Visual Formalism, Liturature Review

Stevenson Gossage
School of Computer Science, Carleton University
1125 Colonel By Drive, Ottawa, ON K1S 5B6
Ottawa, Canada
sgossage@connect.carleton.ca

General Terms

Visual Formalism, Distributed Collaboration

Keywords

Visual Formalism, Distributed Collaboration

1. INTRODUCTION

1.1 Motivation

2. SUMMARY OF THE REVIEWED PAPERS

2.1 Summary: Information Visualization[2]

This paper explains that a visualization is a technique used to display information in a graphical or visual manner with the intent of leveraging our natural visual capabilities. Visualization allows us to process, analyse and understands massive amounts of data very quickly. The problem with visualizations is that we are trying to use them, to quickly understand more and more abstract data, that does not always have an immediately obvious mapping to the real world. This presents a challenge in trying to develop reusable meaningful visualisations. There is ongoing research to try and leverage metaphors and to try understand what type of analytic tasks could be enhance by visualizations based on those metaphors.

The paper talks about visualization developments in several areas including, document and software visualization, distortion techniques, hierarchies, data integration (including data warehousing and data mining) and the world wide web. At the time of publication of this paper in the late 90's there was also some real world commercial applications including applications based on work done at Xerox PARC including Perspective Wall, Cone Tree and Wide Widgets.

The author suggests that, at the time, there was still much work to be done in terms of understanding the potential and usefulness of the technology but it was clear that visualizations would play a huge role in the future of computing. Gershon describes visualization as an emerging discipline which must progress through four fundamental stages and suggests that visualization is progressing at an accelerated pace because many of the step are happening in parallel; "An emerging discipline progresses through four stages. First, it starts as a craft, practiced by artisans using heuristics. Second, researchers formulate scientific principles and theories to gain insights about the processes. Third, engineers eventually refine these principles and insights into production rules. Fourth, the technology becomes widely available."

The paper suggests that widespread usefulness of visualizations will be directly influenced by the content providers' (software design and developers) ability to create visualizations that are accessible and easily understood by a wide range of end users, making them equally useful for users of diverse backgrounds. In order to create these meaningful visualizations in real world applications the systems being modeled and the processes involved must be understood. This understanding in itself could lead to the correct visualization for any particular task.

There are many areas that we need to learn more about in order to better support human learning through visualizations such as what types of interactions make sense for any particular visualization, how we perceive and understand information both visually and non-visually as well as how we search for information and how we adapt those searches based on previous knowledge. A better understanding of these issues and how they relate to specific analytic tasks will inform the creation of more useful, meaningful visualisations that that can truly help end users process more data more accurately and quickly. The ability to create more flexible task oriented user interfaces can also aid in the development of more useful visualizations.

An important point that Gershon makes in the paper is that we also need to understand when graphic visualization may, in fact, be less appropriate. Sometimes a visualization may be harder to understand than words; we need to weigh the options and choose the representation that is most appropriate to allow the users to accomplish the required task.

In conclusion the paper re-iterates that there is great need for the evolution of this emerging technology especially in "The development of scientific and engineering principles for the generation of visualizations (to users with diverse needs and capabilities) and a methodology for solving problems with information visualization are badly needed."

2.1.1 Key Contributions

2.2 Summary: Situational Awareness Support to Enhance Teamwork in Collaborative Environments [4]

This paper was motivated by that fact that today, many modern teams are collaborating using a multitude of visualizations on multiple large format displays. During these activities the situational awareness of the teams is not always ideal. This paper attempts to develop some guidelines to help design better co-located collaborative systems for use with multiple large format displays with affordance for situational awareness.

There were several concepts developed to help support situational awareness on large shared displays including an interface the authors called "Memory Board is an interface that automatically stores and visualizes the activity history". It provides participants access to historical annotations made on other views or visualizations and allows them to be aware of which member is controlling a particular visualisation or display. They also developed the *Highlighting on Demand* interface which allows individuals to highlight or fade out any part of any display using a personal interaction device.

The paper argues that simply providing complex visualizations across or using multiple displays may, in fact, increase the cognitive load and have a negative impact on the collaborative activity. Designing situational awareness into these applications is necessary to properly coordinate team decision making and truly support the collaborative process. One reason is that the participants can be easily distracted by sub-tasks that should in fact be transparent to the activity but in reality are not, like who is in control of a shared artifact or which display or visualisation is the one currently of interest. Shared awareness of these types of interactions are essential and can "leads to informal social interactions and development of shared working cultures which are essential aspects of group cohesion".

