ZuiScat - Querying and Visualizing Information Spaces on Personal Digital Assistants

Thorsten Büring
University of Konstanz
Universitätsstrasse 10
78467 Konstanz, Germany
buering@inf.uni-konstanz.de

Harald Reiterer
University of Konstanz
Universitätsstrasse 10
78467 Konstanz, Germany
reiterer@inf.uni-konstanz.de

Abstract

ZuiScat is a visualization concept for querying large information spaces on Personal Digital Assistants (PDA). Retrieval results are presented in a dynamic scatterplot, which is enhanced by geometric and semantic zoom techniques to provide smooth transitions from abstract visual encodings to data content. The same visualization is also used to manage bookmarks and to serve as a powerful query history tool. User feedback suggests that ZuiScat provides intuitive and efficient data access but still needs further improvement in terms of zoom usability and visual mapping.

Categories and Subject Descriptors

H.5.2 [**Information Interfaces and Presentation**]: User Interfaces - Graphical User Interfaces

General Terms

Human Factors, Design

Keywords

ZuiScat, PDA, handheld, small screen, ZUI, zoom, scatterplot, scatter plot, overview, focus

1. INTRODUCTION

Due to improved performance and connectivity features, Personal Digital Assistants are increasingly used to search large information spaces such as a customer database or the World Wide Web. Unfortunately, the majority of interfaces on today's handheld software present information in a textual format, making use of lists and tables, which usually display fewer than 20 entries on a single PDA screen. Having retrieved a large number of search results, users will thus spend a significant amount of time scrolling or flipping pages in order to identify those records they are actually interested in. In other words, they are forced to apply a tedious, slow and error-prone manual search.

MobileHCI'05, September 19–22, 2005, Salzburg, Austria.

The problem of how to present information more effectively on computer screens has been the subject of extensive research in the field of information visualization. Based on this research, a wide variety of novel graphical interfaces were implemented to lower the cognitive demands required when dealing with high information loads. Most of these tools target large screens and high-end computers, however. PDAs, in contrast, currently feature a 240x320 or 480x640 pixel display, 32-128 MB of RAM and a 203-624 MHz CPU. Moreover, neither a fully featured keyboard nor a mouse is available. Most of the device interaction is done using a stylus (pen) and a set of 4 to 5 hardware navigation buttons. With respect to these constraints, we developed ZuiScat - a concept for a zoomable scatterplot interface to facilitate information retrieval and data presentation on PDAs. The concept is currently implemented as a prototype application which allows users to search an online movie DVD collection. The main ideas behind ZuiScat and the results of an initial user test are described in the following sections.

2. VISUAL INTERFACES

Visual interfaces, in contrast to textual interfaces, use compact abstract representations of data rather than space-consuming words to present information. As a result, the data-pixel ratio is increased, which means that more information can be displayed per screen, while the cognitive overhead caused by scrolling or paging is reduced [15]. The process of mapping abstract data to visual forms is complex. Mackinlay refers to it as defining a sentence in a visual language that expresses relations in a data table [14]. The designer's task is now to find a balance between compressed data representations and the mental effort it will take the user to learn the language. A key solution for achieving this goal lies in the innate human perceptual abilities.

In perceptual psychology, certain encoding mechanisms are known to be preattentive, ie. the user is able to process a given visual information faster than 10 msec per item. Unfortunately, the set of preattentive encoding features suitable for information visualization turns out to be rather limited. The designer is left with the options of spatial position, color, shape, orientation, surface texture, motion and blink coding. Moreover, each of these categories is limited in how many expressions it provides that are easily distinguished. Rapid preattentive processing requires that, for instance, no more than eight colors and less than five orientation steps are used. Furthermore, conjunctions of codes are considered to be non-preattentive, but there are exceptions to this rule:

space or perceived convexity may be combined with color, and motion with target shape [23].

3. SCATTERPLOT

Among the codes presented, spatial position is regarded being the most dominant property [7]. A way to exploit this feature for a given data set is to introduce a scatterplot where each data object is represented by a single symbol which is spatially mapped against an X and Y axis. ZuiScat uses this approach to make extensive search results on PDAs more manageable for the user. Unlike lists or tables, scatterplots can provide an overview of large data sets and clearly reveal clusters, trends, and statistical outliers [21][20]. On the other hand, they are less straightforward to read than tables, but nevertheless most users find them reasonably intuitive [15].

Scatterplots have been used to present abstract data since the early 1800s and are one of the most common forms of data graphics in scientific literature [21]. In information visualization they have also proven successful in supporting a variety of retrieval tasks, e.g. finding movies [2] or houses [24] or aiding pharmaceutical research [1]. A first attempt to implement a scatterplot visualization on a PDA platform was the PalmMovieFinder [8]. The application is based on the ideas of Ahlberg and Shneiderman's FilmFinder [2] and provides access to a collection of movies by spatially mapping them as 8x8 pixel icons to the static axes of year of release (X) and popularity rating (Y). The user queries the database by using direct manipulation filters for movie genre and film classification. Tapping the filter checkboxes results in immediate updating of the scatterplot, showing only those movies (icons) that satisfy the filter requirements. To view a data object's content, the user taps on an icon and the movie details are presented on a new screen.

