Assessing the Usability of an Interactive Information Visualization Method as the First Step of a Sustainable Evaluation

M. Rester[1*], M. Pohl[1], K. Hinum[2], S. Miksch[2], S. Ohmann[3], C. Popow[3], and S. Banovic[1]

[1] Institute of Design and Assessment of Technology
[2] Institute of Software Technology and Interactive Systems
Vienna University of Technology, Vienna, Austria

[3] Department of Child and Adolescent Psychiatry
Medical University of Vienna, Vienna, Austria

[*] markus@igw.tuwien.ac.at

Abstract:
Gravi++ is a visualization method that is designed to support psychotherapists in their work with anorectic girls. During the therapy complex and time dependent data have to be analyzed. Statistical methods cannot be used in this context, and visualizations seem to be a viable alternative. Gravi++ is based on a spring metaphor. It can represent time dependent data easily.
There is still too little systematic and empirically validated knowledge to support the design of such information visualizations. Therefore, extensive evaluation is necessary. The evaluation process is composed of two steps – a usability study and the evaluation of the Gravi++ method as such. The following paper describes the usability study. Methods used were usability inspection / guideline review, heuristic evaluation and focus groups. Heuristic evaluation was a very valuable method for identifying usability problems. Focus groups did not yield very much additional factual knowledge but gave important insights about the subjective importance of usability problems.

Keywords: H.5.2 [Information Interfaces and Presentation]: User Interfaces — Evaluation/Methodology; J.3 [Life and Medical Sciences]: Medical information systems; H.5.1 [Information Interfaces and Presentation]: Multimedia Information Systems — Animations, Evaluation/Methodology; Interactive Information Visualization, Usability.

1. Introduction

Gravi++ is a visualization tool that is designed to support psychotherapists in their work with anorectic girls. During the therapy of these girls a large amount of highly complex data is collected. Statistical methods are insufficient for the analysis of these data because of the small sample size, the high number of variables and the time-dependent character of the data. Only a small number of anorectic girls attend a therapy at any one time. The girls and their parents have to fill out numerous questionnaires before, during and after the therapy. In addition, progress in therapy is often not a linear process but a development with ups and downs. All this indicates that visualizations might be a more appropriate method of analysis than statistics. The aim of the therapists is to predict success or failure of the therapy depending on the results of the questionnaires, and, more generally, to analyze the factors influencing anorexia nervosa in more detail. To do this, they have to find patterns in the data. Gravi++ was developed to support them in this work.

Several authors have underlined the importance of the empirical evaluation of visualization techniques (see e.g. [2]). In the past few years, usability studies concerning visualization methods have become more frequent, and valuable information about the design of such systems has been gathered. Colin Ware's book "Information Visualization – Perception for Design" [23] gives a comprehensive overview about the foundations and principles of so-called sensory representations, that is representations that can be processed by the human brain without learning. Our knowledge about arbitrary representations, that is representations that have to be learned and which are embedded in a specific culture, is less systematic although there are many investigations concerning specific issues. The following paper describes the development and evaluation of an information visualization system – Gravi++ – with special emphasis on usability aspects and arbitrary representations.

2. Gravi++

2.1. Concept

The human perceptual system has a remarkable ability to organize and locate things spatially, judge comparative sizes, distinguish between a large range of colors and patterns, and perceive motion [15]. Gravi++ [4] tries to utilize these human capabilities by positioning icons on the screen. There are two kinds of icons representing (1) patients and (2) questions from the questionnaires respectively. Every patient is attracted by the questions according to the answer she gave. This is modeled with a spring-based system. Every question is connected with every person by a spring (see the upper left illustration in figure 1). The strength of the individual spring depends on the answer the patient gave. This way, every persons' icon position on the screen identifies how she answered the respective questions. This leads to the formation of clusters of persons who gave similar answers (see the lower left illustration in figure 1).

The size of a person's icon can be mapped to any additional parameter (for example to the body mass index of the patients or to the attraction force). In the second case the sphere is larger if it is attracted by higher values. This feature helps to discriminate different icons that are attracted by the same values with a different coefficient. By displaying persons with transparent colors, overlapping icons can still be distinguished.

To visualize the changing values over time, Gravi++ uses animation. The position of each person's icon changes over time allowing tracing, comparing and analyzing the changing values.