Kulyk states that situated awareness is concerned with the individuals knowledge and understanding of the events, information and environment as well as the shared understanding of the team as a whole and their understanding of the past and present and even its impact on future events. Kulyk's work focuses on the following three main aspects which directly relate to and extend Endsley's three levels of situated awareness:

- A person's previous knowledge and understanding which includes the source and nature of relevant events and information.
- Detection and comprehension of the relevant perceptual cues and information from the environment including the comprehension of multiple visualizations and their context.
- Interpretation of theses visualizations and the continuous reconfiguring of understanding and knowledge during collaboration. This supports the awareness of changes in the environment as well as the ability to keep track of work in progress.

Kulyks is actually interested in the impact of situational awareness on team collaboration and specifically what she refers to as *shared situational awareness.* She defines it as the level of the group or team's awareness of individual situational awareness based the above three points. With respect to shared situational awareness, the paper is mainly concerned with answering the following questions:

- To understand the impact of situational awareness on collaboration.
- Understand how to support shared situational awareness in collaborative environments.
- Understand how to leverage the usage of shared large format monitors to support shared situational awareness.
- How can new interactive systems and visualisations be designed and evaluated based on their support for situational awareness and how might these systems actually stimulate new and existing forms of collaboration.

The paper points outs that the level of support for situational awareness varies depending on the collaborative activity. For example a collaborative team that supports emergency dispatch would need a higher level of support when compared to a team working on scientific collaboration. In the long run, whatever the reason for the collaboration, mistakes cost and better situational awareness support can help reduce mistakes. Next the paper talks specifically about situational awareness and scientific collaboration and in particular how large shared displays could be used to better support situational awareness in current omics experimentation in molecular biology. The following key points were made:

- Visualizations on a shared display encourages group discussions.
- The visualization of data on a shared display allows users to quickly determine the quality of the information or data-set.
- Multiple visualizations may be needed and when used there must be a clear visual relationship between these visualizations so as to avoid participants from getting lost, from being distracted and from change blindness which occurs when the participants to do not realize that the focus of the discussion has changed from one visualization to another.
- When multiple shared displays are being used with multiple co-related visualizations then changes in one visualization should also trigger changes in the other related visualizations.
- Difficulties arise when team members from diverse disciplines need to use the same visualizations that may have been designed to be understood by people with a specific background or skill-set as is the case in many scientific visualizations.

To address some of the problems related to situational awareness for shared displays the authors implemented Highlighting on demand interface which allowed the person currently controlling the shared display to highlight or fade out certain visualizations or areas of the shared display(s). This would increase awareness among the team as to the relevant visualization at that moment. A second solution was the memory board interface which stores and visualizes the changes in real-time during team collaboration. This ensures that the entire collaborative process is recorded and all intermediate visualizations, annotations etc. can be retrieved and reviewed. A control interface is suggested as a means of controlling the shared displays and visualizations and giving access information about the visualizations and displays. This was envisioned to run on a shared touch screen as well on the team members individual interaction devices. Knowing who is making changes and what changes are being made is critical to any collaborative effort and the control interface attempt to address this problem.

2.2.1 Key Contributions

2.3 Sumary: Beyond Models and Metaphors: Visual Formalisms in User Interface Design[5]

2.3.1 Sumary

In this paper they argues that the user interface can be designed to support the visual representation of the underlying program's semantics and can help offload cognitive load to facilitate problem solving. Nardi postulates that at the time this paper was published in the early 90's it was time to start focusing more attention on the semantic interface rather then the also important but less interesting supply of user commands to the underlying application. She claims that design and analysis tasks are complex and software designed to support these activities needs to be enhanced with well designed visual interfaces from which the semantics of the application could be easily inferred by the users. The designers of this type of complex application would be well served to have a toolkit of computational building blocks, such that they could piece together the appropriate blocks which could support this visual transfer of application semantics.