ZuiScat does not focus on a dynamic query approach, but rather assumes that users have a specific information need that they are able to formulate in terms of keywords. Hence, the user is provided with a simplistic search interface as known from popular web search engines like google or altavista (see Figure 1). To start the retrieval process for the movie collection, the user enters a query and then hits the Go button. As shown in Figure 2, search results are represented by small rectangles, which are positioned according to the scatterplot dimensions. By default, the dimensions are set to the variables of year of release (X) and lending frequency (Y). Each of the axes is mapped by label coloring to a combo box in the top toolbar. The user can change an axis assignment by selecting another variable from the corresponding combo box. The variables available next to the default ones are: popularity rating, section and language. Changing an axis assignment causes the scatterplot visualization to update its labeling and to redistribute items.

3.1 Mapping Nominal Data to Position

In ZuiScat, both quantitative (e.g. rating) and nominal values (e.g. section) are encoded by spatial position. This approach is crucial. While nominal values do not possess a meaningful order, positioning may give the user the false impression that such a relation does exist [7]. However, we did not observe this effect when evaluating a previously implemented retrieval system called VisMeB [12]. Instead, mapping nominal values to position can sometimes reveal important information to the user that would otherwise be hard

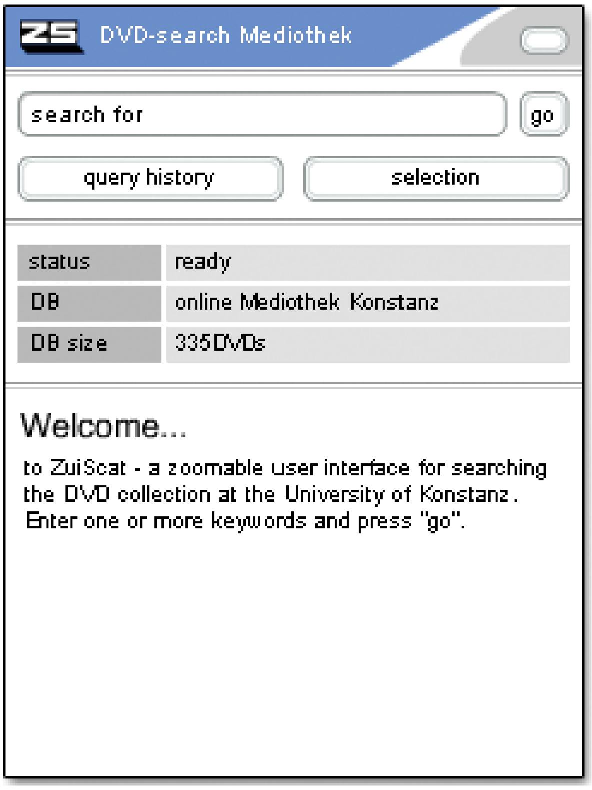

Figure 1: Start screen.

to impart [15]. When targeting PDAs, this approach also provides another benefit: variables that are not spatially mapped would otherwise have to be expressed by introducing additional visual encodings such as shape, or texture of scatterplot symbols, for instance. Even though this technique could increase the data-pixel ratio, it may also make it more difficult for the user to visually scan the scatterplot. Besides that, visual codes are seldom self-explanatory. Further information like a legend would need to be displayed, taking up valuable screen space. Hence, we assume that overall, the advantages of mapping nominal values to position outweigh its potential risks and drawbacks.

3.2 Accumulation of Data

Given a reasonably large information space, there is a high probability that some data objects have similar attribute values and thus also share the same spatial position for the scatterplot visualization (multiple data point), e.g. there are two movies that have a lending frequency of 4 and were released in 1993. Therefore, several DVDs may be represented by a single rectangle. In order to indicate such an accumulation of data, ZuiScat uses a small preattentive set of gray scales (white, light gray, dark gray, black). The darker a rectangle, the more data objects it holds. Although this technique is not precise, it provides a rather intuitive way of improving the overview of item distribution.