Alternatively the change over time can also be represented by traces (see screenshot in figure 1). The size and path of the person's icon is shown corresponding to all time steps or only to a restricted subset (i.e. the previous and the next time step).
To visualize the exact values of each question, rings around the question's icon can be drawn. The ring size corresponds to the attraction to the question. To avoid overlapping rings with the same value, they are put closely side-by-side.
In addition, Star Glyphs [22] can be shown, which communicate the exact values. The edges of the Star Glyph are connected to the corresponding question rings and both are drawn in the same color as the person’s icon. This helps the user to identify the corresponding person.
Missing data is handled by the system in two ways. If a person has answered no questions at a specific time step, the icon of the person becomes transparent. If a value of the size of a person's icon is missing, the icon is shown with a special marking.

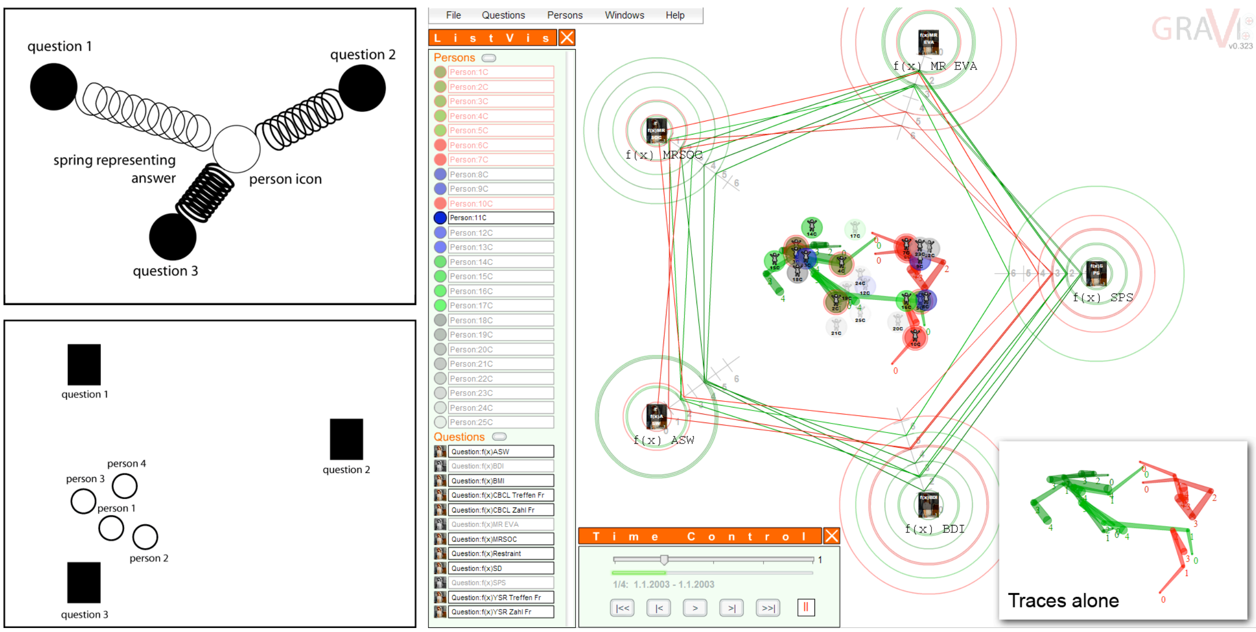

Figure 1. Two concept illustrations and a typical screenshot of Gravi++ (with all sub visualizations enabled).

2.2. Interactions

Gravi++ provides various interaction possibilities to explore the data and get new insights. The icons and visual elements can be moved, deleted, highlighted and emphasized by the user. Each change leads to an instant update of the visualization.

2.3. Implementation

Gravi++ was implemented in Macromedia Flash and features two visualizations that work closely together. There is an overview visualization (ListVis), to select a subset of a large data set. This subset can then be analyzed with the main visualization Gravi++. The data exchange between these two modules is implemented by drag and drop. Both modules support linking and brushing.

3. Theory and Background

The usefulness of an InfoVis tool is not as predictable as that of traditional software systems, because of the remarkable influence of human reasoning processes on the success of the application. So even after careful design and development the outcome has to be evaluated in great detail.