Nardi argues that neither Mental models nor Metaphors are appropriate theoretical foundations to support such building blocks but that these blocks could be developed by leveraging the power of visual formalisms. She argues that different visual formalisms can be used as the building blocks for semantic interfaces and showed how a table visual formalism was implemented and explains the potential for it to be used as a building block. Nardi defines visual formalisms as objects that have their own inherent semantics and don't need to borrow the semantics from some other domains for users to understand them, "They are based on simple visual notations such as tables, graph, plots, panels and maps" []. Visual formalism can be specialized or extended to fit a particular problem or application. An example is the table formalism which has been tailor fit into applications such as spreadsheets, calendars, schedules etc. Nardi recognizes the potential for computer based visual formalism to to support interaction, modification, relationships to other components, notation mapping including handwriting to text conversions etc. Nardi attempts to compare how mental models, metaphors and visual formalisms support the design and development of rich applications that convey the semantics of the application through the user interface and how they support application development "through reusable semantic objects that support application development at a higher level than general programming languages." []. In particular the paper is interested in the design of semantic user interfaces for software that supports complex, highly collaborative design and analysis, with explicit support creative problem solving withing the application. Nardi is quick to point out the usefulness of mental models and metaphors are being questioned in general, only that their usefulness are limited for the specific task of creating collaborative software that supports a semantic user interface and complex problem solving. Nardi argues that visual formalism provide the correct level of abstraction to create a library of reusable extensible objects that can truly help the application developers of such systems. The paper argues that many studies of mental models have shown results that that do not support their ease of use nor our ability to used them to help with complex cognitive tasks. In fact there is little agreement among psychologist as to their exact definition or even of their existence. The paper uses an example of one experiment where participants found it difficult to perform seemingly simple cognitive tasks based on a mental images, a concept even simpler then mental models. To illustrate participants were asked to imagine a tiger, and, were then asked to count the stripes of the tiger, task that most participants found difficult or impossible [**?**]. Other experiment by Reisberg also supported the apparent difficulty of accomplishing other simple tasked based on a clear and simple mental images. Reisberg concludes that mental images are less complete and more rigid then we imagine. Nardi argues and provides support that if these simple tasks are difficult and the mental images are rigid and incomplete then how will mental models provide adequate support for more complex , cognitively intense systems. Nardi cites work by Kintsch [] who suggests a more flexible system, where the structure is generated in the context of the task and rules and propositions are composed based on context. Nardi argues that this more flexible strategy is what is needed for the complex systems under discussion. The work of Rips [] is also referenced as evidence for the indiscriminate use of mental models. In many cases propositional or production rules derived by domain knowledge is simply re-cast as mental models, when they would be better off to be thought of as production rules. The paper argues that to match a mental model to a semantic interface would mean to devise a mental model that was "capable of running mental simulations (exactly as Hollan and his colleagues proposed)" []. Nardi doubts if this type of complex mental even exists. The paper suggests that the task placed on developers to understand the mental models and implement them in a semantic interface is overwhelming since, "This approach gives no tools to designers, but asks them to go out and do what professional cognitive psychologists have been unable to do - convincingly describe meaningful mental models of any but the simplest phenomena." []. Nardi also argues against the views of Hutchins [] that suggests that the user interface and the tasks' must be matched with the user's mental model of those task and that the interaction with the user interface should directly correspond to the users mental model and in this way reducing cognitive load because thoughts are easily translated into thew action needed to be performed on the interface to accomplish the goal. Nardi explains that this type of view limits the interface to a simple means to an ends instead of seeing it as having the potential to support cognitive offloading by providing the necessary visualizations for quick and accurate transmission of semantic information about the system and tasks at hand. It can help guide the user and support them in process of complex problem solving; “it is not a passive receptacle for thoughts emanating from an internal model, but plays an active role in the problem solving process” []. An ethnographic study was preformed to try and understand how people use spreadsheets as computational devices to solve real world problems. The results of the study gave support for the idea that people do not tend to start from mental models, “but instead incrementally develop external, physical models of their problems” []. It was found that users tend to start off with unclear goals and through the use of the spreadsheet they are able to organize their data and start to see relationships which lead to the refinement of their problem set and ultimately to the solution; “the representation of the problem emerges through the medium of the spreadsheet. The representation is not “in the user’s head,” nor was it ever in the head, but becomes an artifact created by the user’s interactions with the program.” []. Nardi states that if the users are not using mental models as a means to solve their problems then how can developers design interfaces based on the mental models of the users? Tools like the spreadsheet leverage a table formalism to support the user’s own development of an external model of their problem set. Users are able to created these personalized model of their explicit problem within the constraints of the tool and guide affordance of said tool. Nardi suggests that focusing on mental models distracts us from understanding user activities in terms of (1) high level task goals and (2) the data structures and what operations and need to be supported to allow users to effectively model their unique problems. This evolutionary modeling results in an externalized models that can be used to leverage what is referred to by Reisberg [] as perceptual knowledge and described it as knowledge that can only be accessed through interaction of external representations. This means that there are some forms of knowledge can only be accessed through visual cues. Nardi believes that is more important to understand how external representations or models aide us in performing complex cognitive tasks and how we can build semantic interfaces to support the the users’ ability to create and interpret these external visualizations using visual formalisms as the building blocks.

