Collaboration in Information Visualization

Fabian Winter

Abstract— In this paper I give a short overview about the research of collaboration in information visualization. Collaboration is examined for developing some models of how people interact and collaborate. Studies showed that collaboration can improve the quality and the time of answers. These models and other research results make it possible to create design-guides that describe what you have to think about configuring a collaborative environment. These guidelines make collaboration more efficient if the collaboration partners are using a well designed visualization tool. New technology affords new ways of collaboration and interaction. At an interactive table for example a small group is allowed to work in a synchronous or asynchronous way on a dataset to discover, collect and share information. Wall size displays are helpful platforms for large group discussions. The Internet allows to create communities like sense.us where everybody is able to analyse data and to discuss about results. So you can work in a distributed way efficiently as well as in a co-located environment. But new technology leads to new problems. Who is allowed to manipulate the dataset and view in a synchronous environment? How to identify the interacting user? Or what about drag-and-drop on a wall-size display? So there is still a need of research.

Index Terms—Collaboration, Information Visualization, computer supported cooperative work, CSCW, Design Guide

1 Introduction

Collaboration is a helpful *tool* in our daily life. If you work, learn or study you have to be able to work collaboratively. But why do you have to work in cooperation with other people in school, in business and so on? In general you can say collaboration is an easy way to solve a complex problem or to collect a lot of ideas. It allows to analyse a big set of data that is to complex for one person. But sometimes the dataset is still to complex for a group to get analysed. In this case you need some kind of techniques to master the complexity of data. Technologies like computers and Internet are able to support collaboration. For instance, bar and pie charts are famous ways of visualising information. But in time of hypervariate data there is a need of new systems that can visualise complex data in an easy understanding way. New technologies like interactive tables and wall-size displays have a need of research, too. They need some applications that are able to use their advantages to make collaboration more efficient. Research tries to solve existing problems and to develop applications that can be used in different collaborative environments. Petra Isenberg [5] for example developed a tree layout for hierarchical data. So each member of a group got the same view on the visualization at a tabletop environment. Another goal of research is to understand the way of how people collaborate. So it is possible to create some models that show how people communicate during the collaboration [3]. With these models you can design applications that make collaboration more efficient.

In this paper first I will give an overview of the research of collaboration and technology. Then I describe a design-guide for applications and some applications itself. After that I will discuss some problems and show possible solutions.

2 Collaboration

In this section I will discuss the aspect of collaboration. First I describe some general things about collaboration and than I give an overview of current research results.

2.1 Characteristics

There are two ways of analysing data: with and without a specific question [3][4]. Without a question means that you go throw the data to discover some interesting connections. Additionally the term collaboration divides into two pairs of characteristics:

- co-located vs. distributed
- synchronous vs. asynchronous

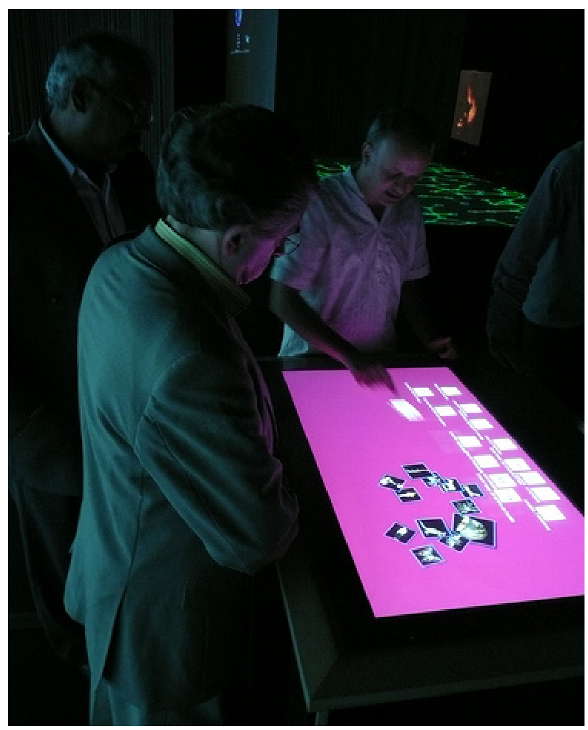

Fig. 1. co-located environment

The properly meaning of collaboration is that a small group is sitting at a table collecting ideas and discussing them to solve some problems or answer a question. You call that kind of work co-located and synchronous because the group's members are at the same location and are working at the same time on a dataset or view of the data. Synchronous views are also called WYSIWIS (what you see is what I see) [16]. However asynchronous means that the users are not working at the same time. Computers and the internet bring new possibilities to collaboration. So you are able to work distributed, too. This means that the group's members do not have to be at the same location. They can work in different rooms, cities or countries. As a result