Nevertheless, some authors posit that there is still too little research paying attention to the users' needs and human cognitive processes (see e.g. [21]). Tory and Möller state that interest in such topics has grown in the past few years but "that major contributions in this area are limited" [21]. They also describe guidelines that should inform the design of user-friendly visualization systems. Craft and Cairns [3] especially comment upon the lack of systematic knowledge about design methodologies. In their view, empirically sound and generalized knowledge about design methodologies is necessary. Guidelines like the well-known Visual Information Seeking Mantra developed by Shneiderman [18] should be validated empirically and made more useful for practitioners by adding context information.

As long as generalized and empirically sound information about design methodologies for information visualization is lacking, it is all the more necessary to test new systems in this area carefully. There are several methods for testing such systems. Plaisant emphasizes the importance of case studies in natural settings and longitudinal studies with the actual users of the systems [16]. Such studies are, of course, difficult to achieve because the actual users of visualization systems usually do not have very much time for user testing, and longitudinal studies over several years are very difficult. Nevertheless, we think that any input of actual users and information about the context of usage will be valuable. Tory and Möller point out that the aim of the study should influence the choice of methodology. They think that usability inspections should take place at the beginning of the development process and rather yield information about the user interface design. In contrast to that, user studies with human users should take place later on and will yield more concrete information about the quality of information visualization method as such. North and Shneiderman's study [14] is a good example for an analytical separation of the investigation of usability aspects and the study of the information visualization method as such. This seems to be all the more necessary as an otherwise good information visualization system might suffer from a bad interface. Therefore, a dedicated usability study early in the design process seems to be necessary for a thorough testing of the system.

Some authors have conducted empirical evaluation studies about systems similar to our own. The results of these studies have informed our development process. Kadaba et al. [6] made a detailed investigation about the use of animation for the representation of time dependent processes. They state that there are not many visualizations that use animation for showing change over time, but the existing studies have positive results. In their own study animation was also used successfully. Morse et al. [11] developed a visualization method called VIBE similar to Gravi++, which also uses a spring metaphor. They compared VIBE to word lists, tables, multidimensional icons and scatterplot graphs. In a detailed user study they found out that VIBE is especially advantageous for more complex queries.

Medical highly structured data, such as psychotherapeutic data, impose an additional challenge due to their complexity and their temporal dimensions. Various visualizations like Worm Plots [10], Zoom Star solution [13], TimeWheel [20], Table Lens [17], Stardinates [8], and LinkVis [9] exist that visualize such kind of data utilizing different approaches. We decided to emphasize the exploratory nature of finding new interdependencies (such as predictors) and support this by multiple interaction possibilities to explore the temporal data thoroughly.

4. Experimental Setting

The study described in this paper is part of an ongoing project to develop methods for the analysis of complex and time dependent data from a large number of questionnaires. In the course of this project the information visualization method Gravi++ was developed as one of the methods of analysis. Because of the lack of systematic guidelines for information visualization Gravi++ is tested extensively. In a first step, the usability of the system is tested to preclude that usability problems are mixed up with problems with the visualization method as such. In a second step, the method as such will be studied, among others with the actual users in their own offices. In the following, the results of the first – the usability – study are described.

4.1. Description of the Sample

The subjects of our study were 27 students of thirteen different but mostly informatics-related studies: informatics (17 B.Sc., 4 M.Sc.), Business Informatics / Information Systems (M.Sc. 1, Ph.D. 1), M.A. in education (3), and Surveying and Mapping (M.Sc. 1). There were 37.04 % female and 62.96% male participants.

It is a generally accepted view that usually 3-5 expert evaluators are enough to find most usability problems [5]. The fact that the subjects were students makes sense in our case because about half of them already attended lectures on user interface design and/or usability engineering. All of them could be easily familiarized with the field of usability evaluation and therefore can be described as semi-experts in this field.

Although the domain of the application is extremely specific, a rather coarse overview seems enough because on the one hand domain experts took part in a participatory software design and development process, and thus allowed for taking care of very specific usability glitches only recognizable by such in-depth experts. On the other hand the large number of subjects should guarantee the acquisition of relevant data for improving the inspected software.

