# CoCoNutTrix: A Study in Collaborative Retrofitting for Information Visualization

Petra Isenberg[1] Anastasia Bezerianos[2,3] Nathalie Henry[4,5] Sheelagh Carpendale[1] Jean-Daniel Fekete[4]

[1]University of Calgary, Calgary, Canada
[2]NICTA, Eveleigh, Australia
[3]Ecole Centrale Paris, Paris, France
[4]INRIA, Orsay, France
[5]Microsoft Research, Redmond, USA

## Abstract

Most information visualization systems incorporate little or no specific support for collaboration. Yet, collaboration has been identified as a means to deal with the increasing amount and complexity of information available to analysts. Re-implementing existing applications to fully incorporate our current understanding of computer supported cooperative work could be exhaustive, time consuming, and expensive. As a first step, we explored how co-located collaborative information visualization and analysis environments can be created in a cost-effective manner. We retrofitted NodeTrix, a social network analysis tool, by extending it to enable multi-user interaction in collaborative environments. We present details of the retrofitting process and results of a user study assessing the usability of our retrofitted system. Our results support the effectiveness of our low-cost collaborative retrofitting for collaborative network analysis and highlight implications for practitioners.

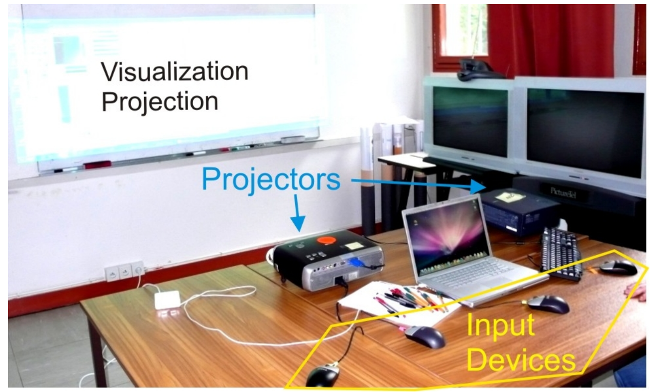


**Figure 1:** *An example of a low-cost setup for co-located collaborative data analysis using four mice, two projectors, and a wall for projection.*

## 1 Introduction

Within human-computer interaction, much of the literature concerned with designing and evaluating co-located collaboration, revolves around dedicated hardware in forms of touch-sensitive displays, input devices, or software. Each of these brings a number of advantages for certain collaboration environments and situations. It might appear that adapting information software for co-located collaboration would require specialized hardware and re-implementing the application to, for example, scale to specific presentation spaces like large high-resolution wall or tabletop displays, to make use of head-mounted displays or CAVES, or to react to other forms of input like direct-touch, gloves, or pens. Combining these approaches—taking large information visualization and analysis systems and re-implementing them to fully incorporate our current understanding of computer supported cooperative work (CSCW)— is exhaustive, time consuming, and expensive.

Our approach is to explore how collaborative information analysis environments can be created in a cost-effective manner both in terms of the required hardware and time. We are motivated by the potential benefits of co-located collaboration around data. Sharing a single information display may enable new types of interaction between analysts and enrich existing collaborations—data interpretations can be discussed and negotiated during the collaboration rather than after, expertise and data analysis skills can be shared, and peer-learning and peer-teaching are encouraged.

To create a low-cost collaborative environment it is possible to use multiple off-the-shelf projectors that can be simply pointed at a blank wall to create a large display, coupled with technical solutions that replace single mouse or keyboard input streams with multiple input devices (e. g., [6]) as can be seen in Figure 1. However, it is not clear to what extent such a simple approach supports collaborative information analysis, what the requirements and challenges are in practice, or whether a low-cost collaborative environment will support the representations and tasks typically involved in information analysis.

As a first step to answering this broader question, we retrofitted a version of NodeTrix [3], a single-user graph visualization environment, to support multiple independent mice. Then, we conducted an observational study to assess how analysts viewed our low-cost environment (e. g., Figure 1), and whether it effectively supported collaborative data analysis among domain experts using real datasets in the context of social network analysis. To ensure that our low-cost collaboration setup was effective under different realistic settings, the observational study was conducted in three research organizations, using technical facilities present in each organizations. With this research we assess one example for transitioning from single user to multiple user information visualization support for co-located collaboration and we present a number of hypotheses for generalizing our findings. Our intention is that from our results and with further research, our knowledge about retrofitting and hence designing co-located collaboration visualization systems will adjust and expand.