---

- *Fabian Winter is studying Media Informatics at the University of Munich, Germany, E-mail: fabian.c.winter@campus.lmu.de*
- *This research paper was written for the Media Informatics Advanced Seminar on Information Visualization, 2008/2009*

Table 1. collaboration characteristics

	co-located	distributed
synchronous	x	(x)
asynchronous	(x)	x

of this distributed collaboration also is mostly asynchronous. But despite of this the most used way of collaboration is still co-located and synchronous [16] [5] [7]. *Figure 1* shows such an instrumented co-located environment. There are a few people standing at an interactive table working on one view of data. But not all combinations between co-located/distributed and (a-)synchronous are sensible *(see table 1)*. As mentioned co-located and synchronous is the most common way of collaboration [16] [5] [7]. Because the research of collaboration in information visualization is quite new [6] co-located and asynchronous is possible but there is a lack of applications. Additionally in collaboration people should work in cooperation, so with an asynchronous view there should also be the possibility to switch in a synchronous view for sharing or discussing information [6]. Distributed systems have been introduced in 1994. In such environment asynchronous views are very sensible. However synchronous might be useful for presenting results or for discussions.

2.2 Research Results

Research examines collaboration to understand how people communicate and interact. Mark et al. for example tried to develop a model that shows the decision process of how people answer questions by discovering some data working collaborative on a visualization system [3]. The results in this study are based on an experiment with three configurations: working alone, remoted and co-located on a Smart-Board (both in groups of two). These three groups are splitted again in two groups differentiating the using visualization application. There where two tasks they had to master: finding the answer on a specific question and to find some interesting connections in the given dataset. During the experiment the groups were video taped for analyse their behaviour. The resulting model got five stages *(see figure 2)*. First the question get parsed into important variables. Then one variable get mapped on an equivalent variable in the visualization tool. After that the visualization gets manipulated. In stage four the results get validated. If there are no more variables to be examined an answer on the question becomes expressed. Otherwise, stages two to four were repeated. Such a model allows to understand how people communicate and solve problems working collaboratively. Regarding this you are able to make information visualization applications more efficient.

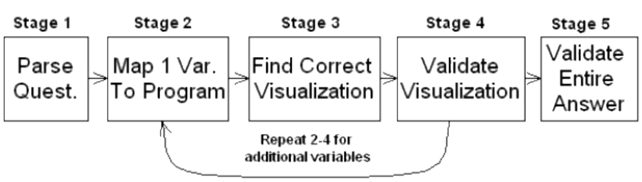

Fig. 2. decision process model by Mark et al. [3]

In a previous study with the same experimental design Mark et al. discovered that: *Groups of two have a greater probability of finding the correct result than an individual* [4]. In free discovery tasks groups create more answers and their results are more meaningful. Other studies showed similar results. [8] proved in an experiment with three different duration (four hours a day, eight hours spread over two days and 24 hours spread over five days) that synchronous collaboration improves the analysing and solving process in time and quality. [1] examined in the study if distributed synchronous collaboration gets improved by using some visualizations. To investigate this there are groups of two. The group's members are remote connected. The groups differentiate in the visualization they are working with: no visualization (all data are on a spreadsheet), unshared visualization (working with a visualization tool), shared view only visualization (the group's members got a visualization tool and are able to see what the partner is doing) and shared full access visualization (the partners can work with to partner's data, too). The group's task was to identify a serial killer supported by their visualization. The finding of this study is that groups with a visualization identified more often the serial killer than groups without a visualization. But the shared view-only visualization is quiet unhelpful. The best is to use a full access visualization. A limitation of this study is that it is not representative amongst others because of the usage of instant messengers for sharing information. Others studies showed that using instant messengers for sharing information introduces some barriers. Using an audio channel might be a possible solution [15].