4.2. Methods

Three different approaches in gathering information for the improvement of usability were adopted all of which targeting different possible insights: usability inspection/guideline review, heuristic evaluation, and focus groups.

4.2.1. Usability Inspection / Guideline Review

The purpose of this very informal way of usability evaluation was to identify easy to find problems. Tognazzini's rather specific principles and guidelines for the design of interfaces [19] as well as Nielsen's very general heuristics [12] formed the basis for this procedure.

Although performed by only one expert, 31 problems were located, one of which deals with inconsistent notations. 27 variant notations in the menu, the toolbar and other locations throughout the software were found. Especially these inconsistencies were carefully remedied in order not to bother the student evaluators to a large extent and to hinder them in finding more complicated problems and to make it easier to cope with the rather complex domain.

4.2.2. Heuristic Evaluation

Nielsen's Heuristic Evaluation [12] is one of the most recognized usability evaluation techniques. A small group of evaluators checks the compliance of given software against 10 usability principles. These principles are of a very general nature and therefore called "heuristics". As almost any other usability assessment method heuristic evaluation should take place in an iterative software design and development process. This method suits our needs in various ways:

- It serves as foundation for the students to deal with the subject of usability and enables them to consider more abstract principles.
- The heuristics, which were available to the students while the experiment both as handout and as part of the report-system, built an excellent link to the learned matter.
- The procedure encourages the students to compose concise and accurate reports on found usability problems and therefore ensures a high-quality evaluation even with semi-experts.
- Every method affects the outcome with its framework. The comprehensiveness of the 10 principles assures the minimization of otherwise probably overlooked areas. Students were motivated to concentrate on the broader meaning of the general description and not to get distracted by the narrow focus of the examples for the principles.

4.2.3. Focus Groups

At focus groups (see e.g. [7]) subjects discuss and comment on their experience with a given topic. They represent a form of group interview. Although there is an important difference between these two: group interviews concentrate on questions of the researcher and responses of the subjects while focus groups are based upon the interaction between the subjects on a given topic.

Focus groups can be rather efficient compared to interviews with the same number of subjects. The strength of this method lies in the interaction between the subjects. This can lead to the finding of problems, which would not have been anticipated by any of the participants individually. The very same characteristic holds a risk one has to deal with: the group-interaction can lead to groupthink. Therefore it is important to conduct at least two focus groups. Also the size of the groups should not be too big so that every participant can take active part in the discussion.

As with classic interviews it is very important to have a well-structured guide of questions at hand ensuring the collection of valuable material. We compiled a catalogue of 15 questions (6 on general impression, 3 on menu, 4 on windows, 1 on context menus, 1 on workspace), which we used for sessions of 90 minutes.

4.3. Lab Setting

In sessions, lasting three hours each, Gravi++ v0.300 was evaluated in May 2005 on seven different computers (from an iMac G3 400 up to a PowerMac G4 dual 2x1250). The visualization ran in full-screen mode of Macromedia Flash Player v7. In the beginning the procedure was once more explained to the subjects. They also received handouts summarizing the heuristics, stating typical tasks performed in the visualization, and explaining in detail the steps of the evaluation procedure. In addition to that, the subjects could at any time ask the investigator if they needed any further help.

4.3.1. Report System

The subjects were assisted in documenting their findings by a report system we developed (see figure 2). It supported them in generating high-quality reports according to the specification by heuristic evaluation. It was for example impossible to finish a report describing a usability

problem without stating the principle violated. The subjects were able to choose a principle of the heuristics but they could also specify any other usability guideline they knew of.
We asked the subjects to take a screenshot of the respective area (as small as possible, as large as necessary) and added the possibility to attach this screenshot to the report. Of course, more than one problem description could be made as every screenshot could contain several violations. Adding the screenshot was not mandatory.
The system was implemented from scratch using Perl and MySQL. It also offers administrative access to the collected reports and supports the investigation with many classification features for further analysis (see also outlook / future work).