3.3 Meta-information

Meta-information about the search, such as query keywords and the number of hits achieved, can be accessed by tapping the arrow button on the top right hand side. A small

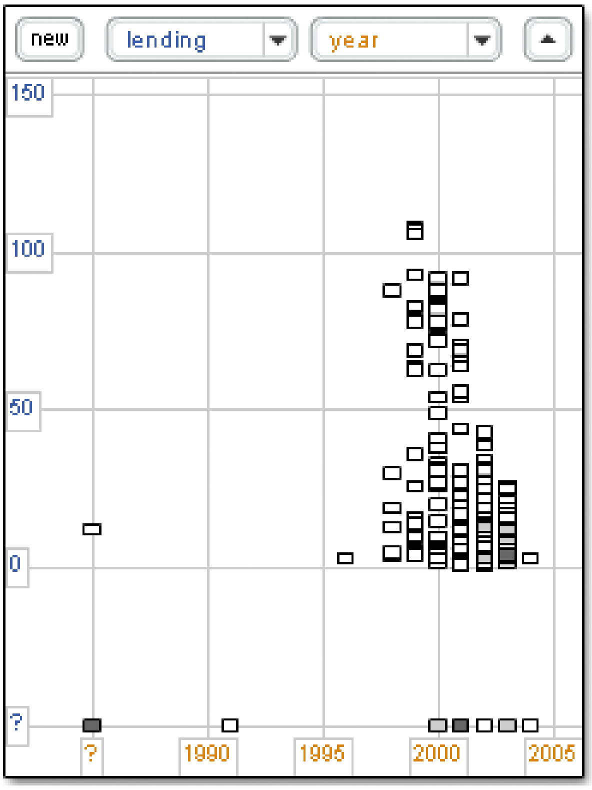

Figure 2: Scatterplot visualization.

window unfolds from the toolbar as shown in Figure 3. Displaying this information on-demand saves screen real estate for the benefit of the scatterplot. Moreover, there is no actual need to display meta-information continuously: most times users will remember the current query executed and a direct, though rough, visual feedback about the amount of retrieval results is already given by the visualization. Tapping anywhere on the screen closes the meta-information window.

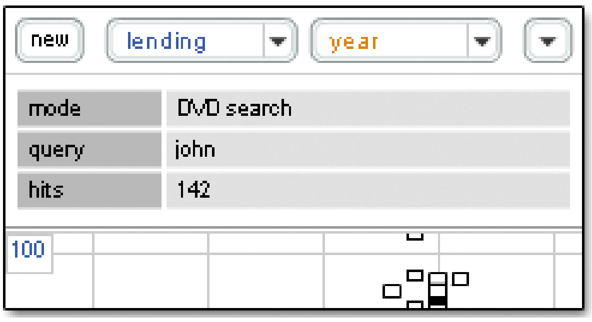

Figure 3: Meta-information about the search.

4. ZOOMABLE USER INTERFACE

As well as being provided with overviews of data, the user must also be able to retrieve the content of specific data objects. ZuiScat features a zoomable user interface (ZUI) to make each item accessible. ZUIs are based on the assumption that navigation in information spaces is best supported by tapping into our natural spatial and geographic ways of thinking [16]. In order to implement this approach on a computer screen, data objects must be organized in space and scale. Users can interact directly with the information space by performing panning and zooming operations. Since a ZUI lets the user view much more information than can normally fit on a single screen [4], it thus provides a valuable solution for enhancing the limited PDA screen real estate.

Different kinds of zoom are to be distinguished. Most common is geometric zoom in which objects are simply magnified. Zooming in, the object's size increases, and vice versa. Semantic zoom, in contrast, allows objects to change their appearance as the amount of real estate available to them changes [16] [9]. Thus, data represented by a point at the default scale may, for instance, grow to a block of text when magnified.

There are only a small number of PDA zoom interfaces that have been implemented as yet. One example is an application called PowerView, which provides structured access to personal information management (PIM) software [6]. Following a flip-zooming approach, the screen is divided into several domain views of information objects (address book, calendar, mailbox and to-do list). When the user selects an object, more screen space is allocated to it so that more content can be displayed. A similar approach was also taken by DateLens, a fisheye calendar for small screens developed at the University of Maryland [5]. Dates are organized in a tabular display where each row represents one week, with seven columns representing the days of the week. Up to 52 rows (representing an entire year) can be displayed on a single screen. The user focuses on a certain date by tapping the corresponding cell. Based on a fisheye algorithm and semantic zoom, the cell expands and reveals more detailed information.

For ZuiScat, we implemented both semantic and geometric zoom techniques to overcome several problems that often apply to scatterplot visualizations. These problems are discussed in the following subsections.

4.1 Reduce Clutter

Selecting scatterplot items of interest by simply tapping them with the stylus is often not feasible. Maximizing the data-pixel ratio for large information spaces may cause data representations to become very small and thus hard to hit accurately. Moreover, many data sets have an uneven distribution, causing the items to overlap and cluster together. In order to reduce the density of clusters, systems like ActiveGraph [15] apply a logarithmic transformation that redistributes data more evenly over the screen. A more manual, PDA-specific, solution is Liquid Browsing [22]. Scatterplot items behave as if floating in oil or liquid and tapping a cluster with the stylus causes a so-called nuzzle effect: representations that originally overlapped are spread out, making it easier to isolate and select single items. Both approaches, however, lead to spatial distortions that may confuse users. Zooming, on the other hand, reduces clutter without changing the spatial relation of the scatterplot. More screen space can be allocated to the focus of interest, while smooth scaling transitions help users to preserve their sense of position and context [19].