3 Technology

Information visualization is very important to show people complex data and how they are connected. Often you realise this with simple bar and pie charts. These can be used for showing election results or in business for turnover trends over the last few years. A famous brainstorming-technique is to create a mind-map where you collect all your ideas and where you mark with lines how these are connected. With that technique you are able to collect and structure your ideas on a specific topic. Since computers became more powerful it is possible to collect complex data and show them in an easy understanding way. So now you are able to visualise for example hypervariate data and so on. New technology is able to support collaboration to make it more efficient. Interactive tables for example like in *figure 1* allows to interact with data. There is no more need for static charts. With such a table it is possible to change parameters during the runtime and to see the influence on the data. Users are able to discover data asynchronous and to switch in an synchronous view for collecting results, sharing information and for discussions on their findings.

But there are still more techniques than interactive tables. Another is wall-size displays. With these displays you are able to visualise a very big dataset. You can present your results or collect ideas that get showed in a well arranged way. These can be a big discussion platform, too. [10] developed some kind of interface for using such a display. SAGE allows to visualise remote data throw the internet on a wall-size display. Remote computers render the data or application that should be displayed and send the pixel(-information) to SAGE. With SAGE you can scale and rearrange the single windows on your display. This brings the possibility to the user to analyse data or share applications in high-resolution and supports the asynchronous and distributed collaboration. A limitation of this system is the availability of high-speed network.

[12] introduced another way of interaction: a interactive floor. There are two kinds of interactive floors: sensor- and vision-based floors. Sensor-based floors are well known as dance floors. Weight sensors in large tiles detect interaction and so it is possible for example to dance a given choreography in a video game. Vision-based floors (like in [12]) are more complex. Cameras on the ceiling detect if someone is standing in the interaction area. [12] uses an other technique. The interactive area is a 12 qm large floor of glass. A projector under the floor gives the user feedback while interacting. Cameras beside the projector detect the shadow of users and so their position. A limitation of this system is that it is very difficult to connect shadows to users while jumping and so on. A possible solution might be ceiling cameras what detect users additional to the cameras under the floor to connect shadows and users. Using this technique in collaborative environments might not be sensible. But I wanted to show that research is developing new ways of interacting that possibly can be useful for collaboration in the future.

But not all techniques can be used for all tasks. Wall-size displays are not sensible for interaction with multiple users and interactive tables should not be used to present some results. The best way is to combine these techniques to use all their advantages to make collaboration more efficient. There are a lot of more things that should be regarded. Some of these things I will describe in the next chapter.

4 Design-guide

Petra Isenberg et al. developed guidelines you should follow if you want to design an application or want to create a computer supported collaborative environment [5] [6]. She refers to other papers and collect their results based on three topics of research: hardware setup, information visualization and collaborative environment. This design-guide is intended for co-located collaboration and will be displayed in this section.

4.1 Hardware Setup

Hardware setup describes the configuration of the environment itself where you want to collaborate. There are a lot of things you have to handle with:

- Size of displays
- Kinds of displays you want to use
- Input for interaction
- Resolution for input and output
- System response during interaction

Display size is an important issue. If you want to analyse a big dataset displays are still to small even for single user data discovering. So there is a need of applications that brings the possibility to the user to interact with data to have a look at the whole data. In collaboration that will be a problem, too. A group of two or more persons should be able to discover the data. In an asynchronous environment each member needs to have a view on the data and an interacting possibility, this makes the space problem worse. So in general you can say as the number of group's members grows the size of displays and workspace needs to increase to support a sufficient viewing and interacting area.

Different kinds of displays bring different ways to configure the collaborative environment. Besides of normal displays for each person you can use tabletops, wall size displays and other stuff. The composition of the environment needs to regard the different qualities of the displays, the task you want to work on and the group's setup. Single displays are useful for individual work on the dataset, so for asynchronous collaboration. Interactive tables should be used for working in a small group. They support users in discussing the data based on its visual representation. However wall size displays are helpful in discussions of large groups or for presentations of some results. So it is a quiet important aspect how to configure the collaborative location.

In both synchronous and asynchronous environments each user should be able to interact with the view. In asynchronous collaboration they can interact with local copies of the view but in synchronous interacting could be a problem. If each user is allowed to interact the access to the view has to be coordinated. A possible solution is to implement multi-focus views where users can mark a view state and can switch between the different states quiet quickly. Additional the identification of the interacting users would be preferable.

Resolution is a big challenge. A disadvantage of large displays is their almost bad resolution. In this case you need to re-design your application to provide a good readability of texts, colours and so on. But input interfaces can have a bad resolution, too. Interacting by fingers or pens is not that accurate as using a mouse. Small items on a high resolution display with low resolution input are difficult to use.