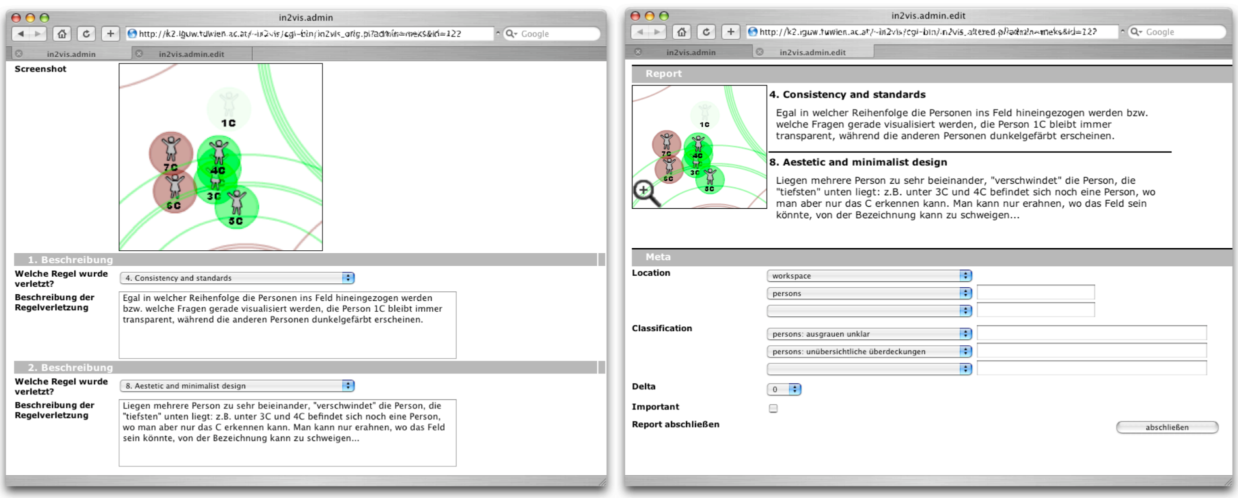

Figure 2. Typical report generated by one of the subjects showing a screenshot of the problem and two problem descriptions of violations of usability principles (on the left). Administrative access to the same report with multiple classification options for the investigator (on the right).

5. Results / Findings

5.1. Lab Experiment (Heuristic Evaluation)

Using the report system the 27 subjects generated 447 reports in which 513 violations/problems are documented. The evaluators show tremendous difference in the quantity of their work (max. 41, min. 5, avg. 19). With each problem description a corresponding rule had to be stated (see table 1).
As mentioned above the 10 principles of Nielsen's heuristics are supplemented by one more alternative where the subjects could enter their own specification. This category was mainly used to document bugs, which in most cases caused a restart of the visualization. 65 out of 75 statements of this category were meant as such bug-descriptions. For the remaining 10 statements the subjects chose to give their own specification of a usability principle. The amount of bugs does not seem surprisingly high if one takes into account the rather early stage in the development of the software where this evaluation took place (Gravi++ v0.300). In a manner of speaking this part of outcome can be described as extensive software testing as the descriptions clearly stated when, where, and how the errors occurred.

Rule	Mentions	Percentage
1. Visibility of system status	63	12.28
2. Match between system and the real world	40	7.80
3. User control and freedom	59	11.50
4. Consistency and standards	105	20.47
5. Error prevention	23	4.48
6. Recognition rather than recall	19	3.70
7. Flexibility and efficiency of use	32	6.24
8. Aesthetic and minimalist design	52	10.14
9. Help users recognize, diagnose, and recover from errors	12	2.34
10. Help and documentation	33	6.43
11. Other Rule	75	14.62
	513	100.00

Table 1. Stated violated principles (Nielsen's heuristic 1-10) in 513 problems found by 27 subjects.

The frequency of assigned principles is affected by multiple factors, including:

- the quantity of true existences of violations in the given category,
- the understanding of the respective principle by the subjects,
- the difficulty of tracking down problems of this sort, and
- the different levels of domain knowledge needed to find problems of different categories.

So it happens that for example the principle "help and documentation" meaning that it is necessary to provide a help system of some sort (which should be searchable, focused on the user's task, etc.) is very easy to understand and very unambiguous. Furthermore, it was very easy to spot in our case, because the rightmost menu entry is named help (following the "standard" part of rule number four) and help simply did not exist at that early stage of development. Therefore 20 subjects filed reports specifying this rule in 33 problem descriptions of which 19 are pinpointing that a help system is generally missing.