In order to perform a zoom operation in ZuiScat, the user first sets a focus point by tapping the screen with the stylus. A blue cross-hair appears at the selected position. By continuously pressing the upper navigation button of the PDA, the scatterplot is smoothly magnified and the area denoted by the focus point moves to the center of the screen. At the now-larger scale, the user is able to accurately tap the item of interest, even if most of the representation is covered by another item (see Figure 4). Zooming out is achieved by pressing the lower navigation button. If not previously stopped, the operation automatically comes to a halt when the visualization is back to the default scale. By dragging the scatterplot surface with the stylus, the user is able to pan the scatterplot without changing the current scale.

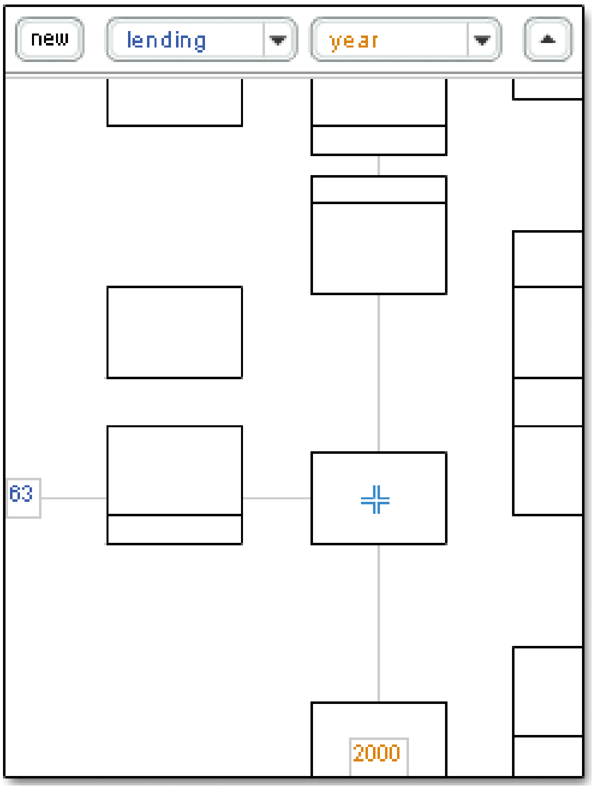

Figure 4: Geometric zoom.

4.2 Transition from Overview to Detail

Having focused on an item of interest, the user wants to switch from abstract representation of data to a content view. To our best knowledge, all scatterplot-based applications implemented so far use either a separate frame, a pop-up window or a new screen to display content. Each of these approaches slows down the search process considerably. Users may not only be forced to perform tedious repetitive tasks (e.g. closing pop-ups) but their attention is also frequently drawn away by other windows. Returning to the scatterplot, users need time to re-orientate. In order to avoid this, at a certain scale ZuiScat switches from geometric to semantic zoom, using the magnified rectangle as a display area for the content it represents. As shown in Figure 5, the rectangle turns into a record card showing a movie title. Further information about the DVD can be accessed by tapping the subcontent units labeled with description, details, poster and trailer. Depending on the current scale, the user may need to perform another zoom-in operation to magnify text or pictures (see Figure 6). Units for which there is no data available are marked gray (disabled).

Overall, the user never actually leaves the scatterplot, which means that no orientation disruptions are caused. Zooming in and out, the user can rapidly gain insight into the data objects of the underlying information space.

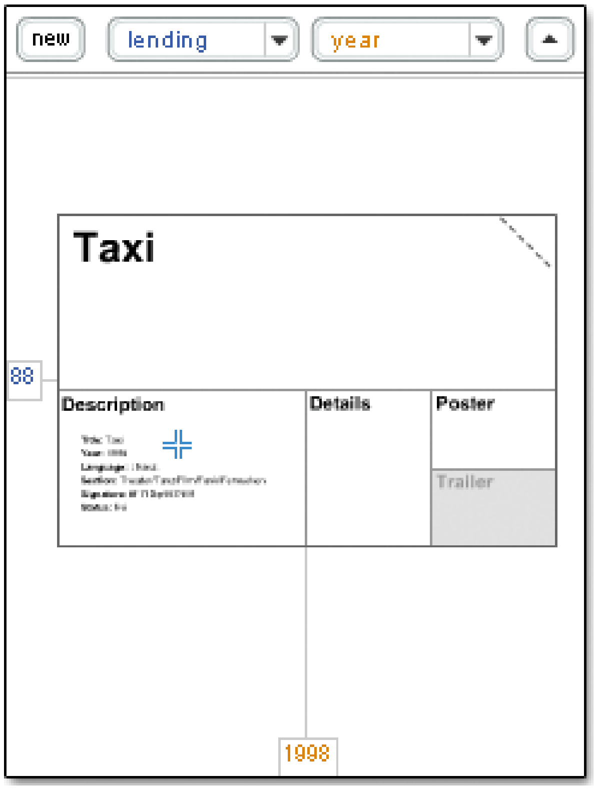

Figure 5: Semantic zoom.