During interaction the user needs quick response so he can see the result of the input. Using a large high resolution display showing a complex visualization is very costly to render. So powerful hardware is a must-have for computer supported cooperative work.

4.2 Information Visualization

In computer supported cooperative work applications do not have to regard just on the technical side of environments. Visualising information is an important aspect to make collaboration more efficient. The way how to visualise depends on the underlying data and task. But there are several other things you have to mind:

- Support creation of mental models
- Different representations of data
- View state history
- Perception

In not computer supported environments users tend to create a mind-map or so called mental models of their ideas to categorise them and to see how they stand in connection [5] [6]. So applications should allow users to organise data and information by themselves to create categories and mental models. This is useful to get an overview of the data, to collect ideas and to discuss possible solutions.

Each person is an individual. Everybody has his own preferences how to work and interact with visualised information. This is a reason why applications should support different representations of the same data. So the user is able to chose the visualization that fits the best. This causes a better and more individual way of interacting with data but complicates the communication of the group. If someone pointed out something interesting it is harder to show it other users. A possible solution is to highlight the point by a marker that appears at the others views, too.

By switching between individual work and shared views group members might lose track of interacting. Working on complex datasets and views it might be hard to pick up the track. A way to preserve users from catching up with their work is to support a view state history. This history saves the different states a user worked on. In the case of losing the track the user looks up his last steps and continues his work.

Working at an interactive table each group member got a different angle respectively perspective on the table. There is still not evaluated how different views on a display influence the admission of information in collaboration. But *"A study by Wigdor et al. evaluted the effect of viewing angle on different graphical variable and suggest that care should be taken in positioning and choosing the appropriate visual encoding as some graphical elements are more robust to distortion than others"*. It seems to be clear that looking at some kinds of charts is more comfortable by not reading them upside down.

4.3 Collaborative Environment

Designing a collaborative environment includes two main topics:

- Coordinating the group's activities and
- Supporting the communication

Applications have to support some features a group needs to work as a team. This makes collaboration more efficient.

In single-user systems often a fixed layout of the workspace is given. This is not a advisable way for teamwork. In collaborative environments users tend to divide their workspace in personal, group and storage spaces. This allows them to discover data in their own space and to discuss, share and collect information or results on the group's workspace.

Complex dataset and view lead to a big list of parameters to manipulate the view and data for explore themselves. Those actions like zooming in or filtering of information makes the system difficult to understand and hard to use because the system transparency becomes less (see next section). To design an easy to use application the user does not have to do a big shift of input mode. Also the number of manipulating dialogs should be as little as possible. This guarantees that the user is able to interact with the system quiet quickly and does not have to read the manual before interacting.

Working on one dataset brings the problem of right- and access management to the user. Deleting information and manipulating a shared view are scenarios that have to be managed. Additional if something changed each user should be notified. There is no general solution. The right behaviour depends on the tasks and kind of application you handle with.

Studies evaluated that users tend to work individually if they got the possibility. So applications should support both individual and cooperative work. This allows the users to solve parts of a problem parallel and to collect afterwards their result as a group. An other scenario can be that users are working parallel on the same problem for example if it is a quiet difficult one and then they discuss their results on the shared workspace.

As mentioned notifying the users because data or settings changed in a shared view during working individually is very helpful. There should also be the possibility to compare different views and to make annotations on the data that appear at all other views. So you can easily add information to some data or add some question that can be discussed. You always have to mind that users should not be forced to across or reach into another users workspace.

Based on these guideline Isenberg et al. developed a visualization for co-located collaboration. This and other applications are described in the next section.

5 Applications

Since computers become powerful to support collaboration they were used to visualise information in an easy understanding way. But as mentioned in the previous section there are more things you have to regard designing an application than the way of representing data. For example the support of communication and interactive analysis of data. Petra Isenberg shows a matrix to describe the emphasis of some available applications *(see figure 3)*. Additional she divides these applications into three different skills a user needs to be able to work with this program. These efforts are called expert developers, savvy designers and novice consumers. Applications mostly got their focus on one of these goals and skills. But there is a trend that leads to more flexible programs [6].