On the other hand the situation is quite different when it comes to rule number nine "help users recognize, diagnose, and recover from errors". This rule was least frequently assigned to discovered problems (2.34%). While this rule states rather clearly how good software should deal with errors ("diagnose and recover" meaning useful and understandable error messages including hints to avoid these), the "recognize" part is the tricky one. It asks the software for communicating the fact that either the user made a mistake or some other application related error occurred. How could an evaluator, who runs into a situation where the system does not respond as expected or the response is simply unclear to the evaluator, know that the reason is a missing error message? Maybe the user sets an action that should not be possible. Then there are two options: If the software unintentionally accepts this action, it could be described as a bug. If the software refuses to react as it is supposed to do but misses to communicate fact and/or reason of the refusal, then it hinders the user to recognize the set action as an error. Both scenarios have in common that an evaluator can hardly guess that this should be specified as rule number nine.

From the 12 problem descriptions referring to rule number nine only one anticipated correctly that an error message was missing. Another single one does the same but in this case it was merely a much too weak feedback in terms of rule number one “visibility of system status”. Then there are 5 descriptions with regard to unclear error messages (“diagnose” and “recover” parts of the rule). Another 3, in which assumedly violations of rule number nine were found, were in fact bugs. And the last 2 to complete the dozen even refer to error prevention in their description but still are assigned “help users recognize, diagnose, and recover from errors”. The latter illustrates the influence of the subjects’ understanding of the heuristics.

“Consistency and standards” was by far the most frequently assigned principle to found problems. An interpretation of this fact shows to be anything but easy. As indicated above there are rules which are understood easily and others which require more insight in the matter of usability. First of all, this rule was easily understood by the subjects during the training. Second, the search for inconsistencies can be described as a very formal action, which implies that an evaluator does not need to have deeper knowledge of the domain. Third, this rule is very general compared to others, especially by the aggregation of both “inconsistencies” and “standards” which are similar somehow but could also stand for their own. Maybe Nielsen accredited extra importance to this rule by this aggregation. Fourth, the adherence to this rule could be especially fundamental with interactive information visualization and that is why the evaluators concentrated on this principle. As a matter of fact, only a comparative study of several interactive information visualizations with the method of heuristic evaluation could help to clarify this complex question.

An investigation of the reports quickly showed that often more than one problem was dealt with in one and the same description. The reverse is also true. For one problem the subjects sometimes filed a report in which they gave more than one description of the same problem specifying different violated principles. The latter only seems natural given the general nature of the principles.

5.2. Focus Groups

There were three focus groups – one with 15, one with ten and one with two participants. Originally, we had planned to have two focus groups with a fairly even number of participants but the two students in the last focus group were prevented from attending the second focus group. Every focus group lasted about 90 minutes. A moderator was present to structure the discussion. We prepared a guide of questions for the focus groups. Some of the questions had to be answered by all participants to ensure an even participation but the discussion in the groups was rather lively anyway. One of our team members took notes of everything that was said during the focus groups. These notes were transcribed afterwards and served as basis for further analysis. The goal of the focus groups was to clarify usability issues during the discussion and to get some information about the subjective importance of usability problems.

In response to the question of the biggest usability problem, the subjects made 46 statements, 27 of which were different ones. This was one of the questions that every participant had to answer to ensure a comprehensive overview of the attitudes in the group. Though, the question was not that strict as we allowed for mentioning more than one “biggest” problem per person. That explains the discrepancy of 46 “biggest” problems but only 27 subjects (the 27 different problems are a coincidence). 10 of the 27 different problems were mentioned more than once (see table 2).

Biggest Usability Problem	FG1	FG2	FG3	Total Mentions
Undo/Redo is missing.	3	4		7
Attraction Field: which circle and person do correspond.	3			3
Performance problem.		2	1	3
Time control feedback is confusing.	3			3
Traces: many bugs (size, disappear, remain, numbers remain)	1	2		3
Everything should be controllable via menu.	2			2
Help is missing.	2			2
Reset Window Position is missing.	2			2
Bug: load / save.	1	1		2
No project-files but saved states.		2		2
				29

Table 2. The 10 biggest usability problems stated in focus groups (29 of 46 Mentions). Total Mentions >1.