4.3 Multiple Data Points

Another problem that scatterplot visualizations often face is caused by multiple data points [1]. To make each of the items equally accessible, while at the same time preserving the spatial layout consistency, the principle of closure (part of the Gestalt laws [13]) is used as an auxiliary tool to support comprehension. When adding semantic zoom, all items of the multiple data point are displayed as separate record cards but grouped together by the outline of the initial rectangle (see Figure 7). That way, the user intuitively knows that the DVDs presented belong together, i.e. share the same value pair for the given scatterplot dimensions. In order to fit several record cards into a single rectangle, they must be drawn at a smaller scale, but due to the ZUI this aspect is not crucial.

5. USER-GENERATED INFORMATION SPACES

In information retrieval systems, users must be able to execute queries and to bookmark or save items of interest. In doing so, they generate two additional information spaces, namely a query history and a selection. In ZuiScat, both spaces are again visualized in a zoomable scatterplot. Hence,

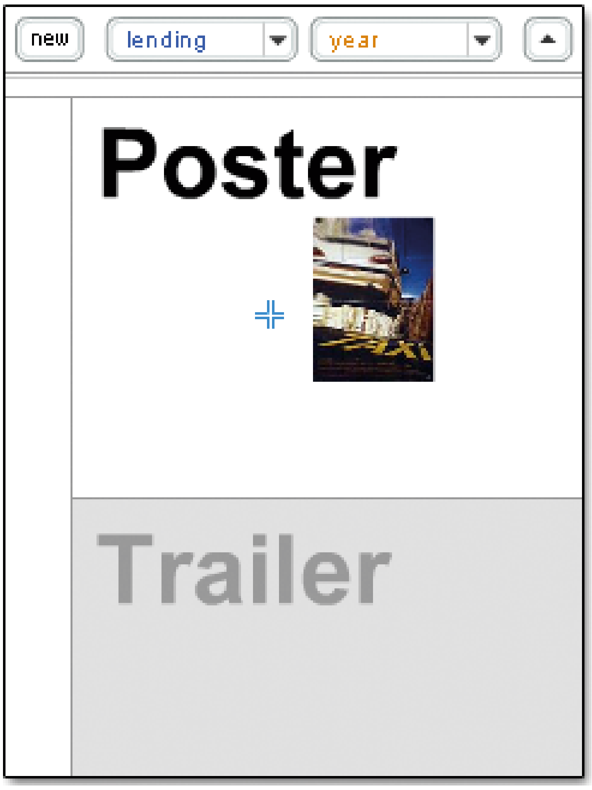

Figure 6: Zooming in on a subcontent unit.

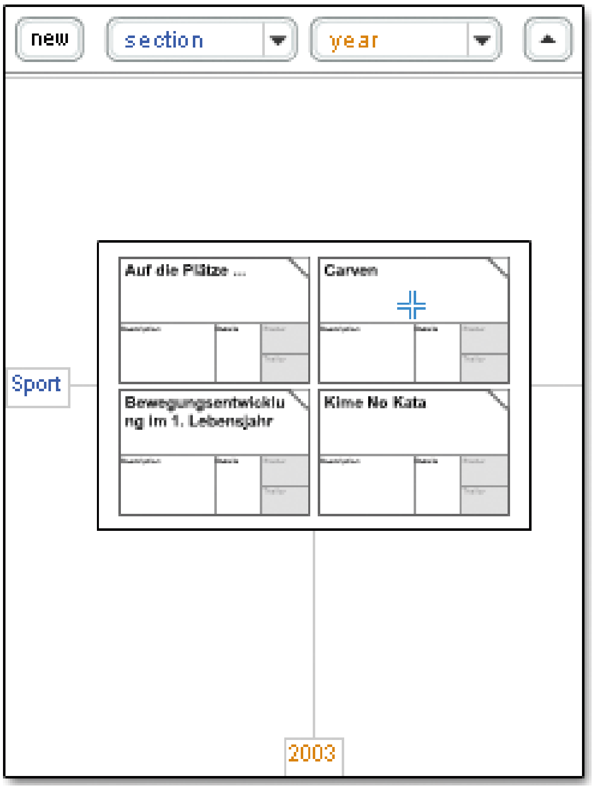

Figure 7: Multiple data point by Gestalt laws.

users only have to learn a single interface for viewing and interacting with different data sets.

5.1 Selection

While exploring the DVD collection, the user wants to bookmark items of interest for further consideration. This is done by folding the edge of the item's record card by tapping it with the stylus (see Figure 8). Tapping the selection button on the start screen (see Figure 1) loads a scatterplot that shows all DVDs previously bookmarked. Since the selection is a subset of the DVD collection information space, the same variables (year, lending frequency, rating, section and language) can be used for spatial positioning. In a future version of ZuiScat, additional information may be included for each item, such as the time of bookmarking and a reference to the query by which the DVD was returned.