Next the paper attempts to provide support for the argument that metaphors are also “unsuitable for expressing rich application semantics, and inappropriate for the reusable computational structures we seek” []. Nardi states that the simple, once useful metaphors of the desktop, Rolodex, or trash can icons are no longer need to help end user understand how to interact with their computers and applications. The user interface has also moved beyond simply relaying user commands to the underlying system, it now crucial to understand how the interface can convey rich application semantics and provide support for complex cognitive operations. Metaphors lack precision and are incomplete representations of any bit the most simple applications. “Metaphors are good at suggesting a general orientation, but not good at accurately encoding precise semantics” [], but it is precisely those semantics that the interfaces under discussion is trying capture, convey and support. Another problem is that metaphors are not reusable and not extensible and become stale, and therefore could not be used to support developers in creating these types of applications. Carroll and Thomas [] suggest eight rules to help develop or recognize good metaphors, the first one states that metaphors are system dependant and must also account for the expected user of the system. This alone ties metaphors to the systems they represent and gives evidence for their lack of re-usability and extension. Also, metaphors, become stale, “However, for most computer systems there will come a point at which the metaphor or metaphors that initially helped the user understand the system will begin to hinder further learning... When the original metaphor begins to fail, either new, more detailed metaphors will have to be introduced or the metaphorical approach will have to be abandoned in favor of a more literal approach.”[]. Nardi also talks about the impact of prior knowledge and how it can sometimes temp us into seeing and using metaphors which simply are no there. As an example Carroll [] suggests that a ledger sheet is the metaphor representing spreadsheets, Nardi argues that in fact spreadsheets are simply tables and that there are no metaphors that can describe the generality and expressive power of the table. At best metaphors could be used to express specific uses of tables, but nothing that could come close to capturing the entire possible semantics possible through the use of tables. Nardi argues that it is far to easy to “confuse any kind of prior knowledge with metaphor ” []. Next the paper turn its attention of explain why visual formalisms are the appropriate method to help designer and developer convey application semantics through the user interface. Harel clearly expresses the need to support complex cognitive activities and problem solving through semantic interfaces when he said, “They will be designed to encourage visual modes of thinking when tackling systems of ever-increasing complexity” []. We use physical or real visual formalism all the time in everyday life, including maps, tables sketches and graphs; digitized visual formalisms bring new a dimension previously unavailable. Notably, computerized versions can be stored, search, retrieved, versioned, they also afford virtually instant sharing (or even real-time sharing) and can support modification by collaborators near and far (with whom the visualizations were shared). These visual formalism must provide support for the user to be able to draw meaning not just from viewing them in a static manner, rather, by the ability to manipulate, compose, add or subtract them, search, filter and combine them in ways that make sense for their particular application. Implementing these allows our designers and developers the opportunity to use them as building blocks to create complex semantic interfaces that support hard cognitive process through visual guidance and visual support of semantic reasoning. Nardi suggests that visual formalisms are application frameworks because they provide the structure in which to support a specialized problem. As an example, the spreadsheet is a specialization of a table formalism, the table formalism provides the underlying base functionality of the system, but the specialized implementation (the spreadsheet in this case) provides the structure and constraints that allow the end user to focus on their dataset and allow the problem to emerge in the form of an external model, created and guided by, and within the boundaries of those constraints. In this way “The initial phase of a modeling problem is reduced to simply recognizing a format into which a problem is framed, rather than being faced with the necessity of inventing a format from scratch” []. Nardi describes the strengths and properties of visual formalism in the following list:

- *Exploitation of human visual skills.* Visual formalisms are based on human visual abilities, such as detecting linear patterns or enclosure, that people perform almost effortlessly. Visual formalisms take advantage of our ability to perceive spatial relationships and to infer structure and meaning from those relationships (Cleveland, 1990). Visual formalisms are capable of showing a large quantity of data in a small space, and of providing unambiguous semantic information about the relations between the data.

