactions onComputer-Human Interaction 11(1), 90–119 (2004)
4. Bederson, B., Grosjean, J., Meyer, J.: Toolkit Design for Interactive Structured Graphics IEEE Transactions on Software Engineering (TSE). HCIL-2003-01, CS-TR-4432, UMIACS-TR-2003-03 (January 2003)
5. Bederson, B.B., Hollan, J.D.: PAD++: A zooming graphical user interface for exploring alternate interface physics. In: UIST 94: 7th ACM Symposium on User Interface Software and Technology, pp. 17–27. ACM press, New York (1994)
6. Bederson, B.B., Meyer, J., Good, L.: Jazz: An extensible zoomable user interface graphics toolkit in Java. UIST'00, ACMSymposium on User Interface Software and Technology, CHI Lett. 2(2), 171–180 (2000)

7. Cunliffe, D., Taylor, C., Tudhope, D.: Query-based Navigation in Semantically Indexed Hypermedia. In: Proceedings of the eighth ACM conference on Hypertext, Southampton, United Kingdom, pp. 87–95. ACM Press, New York (1997)
8. Designklicks visual presentation of art 2007, URL: (last visited: 09.02.2007) http://designklicks.spiegel.de/
9. Furnas, G.W., Zhang. X.: MuSE: A Multiscale Editor. In: Proceedings of ACM UIST'98, pp. 107–116 (1998)
10. Garrett, J.J.: Ajax: A New Approach to Web Applications. Adaptive Path LLC (February 18, 2005), Url: (last visited: 09.02.2007) http://www.adaptivepath.com/publications/essays/archives/000385.php
11. Grokker seeking system. URL: (last visited: 09.02.2007) http://www.grokker.com/
12. Liveplasma seeking system. URL: (last visited: 09.02.2007) http://www.liveplasma.com/
13. Macao Ajax Framework. Url: (last visited: 09.02.2007) http://macao.sourceforge.net/
14. Microsoft Windows Presentation Foundation (WPF) at msdn. Url: (last visited: 09.02.2007) http://msdn2.microsoft.com/en-us/netframework/aa663326.aspx
15. Mochi Kit Ajax Framework. Url: (last visited: 09.02.2007) http://mochikit.com/
16. Olston, C., Chi, E.H.: ScentTrails: Integrating Browsing and Searching on the Web. ACM Transactions on Computer Human Interaction 10(3), 177–197 (2003)
17. Perlin, K., Fox, D.: Pad: an alternative approach to the computer interface. In: proceedings of Computer graphics and interactive techniques (1993)
18. Ken, P., Jon, M.: Nested user interface components. In: Proceedings of the 12th annual ACM symposium on User interface software and technology, Asheville, North Carolina, United States, November 07-10, 1999, pp. 11–18. (1999)
19. Raskin, J. (ed.): The Humane Interface. New Directions for Designing Interactive Systems. Addison-Wesley Verlag, Reading, MA (2000)
20. Rose, D.E.: Reconciling information-seeking behavior with search user interfaces for the Web. JASIST 57(6), 797–799 (2006)
21. Schaffer, E., Straub, K.: The answer you're searching for is...browse. In: UI Design Update Newsletter (January 2005), Url: (last visited: 12.02.2007) http://www.humanfactors.com/downloads/jan052.htm
22. Sun Microsystem: Project Looking Glass. Url: (last visited: 12.02.2007) http://www.sun.com/software/looking_glass/
23. Toffler, A.: Future Shock. Random House Inc., Reissue Edition (September 1984)
24. Turetken, O., Sharda, R.: Development of a fisheye-based information search processing aid (FISPA) for managing information overload in the web environment (ISSN: 0167-9236). In: Decision Support Systems, June 2004, vol. 37(3), pp. 415–434. Elsevier Science Publishers B.V, Amsterdam, Netherlands (2004)
25. Tweedie, L., Spence, B., Williams, D., Bhogal, R.: The Attribute Explorer. In: CHI'94, Boston, Massachusetts USA, April 24-28, 1994, ACM, New York (1994)
26. Ware, C.: Information Visualization: Perception for Design. Morgan Kaufmann, San Fransisco, Kalifornien (2004)
27. Allison, W., James, L., Michael, S.: Goal-directed zoom. In: CHI 98 conference summary on Human factors in computing systems, Los Angeles, California, United States, April 18-23, 1998, pp. 305–306 (1998)