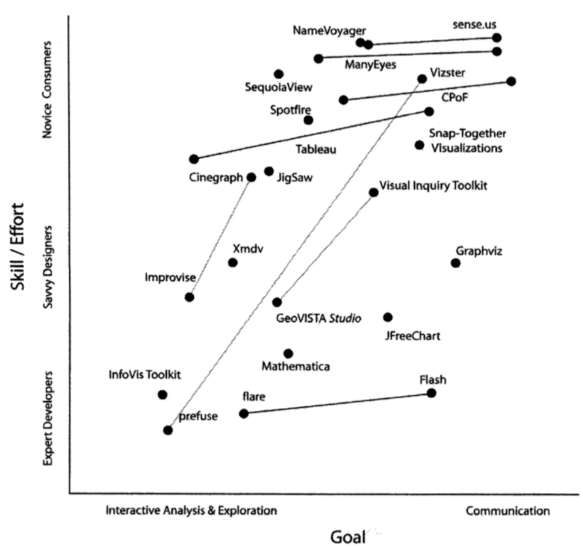

Fig. 3. Analysing applications [6]

Flash, a well known animation application, is classified as an expert communication tool. Because of it's time-line based environment and there are no developer tools for data analysis it is not useful for dynamic exploration of data. So it is still used for interactive graphics on the internet [6].

Spotfire and sense.us are other programs that were classified. In the following I will tell something about this applications and will describe Petra Isenberg's tree layout.

5.1 InfoZoom and Spotfire

The first applications that were designed for data analysis were developed for single users on a desktop environment [5]. Two of these tools are InfoZoom and Spotfire. They are quiet equal but there is one big difference: the system transparency.

System transparency describes the way of manipulating a program. As the number of options and property dialogues grows the transparency decreases [9]. InfoZoom is a high transparent application. It supports three views: wide, compressed view and overview mode. In each view all values and information are displayed. This affects the comprehensibility of the data [9]. Spotfire in contrast is not very tranparent. *Figure 4* shows Spotfire. There are several types of visualisations you can use for data analysis parallel. Bar and pie charts plus scatterplots and graphs are just a few of them. You are able to interact with the data just in time. You can select and deselect values and enlarge and diminish ranges of data that should be shown. So there are a lot of options and properties you can change.

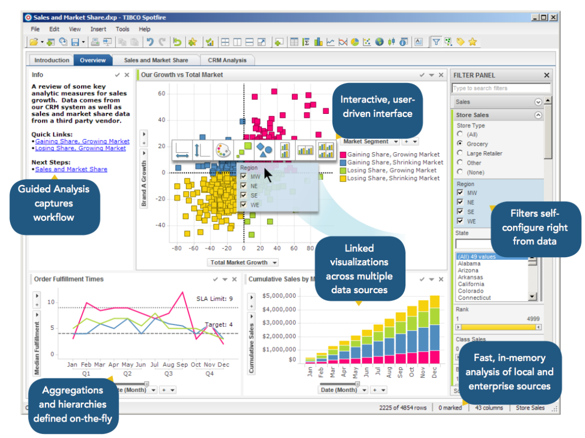

Fig. 4. the spotfire application [17]

As a result of this if you want to use this application you have to orient oneself. [9] referred this as cognitive setup costs because first you have to make your settings and than you can start with analysing data. Sometimes this process takes a long time because you do not find straightaway the right view.

The study of [4] showed that using an application with a higher system transparency leads to more correct answers on a task than using an application with lower transparency. While working on a task of free data discovery using programs like InfoZoom produce more (correct) answers whereas programs like Spotfire produce more meaningful respectively complex answers. But in general you can say working collaboratively using a visualization tool is more efficient than working alone on a dataset.

Despite it is hard to use them in a collaborative environment. As a single-user application you might be able to open more than one instance of the program or to use it distributed so that each group member is allowed to discover the data on it's own. But changing or deleting data might be a big problem. Other users do not become notified that something changed and so they are working on a dated version of the data. Because of this lack of communication possibilities these two programs are classified as exploration tools *(see figure 3)*. In collaborative environments configured with new technologies there might be an other problem using these programs. Because they were designed as desktop applications they do not support multi touch. On an interactive table for example they can be displayed and used but just by one person. Additional depends the person's perspective on the program on the sitting position at the table. This makes them not sensible for tabletops but for wall-size displays.

In collaboration you can say these tools can be used co-located and synchronous but this raises the question of who is allowed to manipulate the view and dataset. So they might be useful to give an overview on the data and establish a basis for group discussions on wall-size displays.