- *Strong semantics.* Of the visual formalisms we have identified, graphs have defined for them the most formal semantics, and panels the least. Any implementation of a visual notation as a visual formalism will supply its own formal semantics. A computer-based version of a panel may only specify that a panel is an arbitrary collection of objects that belong together. Its strength as a presentation format makes it an extremely useful object, and an implementation might have a great deal of presentation functionality in addition to the semantics of being able to handle a collection of objects correctly for a given application.

- *Manipulability.* Visual formalisms are not static displays, but allow users to access and manipulate the displays and their contents in ways appropriate to the application in which they are used.

- *Specializability.* Visual formalisms provide basic objects that can be specialized to meet the needs of specific applications. They are at the right level of granularity neither too specific nor too general. Visual formalisms are appropriately positioned between the expressivity of general programming languages, and the particular semantics of applications.

- *Broad applicability.* Visual formalisms are useful because they express a fairly generic set of semantic relations, relevant to a wide range of application domains. Because a large number of applications can be designed around a given formalism, visual formalisms will eliminate a great deal of tedious low-level programming, as well as give developers ideas about editing and browsing techniques with which they may not familiar, such as the use of fish-eye views for large datasets [].

- *Familiarity.* Because the standard visual notations are so useful, they are found everywhere. Not only do they draw on innate perceptual abilities, but through constant exposure we become very familiar with them. Our schooling explicitly trains us in the use of the basic notations; e.g, using calendars and learning matrix algebra provide experience with tables. Everyday activities provide opportunities to create and use visual notations, such as writing a laundry list or reading a map.

Finally to wrap things up, the paper discusses their implementation of a table visual formalism as an illustration of how a visual formalism (or more generally a set of them) can be used to help developers build rich semantic interfaces for systems designed to support complex cognitive tasks including problem solving. To start with, the implementation of any visual formalism must start off with by defining the semantics of the visual formalism. This involves identifying how individual components are structured, how they relate to each other and what operations need to be supported. The presentation and layout as a whole and as individual components must be considered and formalized. The application semantics within the constraints of the visual formalism must be clearly understood. This is really defining how the framework can express the underlying semantics of the system. Their implementation creates a more flexible table then you might find in a typical spreadsheet application but in keeping with the proclaimed advantages, it could be used to create a spreadsheet. The table formalism is a component that can contain other components (cells). Each cell is disjoint but can be set to span rows or columns as well as be merged with other cells or split into multiple cells. Cells can also be added (created), deleted, moved, copied and pasted. There is support for labeling rows and column as well as naming regions (a group of cells). The ability to iterate over cells or regions, to scroll, boith horizontally and vertically as well as support of zooming in and out cells, rows, columns or regions. There is support for filtering (essentially temporarily hides certain cells, regions, rows or columns). Another big difference of the table formalism is that the cells can contain anything from a primitive to a complex class or widget or even an application. To finish the paper provided strong support for the use of visual formalisms to aide developers in the design of complex systems. It clearly illustrates where mental models and metaphors fall short as a theoretical foundation for the creation of these types of systems. Finally it gave a good concrete description of a table visual formalism.