One of the most evident characteristics of the gathered material is that, except for a few statements (most prominently "undo/redo is missing"), the problems mentioned by the different focus groups were quite divergent. That confirms the claim for the necessity of conducting more than one focus group. On the one hand if like in this case one statement is mentioned not only in different groups but also to such an extent like "undo/redo is missing" (3-4-0) it gives an indication of the importance of the problem. On the other hand 23 of overall 27 different problems were mentioned only in one of the three groups.
A problem's importance may be assessed by the following factors:

- total number of mentions within all groups
- number of groups in which the problem is stated
- distribution of the total number of mentions across groups

What can be said at this point is that the focus groups did not reveal any new problems compared to the heuristic evaluation. But they give an insight in the overall assessment of the software by the subjects. What seems to be the most important strength of this method when used together with heuristic evaluation is the rather efficient way to spot dramatic problems. Much less time consuming as a formal severity rating it gives a first approximation of the most important problems one has to take care of in the next iteration within the software development cycle.
The comparison of gathered material in heuristic evaluation with the one gathered in focus groups shows to be difficult. That starts with the fact that for example the missing of undo/redo can easily be assigned to multiple principles ("user control and freedom", "consistency and standards", "help […] recover from errors", etc.). Each of these assignments can be argued solidly. So for a more detailed comparison one has to classify both reports and mentioned problems in focus groups accordingly.
Generally speaking a comparison may not be as useful anyway because the two methods do represent two different perspectives on the same area and therefore do complement each other in a very meaningful way.

6. Conclusion

The careful evaluation of interactive information visualization has to take place at different stages (e.g. usability, visualization method, case study). Heuristic evaluation shows to be a proper method for assessing and improving the usability because of the general framework it provides. Especially with the addition of screenshots, which helps with comprehending, reproducing, and interpreting, it serves well for the improvement of visualization software in an iterative development cycle. The downside of the mentioned generality is the need to interpret the gathered material carefully to compile directions and guidances for the developers especially if the extent of gathered material becomes substantial.

Focus groups give an insight into the overall view of evaluators on software. It can therefore efficiently provide useful knowledge on how dramatically various problems are experienced. This helps in assessing the importance of respective improvements.

Any of the three mentioned methods gives a different perspective on usability issues, which complement each other to a broader view. We think that the combined evaluation methods therefore ensure a thorough usability evaluation although other methods complementing one another might be used.

The visual elements of Gravi++ use several advantages from a cognitive perspective [1]. Especially, the interactive manipulation can help the user get new insights through the data. The combination of different visualization techniques, like Star Glyph, traces, overview visualization, and the Gravi++ core itself increases the possibilities to find interdependencies. Nevertheless, some problems and shortcomings of Gravi++ are still not solved. Incomplete data leads to incomparable person icon positions because there is no attraction from questions that were not answered. Furthermore, too many questions lead to clutter and make the interaction rather difficult.

7. Outlook and Future Work

We gave a preview (see figure 2) of the work currently in progress: the classification of the 513 reports gathered by heuristic evaluation in order to compare the results of focus groups and heuristic evaluation in detail. For this classification we choose different approaches namely a three-stage location, a unique identification of reported problems, and the labeling of important reports as seen by the investigator. The usefulness of the mentioned interpretations of the gathered material for the further improvement by designers and developers will be assessed.

In the long run after completing the usability study the next steps will be a large-scale study on the visualization method as such both in a quantitative manner with experiments in labs as well as in a qualitative manner with a case study of the real user with their real tasks to ensure ecological validity. We will also test whether Gravi++ can be used in other knowledge domains (e.g. social sciences, history, …) with experts from these domains. This will assure the external validity of the overall study and round off a sustainable evaluation of Gravi++.

8. Acknowledgements

The project “Interactive Information Visualization: Exploring and Supporting Human Reasoning Processes” is financed by the Vienna Science and Technology Fund (Wiener Wissenschafts-, Forschungs- und Technologiefonds WWTF) [Grant WWTF CI038].