5.2 sense.us

[7] introduces a distributed and asynchronous web based application: sense.us. As a multi-user application it is designed for a general audience with an emphasis on communication *(see figure 3)*. It includes an amount of visualizations of US census data over the last 150 years. Users are allowed to discover the dataset on their own and to discuss their findings with other users. Because of it's focus on communication the four main features are:

- doubly-linked discussions
- graphical annotations
- saved bookmark trails
- social navigation

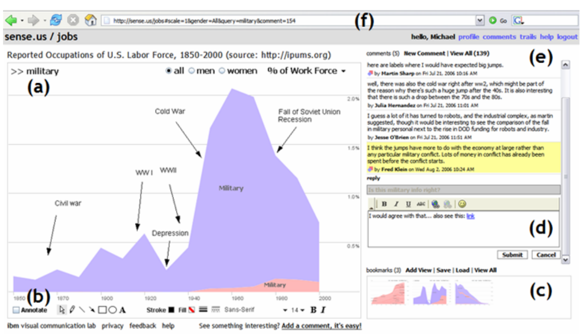

Fig. 5. the web based application sense.us [7]

Figure 5 shows the sense.us system. Different features are marked by a letter. *Section (a)* shows the main feature of this application: the graphical data visualization. There the current set of data is shown and you are allowed to change some parameters just in time for a better exploration of data. Because of the application's focus on communication you can write a comment on the current state of the visualization *(section (d))*. Other users analysing the same visualization are able to see the comment in *section (e)*. The meaning of doubly-linked discussions is that you are allowed to do both writing a comment that links to a view and discovering a view that links to a discussion on itself. But there are more possibilities to communicate than writing a comment. Graphical annotations are very helpful to show other users some interesting things in an easy understanding way. Writing a comment on a visualization you can make some of these annotations that were linked to your comment, using the annotations bar in *section (b)*. You can use different 2D-Objects like lines and rectangles. Arrows, Text and freehand paintings are also possible to create your annotation. Clicking on the comment the annotations become displayed to the user in *section (a)*. So you can show easily what you are talking about or to give some non displayed information. The URL in *section (f)* always displays the current state of the view. Changing a parameter changes the URL. This allows to use the browser's back and forward buttons for an easier navigation through the visited visualizations. Using the URL permits to use the browser's bookmark function. But the application includes such a feature, too. Bookmarking the current state of view a little thumbnail becomes displayed in *section (c)*. With these bookmarks you can compare different views or data discovering some interesting connections or add them as a link in your comments. Social Navigation means the availability of profile pages where all comment and views the user works on are shown. Additional the application includes comment listing pages with all added comments. These might be quiet a lot so you are able to sort and search for a comment and to filter for comments on a specific view.

In conclusion sense.us is a good example for a distributed and asynchronous collaborative application. But there are still some issues. For example if the data changes, and so the visualization, what about the graphical annotations? Despite a user study showed that users enjoyed exploring data and that the four main features are very helpful [7].

5.3 Isenberg's Tree Layout

Based on Petra Isenberg's design-guide she developed a tree layout bearing the guidelines in mind [5]. But I have to note this is just a layout and not a stand-alone application that can be used for data analysis. With this layout I just want to show a possible implementation of the guidelines.

5.3.1 Hardware Setup

The section hardware setup is not important describing the tree layout. Despite I want to give a short summary of the used environment to show in a detailed way the implementation of the guidelines.

For interaction there is used a touch-sensitive tabletop display. With a resolution of approximately six mega pixels the environment can be used by small groups of two to four people. Because of the maximum allowable touches of two and the size of the table, groups of five and more are not sensible. Interacting users can not be identified. But the layout can be used on wall size displays, too.

5.3.2 Information Visualization

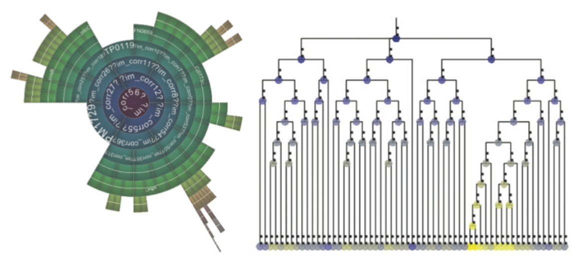

Fig. 6. two tree layouts by Petra Isenberg et al. [5]

The designed system supports hierarchical data that become displayed as a radial tree or alternatively as a cladogram. *Figure 6* shows on the left an example of the radial tree and on the right the associated cladogram. The main advantage of a radial tree is that each group member sitting at the table is allowed to look as good as other users at the visualization no matter of the sitting position. A cladogram in contrast ist useful to be displayed on a wall size display for discussions and so on. All visualizations can be freely positioned , zoomed and rotated on the screen and each of them got it's own menu buttons.