5.2 Query History

ZuiScat stores each executed query in a query history and assigns meta-information to it. Tapping the query history button brings up a scatterplot where each item represents a previous search. The user can trace and identify specific queries by adjusting the scatterplot dimensions. Dimension variables available in the query history are: time of query execution, number of keywords, number of hits and number of items bookmarked. Zooming in on an item, the query text appears along with the meta-information stored for that particular search (see Figure 9). The user is also provided with the options of either loading the query keywords into the search field on the start screen or re-executing the query.

6. PROTOTYPE AND USER TEST

A ZuiScat prototype was developed to conduct an initial user test based on observation and interviews. This first test is meant to show us whether users can comprehend the basic visualization and interaction strategies used. For future versions of ZuiScat, however, more complex test settings are needed that will allow us to compare the efficiency of the novel interface with conventional, table-based PDA displays. Also, it might be worthwhile to explore the impact of stationary usage versus mobility.

The prototype was implemented in Macromedia Flash and currently provides access to a movie DVD collection of 335 items. The small size of the information space is owed to the limited performance of Flash on PDAs. The application targets a minimum screen resolution of 240x320 pixels and runs on every platform for which Flash Player 6 (or higher) is available. For the user test, the prototype was installed on a Hewlett-Packard iPAQ hx4700 Pocket PC running Windows Mobile 2003. The device features a 624 MHz processor, 64 MB SDRAM and a 480x640/64K color VGA touchscreen.

6 Users (2 male and 4 female) were asked to perform several retrieval tasks using ZuiScat on the PDA under stationary conditions. The user group consisted of a journalist, a project manager in business development and 4 students of non-IT-related university courses. Age ranged from 21 to 33 years. Only one user had previous experience using a PDA, but all of them had basic computer skills such as using a word processor, browsing and searching the Internet and writing emails.

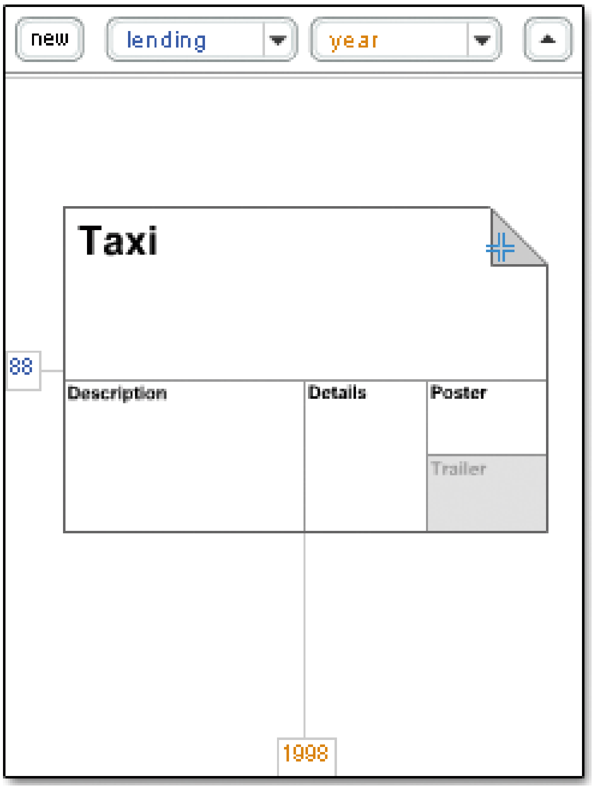

Figure 8: Bookmarking a DVD by folding the edge of the record card.

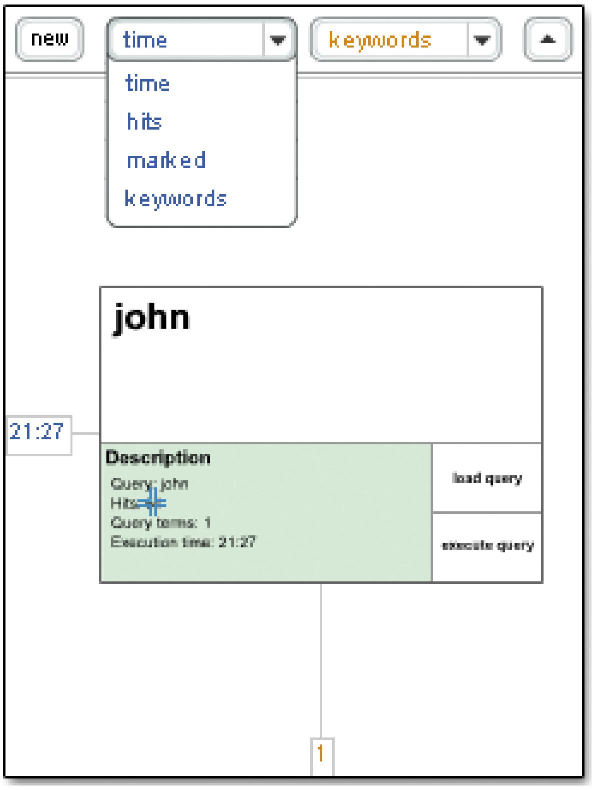

Figure 9: Query history showing a previous search for movies related to the name "john".