9. References

1. Card, S., Mackinlay, J. D., and Shneiderman, B., *Readings in Information Visualization*. Morgan Kaufman (1999), Chapter 1.
2. Chen, C. Empirical Evaluation of Information Visualizations: an Introduction. In *International Journal of Human-Computer Studies*, 53 (5), (2000), 631-635.
3. Craft, B. and Cairns, P. Beyond Guidelines: What Can We Learn from the Visual Information Seeking Mantra? In *Proc. of the Information Visualization '05*, (2005), 110-118.
4. Hinum, K., Miksch, S., Aigner, W., Ohmann S., Popow, C., Pohl, M., Rester, M. Gravi++: Interactive Information Visualization to Explore Highly Structured Temporal Data. In: *Journal of Universal Computer Science (J.UCS) Special Issue on Visual Data Mining*, (2005), forthcoming.
5. Holzinger, A. Usability Engineering Methods for Software Developers. In *Comm. ACM*, 48(1), (2005), 71-74.
6. Kadaba, N.R., Irani, P.P., and Toulouse, M., Visualizing Time Dependent Semantics: An Application to Quantum Algorithms. In *Proc. of the Information Visualization '05 conference*, (2005), 182-187.
7. Kuniavsky, M. *Observing the User Experience. A Practitioner's Guide to User Research*. San Francisco, San Diego, New York: Morgan Kaufmann Publishers, (2003).
8. Lanzenberger, M., Miksch, S., and Pohl, M. The Stardinates - Visualizing Highly Structured Data. In *Proc. InfoVis '03*, IEEE Computer Society, (2003), 47-52.
9. Lanzenberger, M., Miksch, S., Ohmann, S., and Popow, C. Applying Information Visualization Techniques to Capture and Explore the Course of Cognitive Behavioral Therapy. In *Proc. ACM symposium on Applied computing (SAC '03)*, ACM Press, (2003), 268-274.
10. Matthews, G. and Roze, M. Worm Plots. In *Computer Graphics and Applications*, 17(6), IEEE, (1997), 17-20.
11. Morse, E., Lewis, M., Korfhage, R., and Olsen, K. Evaluation of text, numeric and graphical presentations for information retrieval interfaces: User preference and task performance measures. In *Proc. of the IEEE International Conference on Systems, Man, and Cybernetics*, (1998), 1026-1031.
12. Nielsen, J.(ed.) and Mack, R.L.(ed.). *Usability Inspection Methods*. New York: John Wiley & Sons, Inc. (1994).
13. Noirhomme-Fraiture, M. Visualization of Large Data Sets: The Zoom Star Solution. In *International Electronic Journal of Symbolic Data Analysis*, (2002).
14. North, Ch. and Shneiderman, B. Snap-together visualizations: can users construct and operate coordinated visualizations? In *International Journal on Human-Computer Studies.* Vol. 53, (2000), 715-739.
15. Olsen, K. A., Korfhage, R. R., Sochats, K. M., Spring, M. B., and Williams, J. G. Visualization of a Document Collection: The VIBE System. In *Information Processing & Management*, 29(1), (1993), 69-81.
16. Plaisant, C. The Challenge of Information Visualization Evaluation. In *Proc. AVI'04*, ACM Press (2004), 109-116.
17. Rao, R. and Card, S. K. The Table Lens: Merging Graphical and Symbolic Representations in an Interactive Focus+Context Visualization for Tabular Information. In *Proc. CHI '94*, ACM Press, (1994), 318-322.
18. Shneiderman, B. The Eyes Have It: A Task by Data Type Taxonomy for Information Visualizations. In *Proc. IEEE Symposium on Visual Languages*, IEEE Computer Society Press (1996), 336-343.
19. Tognazzini, B. Tog on Interface. Addison-Wesley Publishing, Reading, MA, (1992), 301-315.
20. Tominski, C., Abello, J., and Schumann, H. Interactive Poster: Axes-Based Visualizations for Time Series Data. In *Poster Compendium of InfoVis'03*, IEEE, (2003), 68-69.
21. Tory, M. and Möller, T. Human Factors in Visualization Research. In *IEEE Transactions on Visualization and Computer Graphics*, 10 (1), (2004), 72-84.
22. Ward, M. O. XmdvTool: Integrating Multiple Methods for Visualizing Multivariate Data. In *Proc. Vis '94*, IEEE Computer Society Press, (1994), 326-333.
23. Ware, C. *Information Visualization – Perception for Design*. Morgan Kaufmann Publishers, San Francisco, San Diego, New York, USA, (2000).