Supporting mental models is a very important feature an application should include. With these models you can collect all important information to get an overview of all relevant data and to create a possible solution on a specific question. With Isenberg's system you are allowed to pool different visualizations into a container. These containers of visualizations can be easily scaled and repositioned. Visualizations can be added and deleted just by moving into and out of a container. But these containers pool only whole visualizations and not single information. So containers can be used to summarise relevant visualizations but not to collect important connections between them.

Individual preferences on a view are supported by the two different visualization types: radial tree and cladogram. By dragging on a special widget the visualization becomes changed. This is quiet complicate because each user needs such a widget. Else he has to walk around the table to the place where the widget is to change the representation. Or he has to disturb other users during work to hand the widget to the user. A better solution might be to add an additional button to the menu bar of the visualizations.

5.3.3 Collaborative Environment

Supporting different kinds of collaboration is a very important feature. With Isenberg's system it is possible to work both synchronous and asynchronous. Users can work individually on a copy of the dataset and can switch easily to cooperation by scaling a visualization to get a closer look. The number of visualizations is limited by the using graphics hardware.

As mentioned as lot of visualization applications are design for desktop environments. Using these on a tabletop is not sensible because they do not support the special features and the different workflow. So visualization parameters should not be changed by dialogues and fixed menus. The implemented system uses widgets to change these parameters. For example to change the colour of a visualization you have to drag it on a colour-widget or the other way around. This allows that each user got it's own set of changing widgets that can be freely positioned on the screen.

Communication is a very important aspect that each collaborative data analysis application should support. The presented system allows to create annotations to show other users interesting things or to add intermediate results. Using the finger as a low resolution input so called sticky notes are useful annotations. They look like post-it's that point to an information or contain some kind of result. Using a pen as a higher resolution input you can paint your annotation like arrows and so on directly into the visualization.

In general you can say Isenberg supports a lot of features that an efficient collaboration environment requires but there are still a few issues that need to be worked on.

6 Discussion

The research of collaboration in information visualization is a quiet new topic [6]. As mentioned there are a lot of things you should regard designing a visualization application or configuring a collaborative environment. I described a few more things that still have a need of research. But especially using new technologies like wall size displays causes some problems I will explain in the following. [2] gives a short overview of some of these problems and proposes a few solutions.

Using big displays implicates that the user has to move the mouse on the screen over a large distance. On displays with a lower refreshing rate or a bigger size while moving the mouse fast the cursor seems to jump over the display. This increases the possibility of losing track of the cursor. [13] introduces a possible solution: the high-density cursor. High-density cursor adds additional cursor images between the previous and current cursor position during one refreshing time. If the screen becomes refreshed high-density cursor adds new cursor images if needed. This technique is alike the windows mouse trail.

In a multi display environment there is the problem of different display sizes, resolutions and offsets. Moving the cursor from one display into an-other the cursor's position becomes *warped*. So targeting across multiple screens becomes very difficult. Mouse ether, described in the paper of [14], might be a possible solution. Before starting you have to configure mouse ether so that the system gets to know the screen sizes, resolutions and positions towards each other. After that you are able to move the mouse trough the non displayed space between two screens, the so called ether. This allows to move the cursor from a display to another more precisely. If the cursor gets lost the system moves it to the next visible position on a screen.

Drag-and-drop and moving operations on interactive wall size displays is quiet difficult. The user needs to walk along the display to the destination where he wants to drag or move. There are still some solutions to solve this problem [11]. Push-and-throw for example displays during the drag operation a half transparent miniature of the desktop. The user can drag onto the miniature. While dropping the dropped icon moves to the according position on the desktop. As an disadvantage the resolution of this operation is not that good. Drag-and-pop is an-other approach. Here while dragging compatible icons are displayed near to the cursor position. For example while dragging a html document, the trash bin, folder and browser icons are displayed. Moving the icon to an other position is not supported. Push-and-pop is a combination of both approaches. There a desktop miniature including the compatible icons become displayed. The user can drag the icon on an-other or on a position. This allows to drag and move icons. In a study [11] push-and-pop has been almost the fastest technique.