After being given a short introduction on how to use a PDA, users were first asked to try out the scatterplot interface on their own while thinking aloud. Support was provided as required. Subsequently, the participants had to solve a set of retrieval tasks that involved an increasing effort in scatterplot adjustment and zoom operations. Finally, they were asked to give their overall impression of the software and to add any additional comments or observations they felt were pertinent to the improvement of ZuiScat.

The user test showed that once participants understood the basic principles of the visualization, they were able to solve all retrieval tasks quickly with only a minimum of support. Moreover, the concept of using semantic zoom to display data content inside the scatterplot proved to be intuitive and easy to grasp. Users appreciated the concise design of the interface and that it was not overloaded with information. User feedback also helped to identify important issues for improvement with regards to zoom interaction and visual encodings.

Zooming. The separation of steering and zoom interaction, which was intended to simplify navigation, proved a failure. Users found the sequential procedure tedious and slow and would clearly prefer a one-step and purely pen-based zoom interaction. Navigation buttons, as suggested by user behavior, could instead be used for view panning. Zooming in on an item, users also experienced difficulties in accurately achieving the right magnification scale for reading text. With regard to this, it may be useful to introduce a more bounded or guided way of zooming. In his book, Raskin for instance suggests using predefined zoom ratios at which zooming slows down or briefly stops, making it easier to get characters to standard sizes [17].

Visualization. While not having a problem with the spatial mapping of both quantitative and nominal data, users got confused with the gray scale encoding for multiple data points. This effect is likely to be related to the size of the scatterplot items. The smaller a panel becomes, the harder it is for the user to distinguish between different shades of the filling color.

User feedback concerning the experimental bookmarking mechanism turned out to be rather ambiguous. Being asked to bookmark a DVD, users did not recognize the edge of the record card as a tappable area but rather kept looking for a check box. Once explained, they found the edge-folding-metaphor appealing and clear. Users also appreciated the concept of a visual query history. Though powerful, this feature is more demanding than a simple back-button approach and thus needs explanation. Considering the screen design, a better visual differentiation between query history and retrieval scatterplot was suggested to avoid confusion of modes. Also, the visual mapping of combo boxes to scatterplot axes needs to be improved.

7. FUTURE WORK

While Macromedia Flash proved to be a good choice for rapid prototyping, the runtime environment lacks performance when it comes to processing large amounts of data on PDAs. Hence, ZuiScat will be re-implemented in either C# or embedded C++ to allow the visualization of medium to large information spaces. A user-centered design approach will help to iteratively improve the application's usability. Current milestones for improvement are as follows:

- As indicated by the user test results, the encoding of spatial accumulation of data and the mapping of combo boxes to scatterplot axes need to be revised to be more intuitive.

- By zooming in on a multiple data point, the user has already reduced the information load to a potentially small number of items. However, even though the visualization of data by the principle of closure (Figure 7) was immediately grasped by all participants, it does not allow for efficient comparison of items. In order to fulfill this requirement, we are planning to provide the user with the additional Hypergrid tool [18, 11, 10]. Hypergrid is another recent project of our HCI working group and was inspired by semantic zoom applications like DateLens. Just as in a conventional table, each row of the Hypergrid holds attribute values for a single item. The user can expand and collapse table cells to access more detailed content while still being provided with an overview of the complete data set. A prototype visualization for PDAs has already been implemented and thus the next step is to achieve a seamless integration of both Hypergrid and zoomable scatterplot interface

- Given the current development state of ZuiScat, the user is only able to encode two variables at a time. Thus, there is no convenient way of identifying, say, all movies in the section "Theater" that were released between 1995 and 2000 and have a popularity rating of at least 5. A visual filter mechanism may prove useful to support such operations.

- Special attention will be given to the pen-based zoom interaction. Different approaches involving zoom ratios and pen gestures must be implemented and evaluated. Pen gestures may not only support zoom operations but could also serve as a valuable alternative to space-consuming visual controls such as buttons. Examples of this approach are marquee menus, as proposed by Baudisch et al [3].

- For accessing highly scattered data, users need to perform several sequential zoom and pan operations. The system could facilitate this task by allowing users to set multiple focus points (anchors). Users would switch between different anchors by executing a single command. All panning and zooming required would then be handled by the system while animated transitions help to avoid orientation disruptions.

8. CONCLUSIONS

We have described a new visualization tool called ZuiScat that enables users to query and visualize information spaces on PDAs. User test results indicate that the combination of scatterplot visualization and a zoomable user interface can provide a powerful solution for overcoming limited screen real estate. Yet, the potential of the idea is not exhausted. Several options have been identified to improve and enhance the current concept, laying the groundwork for future versions of ZuiScat.

9. ACKNOWLEDGMENTS

This work was supported by the DFG Research Training Group GK-1042 "Explorative Analysis and Visualization of large Information Spaces".