2.3.2 Key Contributions

2.4 Summary: Heuristics for Information Visualization Evaluation

This paper defines heuristic evaluation as a quick and lightweight method to predict and spot issues related to the usability of an applications' user interface. The heuristic approach involves a small group testing the system based on the heuristics or guidelines that are relevant to the system. Heuristics can often be used to teach novice users and as a method of identifying design patterns. Like design patterns heuristics help by providing an over arching language from from which the system can be talked about and promote reuse especially through the evolution of design patterns. Zuk describes heuristic evaluation as commonplace in the field of HCI but, its application to information visualization is still emerging. The author states that we are at a point where there is a need to create a small set of general overarching heuristics that could be applied to most information visualization system, similar to Neilson's 10 usability heuristics for the design of user interfaces. The paper describes this as an exciting and open area for future study. Zuk suggests that a hierarchical grouping of existing heuristics combined with different tree searching algorithms could help in selecting a core set of heuristics. Otherwise, one could follow Neilson's method of reducing a large set of problems into a small set of easily understandable heuristics. Next the paper describes the process of heuristic evaluation as something that is apt to evolve but nonetheless describes that an iterative approach using five evaluators has been found to be effective. In the first pass the evaluators try and get a general sense of the application and a second pass is used to focus on applying the heuristics to specific elements of the user interface. The exact number of evaluators will most likely be application specific and dependant on the complexity of the visualizations. The paper refers to work by Tory and Moller /cite who suggest using a combination of traditional usability experts and visualization or data display experts. It is still to be determined what the exact characteristics of information visualization evaluator should be. The paper describes work done by Craft and Cairns [] as determining that there is still much work to be done in this area, "They conclude by calling for a more rigorous design methodology that: takes into account the useful techniques that guidelines and patterns suggest, has measurable validity, is based upon a user-centred development framework, provides step-by-step approach, and is useful for both novices and experts."[] The paper also warns that there could be problems if we blindly apply existing knowledge and methods of usability heuristics to information visualization and suggest that caution should be used and further study is needed. Zuk noted in the case study that the evaluators had problems with the heuristics that were too specific and felt a more general vocabulary would increase its usefulness. It is was also noted that in general more precise, clear and understandable descriptions of each heuristics would facilitate their use and help ensure that they are used as intended and it leaves less room for misinterpretation. The case study revealed that some problems were detected by distinct heuristics which the author suggested could be used as way to see the problem from distinct points of views and possibly could lead to multiple solutions. It was also found that assigned priorities could help resolve issues when conflicting heuristics were found. Another suggestion was to let domain experts resolve heuristic conflicts. The authors also identified some problems that may be common among visualizations in general. Some of these issue include:

- Poor contrast can produce hidden or difficult to see visual components.
- Poor choice of colors could lead to confusion in terms of understanding or misinterpreting the relationships between visualizations.
- Tool tips lack the detail required to make them useful.

The paper notes that some heuristics would be better used as guidelines for the designers and or developers of the system and other might be better for heuristic evaluation of the visualization. Zuk suggests that paired evaluator teams could be comprised of evaluators paired with domain experts, this could help in a more accurate application of the heuristics. It was also noted that usability heuristics could could be useful in combination with the visualization ones but note that this requires further study. Another open research are would be to discover if tools could be used to support the evaluation of visualizations.

2.4.1 Key Contributions

2.5 Summary: A Collaborative Dimensions Framework: Understanding the Mediating Role of Conceptual Visualizations in Collaborative Knowledge Work

The goal as stated in the paper âĂIJis to identify the factors that contribute to the choice of an effective visual representation for collaborative knowledge work to support distributed cognition, and to organize them in a conceptual frameworkâĂİ. Bresciani focuses on visualizations as artefacts and claims that this focus allows for decision factors to be expressed in terms of dimension that either support or interfere with certain kinds of visualization uses. These dimensions are not independent and often, supporting one dimension will have a negative impact on the support for another. Understanding the relationships between dimension allows designers of collaborative visualizations to pick and leverage those dimension that make the most sense for the application being developed. The author suggests that this approach will help build better support for collaborative activities then could be achieved using general guidelines that should apply to all visualizations as might be the case using a heuristic approach. This paper leverages research from three main fields including:

- Blackwell's work on the cognitive dimensions of notations framework [1] and Hunhausen's work on communicative dimensions framwork [3]
- Star's research on the boundary object paradigm [6]
- Thirdly a literature review on visualization technologies, in particular the information visualization work of Shneiderman and Karabeg as well as the knowledge visualization work of Eppler and Suthers.

Informed by this theoretical foundation and a combination of techniques used to gather the experiences and knowledge of expert practitioners who employ visualizations in their daily collaborative activities, a collaborative visualizations framework was developed to help understand how visualizations can be leveraged to support collaborative activities and how they can facilitate the distribution of knowledge across team members. The framework consisted of the following seven dimensions:

- *Visual impact*: The extent to which the visualization is attractive and if it facilitates attention and recall.
- *Clarity*: Is the visualization self-explanatory, easy to understand and dose it reduce cognitive effort.
- *Perceived finishedness (provisionality)*: Is the visualization perceived by the user as a final product.
- *Directed focus*:Does the visualization direct the users attention to any particular area or areas.
- *Inference support*: Do the constraints of the visualization help or support the creation of new and novel insights. âĂIJInference support is the core differentiator and added value of visualization over text: it allows to gain new understanding âĂIJfor freeâĂİ just by changing the visualization type, the focus, or the representational constrains.âĂİ
- *Modifiability*: Is there support for dynamic modification of the visualization?
- *Discourse management*: Does the visualization support control over the discussion and workflow of the activity.