In conclusion apart from a few issues research is on a good way to make collaboration more efficient. Collaboration itself is a well examined topic. Some problems like deleting and manipulating view and data in a synchronous collaborative environment can not be solved because they depend on the underlying task. Isenberg's implementation of her guideline still has some constrictions. So now it is time to develop applications and to advance interaction based on the explored design-guides to use the features and advantages of new technologies.

References

[1] S. K. Aruna D. Balakrishnan, Susan R. Fussell. Do visualizations improve synchronous remote collaboration? In *CHI 2008 Proceedings - Visualization to Support Information Work*, Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh, PA 15232, USA, April 2008. Human Computer Interaction Institute, Carnegie Mellon University, ACM New York, NY, USA.

[2] P. Baudisch. Interacting with wall-size screens. In *CHI 2006*, Microsoft, One Microsoft Way, Redmond, WA 98052, USA, 2006. Microsoft Research.

[3] A. K. Gloria Mark, Keri Carpenter. A model of synchronous collaborative information visualization. In *Proceedings of IV03 7th International Conference on Information Visualization*, pages 373–381. Department of Information and Computer Science, University of California, Irvine, July 2003.

[4] V. G. Gloria Mark, Alfred Kobsa. Do four eyes see better than two? collaborative versus individual discovery in data visualization systems. In *Proceedings of the Sixth International Conference on Information Visualisation*, pages 249–255. Department of Information and Computer Science, University of California, Irvine, 2002.

[5] P. Isenberg and S. Carpendale. Interactive tree comparison for co-located collaborative information visualization. *IEEE TRANSACTIONS ON VISUALIZATION AND COMPUTER GRAPHICS*, pages 1232–1239, 2007.

[6] H. J., H. F., S. Carpendal, W. C., and I. P. Creation and collaboration: Engaging new audiences for information visualization. *Lecture Notes In Computer Science*, pages 92–133, 2008.

[7] M. W. Jeffrey Heer, Fernanda B. Voyagers and voyeurs: Supporting asynchronous collaborative information visualization. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 1029–1038. ACM Press New York, NY, USA, April 2007.

[8] A. J. S. e. John M. Linebarger. Benefits of synchronous collaboration support for an application-centered analysis team working on complex problems: A case study. In *Proceedings of the 2005 international ACM SIGGROUP conference on Supporting group work*, pages 51–60. ACM New York, NY, USA, 2005.

[9] A. Kobsa. An empirical comparison of three commercial information visualization systems. In *IEEE Symposium on Information Visualization*, pages 123–130, 2001.

[10] R. J. e. Luc Renambot, Byungil Jeong. Sage: the scalable adaptive graphics environment. In *Proceedings of WACE*, pages 2004–09, 2004.

[11] P. B. e. Maxime Collomb, Mountaz Hascoeat. Improving drag-and-drop on wall-size displays. In *Proceedings of Graphics Interface 2005*, pages 25–32. Canadian Human-Computer Communications Society School of Computer Science, University of Waterloo, Waterloo, Ontario, Canada, 2005.

[12] K. J. K. e. Ole S. Iversen. Igamefloor: a platform for co-located collaborative games. In *Proceedings of the international conference on Advances in computer entertainment technology*, pages 64–71. ACM Press New York, NY, USA, 2007.

[13] G. R. e. Patrick Baudisch, Edward Cutrell. High-density cursor: an improved mouse cursor that accelerates target acquisition. unknown.

[14] K. H. Patrick Baudisch, Edward Cutrell and R. Gruen. Mouse ether: Accelerating the acquisition of targets across multi-monitor displays. In *Conference on Human Factors in Computing Systems*, pages 1379–1382, Microsoft, One Microsoft Way, Redmond, WA 98052, USA, April 2004. ACM New York, NY, USA.

[15] P. Scupelli, S. Kiesler, S. Fussell, and C. Chen. Project view im: a tool for juggling multiple projects and teams. In *Conference on Human Factors*

in Computing Systems, pages 1773–1776. ACM New York, NY, USA, 2005.
[16] D. Snowdon. A subjective virtual environment for collaborative information visualization. *Virtual Reality Universe*, 97:2–5, 1997.
[17] tibco. *Spotfire Overview*, November 2008.