10. REFERENCES

[1] C. Ahlberg. Spotfire: an information exploration environment. *SIGMOD Rec.*, 25(4):25–29, 1996.

[2] C. Ahlberg and B. Shneiderman. Visual information seeking: tight coupling of dynamic query filters with starfield displays. In *CHI '94: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 313–317. ACM Press, 1994.

[3] P. Baudisch, X. Xie, C. Wang, and W.-Y. Ma. Collapse-to-zoom: viewing web pages on small screen devices by interactively removing irrelevant content. In *UIST '04: Proceedings of the 17th annual ACM symposium on User interface software and technology*, pages 91–94. ACM Press, 2004.

[4] B. B. Bederson and A. Boltman. Does animation help users build mental maps of spatial information? In *INFOVIS '99: Proceedings of the 1999 IEEE Symposium on Information Visualization*, pages 28–35. IEEE Computer Society, 1999.

[5] B. B. Bederson, A. Clamage, M. P. Czerwinski, and G. G. Robertson. Datelens: A fisheye calendar interface for pdas. *ACM Transactions on Computer-Human Interaction*, 11(1):90–119, 2004.

[6] S. Björk, L. E. Holmquist, P. Ljungstrand, and J. Redström. Powerview: structured access to integrated information on small screens. In *CHI '00: CHI '00 extended abstracts on Human factors in computing systems*, pages 265–266. ACM Press, 2000.

[7] S. Card. Information visualization. *The human-computer interaction handbook: fundamentals, evolving technologies and emerging applications*, pages 544–582, 2003.

[8] M. D. Dunlop and N. Davidson. Visual information seeking on palmtop devices. In *Proceedings of HCI2000, volume 2*, pages 19–20, 2000.

[9] G. W. Furnas and B. B. Bederson. Space-scale diagrams: understanding multiscale interfaces. In *CHI '95: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 234–241. ACM Press/Addison-Wesley Publishing Co., 1995.

[10] C. Grün, J. Gerken, H. C. Jetter, W. König, and H. Reiterer. Mediovis - a user-centred library metadata browser. To be published in Research and Advanced Technology for Digital Libraries 2005. Proceedings of the 9th European Conference, ECDL 2005.

[11] H. C. Jetter, J. Gerken, W. König, C. Grün, and H. Reiterer. Hypergrid - accessing complex information spaces. To be published in People and Computers XIX - The Bigger Picture, Proceedings of HCI 2005.

[12] P. Klein, H. Reiterer, F. Müller, and T. Limbach. Metadata visualisation with vismeb. In *IV '03: Proceedings of the Seventh International Conference on Information Visualization*, page 600. IEEE Computer Society, 2003.

[13] K. Koffka. *Principles of Gestalt Psychology.* Harcourt Brace, New York, 1935.

[14] J. Mackinlay. Automating the design of graphical presentations of relational information. *ACM Trans. Graph.*, 5(2):110–141, 1986.

[15] L. Marks, J. A. Hussell, T. M. McMahon, and R. E. Luce. Activegraph: A digital library visualization tool. *International Journal on Digital Libraries, Special Issue on Information Visualization Interfaces for Retrieval and Analysis*, 2005.

[16] K. Perlin and D. Fox. Pad: an alternative approach to the computer interface. In *SIGGRAPH '93: Proceedings of the 20th annual conference on Computer graphics and interactive techniques*, pages 57–64. ACM Press, 1993.

[17] J. Raskin. *Humane Interface: New Directions for Designing Interactive Systems.* Addison-Wesley Publishing, 2000.

[18] H. Reiterer, H. C. Jetter, W. König, J. Gerken, and C. Grün. Zoomtechniken zur Exploration komplexer Informationsräume am Beispiel Hypergrid. To be published in Mensch & Computer 2005: Kunst und Wissenschaft - Grenzüberschreitungen der interaktiven ART.

[19] B. Shneiderman. The eyes have it: A task by data type taxonomy for information visualizations. In *VL '96: Proceedings of the 1996 IEEE Symposium on Visual Languages*, pages 336–343. IEEE Computer Society, 1996.

[20] R. Spence. *Information Visualization.* Adisson-Wesley, 2001.

[21] E. R. Tufte. *The Visual Display of Quantitative Information.* Graphics Press, Cheshire, Connecticut, U.S.A., 1983.

[22] C. Waldeck and D. Balfanz. Mobile liquid 2d scatter space (ml2dss). In *IV '04: Proceedings of the Information Visualisation, Eighth International Conference on (IV'04)*, pages 494–498. IEEE Computer Society, 2004.

[23] C. Ware. *Information Visualization: Perception for Design.* Morgan Kaufmann Publishers, 2 edition, 2004.

[24] C. Williamson and B. Shneiderman. The dynamic homefinder: evaluating dynamic queries in a real-estate information exploration system. In *SIGIR '92: Proceedings of the 15th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 338–346. ACM Press, 1992.