The paper applied the dimensions to real world visualizations to determine to what extent the different dimensions were supported by the visualizations. They identified six different but common forms of collaboration and suggested that some dimensions would be more appropriate for certain collaborative activities, The following lists each of the six collaborative activities along with the dimensions that provide the best support for those activities:

- idea generation (high visual impact, low clarity, low perceived finishedness, high modifiability)
- general knowledge sharing (high visual impact, low perceived finishedness)
- problem analysis (low perceived finishedness, high clarity, high inference support)
- option evaluation (high clarity, high directed focus, high inference support, moderate-low modifiability)
- deliberation or decision elaboration (high clarity, high directed focus, high inference support, moderate-low modifiability)
- planning (high modifiability)

As noted the above dimensions are not always independent and careful consideration must be made when deciding the level of support for each dimension. This concept of trade off between dimensions was first introduced by Green and Petre []. As an example of these trade-offs the paper refers to the fireworks visualization from Let's Focus collaborative software which was used as one of the three visualizations to which they applied the collaborative dimensions. The application of the dimensions to the fireworks visualization revealed several relationships or trade-offs between the dimensions. Visual impact had effect on clarity and on directed focus. The high visual impact of the fireworks visualization was distracting and therefor reduced its clarity and it had reduced support for directed focus because of the appealing nature of the visualization in fact, âĂIJWhen the visual stimuli are very high, then the focus will diminish because the attention is caught more by the aesthetics than the contentâĂİ. There were also trade-offs between modifiability and perceived finishedness and between modifiability and discourse management. The visualization had a high level of support for discourse management through the Let's Focus software package that provided the visualization. It also had high perceived finishedness âĂIJbecause the diagram resembles a final piece of work instead of a discussion tool.âĂİ If a visualization is perceived as finished, then it will be less likely that collaborators will feel like they should contribute or modify it. If there is no tool support for discourse management then high modifiability usually results in in low discourse management because coordination of the group becomes more difficult.

Another important consideration is the medium on which the visualization is delivered. âĂIJThe choice of alternative media such as a whiteboard, paper or computer-based interaction, strongly affects people's willingness to make contributions and collaborate.âĂİ If people feel that modifications of the visualization are permanent then they are less likely to want to contribute, also the perceived finishedness can be affected by the medium, a peice of paper tends to be though of as more provisional where as a visualization on a computer monitor has more of a sense or permanence even though the computer based visualization may in fact have high support for modification.

2.5.1 Key Contributions

2.6 Sumary: Distributed and Collaborative Visualization

2.6.1 Key Contributions

3. DISCUSSION

4. CONCLUSIONS

5. REFERENCES

[1] A. Blackwell and T. Green. *Notational systemsï£¡the cognitive dimensions of notations framework*, page 103ï£¡134. Morgan Kaufmann, 2003.

[2] N. Gershon, S. G. Eick, and S. Card. Information visualization. *interactions*, 5:9–15, March 1998.

[3] C. D. Hundhausen. Using end-user visualization environments to mediate conversations: a 'communicative dimensions' framework. *Journal of Visual Languages and Computing*, 16:153–185, 2005.

[4] O. Kulyk, G. van der Veer, and B. van Dijk. Situational awareness support to enhance teamwork in collaborative environments. In *Proceedings of the 15th European conference on Cognitive ergonomics: the ergonomics of cool interaction*, ECCE '08, pages 5:1–5:5, New York, NY, USA, 2008. ACM.

[5] B. Nardi and C. Zarmer. Beyond models and metaphors: visual formalisms in user interface design. In *System Sciences, 1991. Proceedings of the Twenty-Fourth Annual Hawaii International Conference on*, volume ii, pages 478 –493 vol.2, Jan. 1991.

[6] S. L. Star and J. R. Griesemer. Institutional ecology, 'translations' and boundary objects: Amateurs and professionals in berkeley's museum of vertebrate zoology, 1907-39. *Social Studies of Science*, 19(3):387–420, 1989.