## 2 Collaborative Retrofitting of NodeTrix

### 2.1 A Short Introduction to NodeTrix

NodeTrix [3] is a hybrid visualization in that it combines a node-link representation and an adjacency matrix-based representation of a social network in a single view. This makes it possible to view all data entities represented as nodes and all inter-node relationships as links. Alternatively, one can view all data entities as labels in rows and columns in a matrix and their relationships as the matrix cells. Most importantly, the two representations can be used in combination, with part of the data presented in either node-link or matrix form (Figure 2). Whether a particular entity in the data is shown in either of these representations is interactively controllable. For instance, one can group node-link data entities to form a matrix, or select a data entity and drag it into or out of any given matrix. This interactive dual representation combines in a single view the benefits of node-link diagrams and adjacency matrix-based representations, and is conducive to visual data exploration. Figure 2 gives an overview of the visualization in which communities within a computer-science department are grouped together in matrices and connected by links representing co-authorship relations.

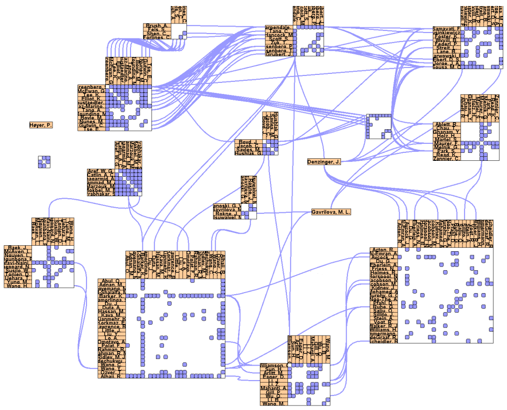

**Figure 2:** *NodeTrix Visualization integrating node-link and matrix visualizations. This image shows the co-authorship network of a university department in which research labs have been grouped into matrices.*

### 2.2 Choice of NodeTrix for Collaborative Work

To explore collaborative retrofitting of existing information visualization, we wanted to begin with a tool that seemed a promising candidate in its existing state. Thus, we first considered published guidelines for information visualization design for co-located collaboration [4]. According to these guidelines we found a promising candidate in NodeTrix [3]. Specifically, it supports:

- *Free categorization of items*: Nodes can be grouped into matrices with a lasso gesture. Single matrices can be dissolved with a single click. Nodes can be added to or removed from matrices with drag-and-drop. Hence, work on a given item can be done independently from work on others. This could support concurrent work.
- *Free workspace organization*: Data items can be freely repositioned. This allows individuals to work on the task in different areas of the display.
- *Individual viewing preferences*: Through a number of local changes in the representation, individuals can adapt parts of the representation to their own preferences.
- *Fluid interaction*: The number of changes of input modality, the manipulation of interface widgets, and dialogs is kept to a minimum and can improve the coordination of activities within a group.
- *Focus on mouse interaction*: Almost all actions are mouse interactive, which makes the tool accessible to retrofitting for multiple inputs. The keyboard is only required for three tasks: to type labels, to trigger undo, and to trigger a graph re-layout.
- *Minimal global changes*: NodeTrix includes only two possibilities for global changes, limiting the possibilities for accidental changes that affect all users which may lead to less interruption of the group work.

In addition, several practical aspects of NodeTrix made it a good candidate for our work. It has previously been used successfully with experts in the context of social network analysis and has been shown to be useful in single-user work [3]. We also had access to the underlying source code and could make necessary adjustments to introduce concurrent inputs. We nick-named our retrofitted co-located collaborative NodeTrix—*CoCoNutrix*.

However, some of the guidelines as outlined in [4] are not specifically supported. There is no specific support for communicating findings or discoveries, solving conflicts of interaction, graphical history, or maintaining individuals' awareness of each other's efforts. Thus, while NodeTrix presents a promising starting point, it is not clear whether NodeTrix will help group members to collaborate effectively. Through an observational study and interviews we explore how participants utilize our retrofitted collaborative software and whether such minimal retrofitting can sufficiently support collaboration for data analysis. We are interested in the following questions:

- Is communication between analysts enabled?
- Do interaction conflicts occur that hinder the collaboration?
- Can group members stay aware of each others' work?
- Are group insights achieved?
- What is the qualitative analysis experience with the system?

### 2.3 Implementation Details

To implement CoCoNutrix we made adjustments to the underlying source code. We kept all our re-implementation choices to a minimum. Wherever possible we opted to leave things as they were, as our goal was to study whether a minimal retrofit would accrue collaboration benefits.

#### 2.3.1 General collaboration support

One of the challenges in re-designing software for collaboration is that global changes should be kept to a minimum to avoid interrupting group work. Yet, many information visualization systems, NodeTrix among them, offer a high number of parameters to change the visualization output. In our retrofitted tool we turned off menu bars and control panels and chose appropriate default values for all visual features such as link width, colour, or label size appropriate for our task and dataset.

Since the main current operating systems do not support multiple windows to be in focus we chose to provide a fullscreen visualization environment, in which no accidental resizing, repositioning, or a change of focus of the application windows could occur. Since all control panels were turned off already this was achieved by giving all available screen space to the rendered visualization. In other applications in which multiple windows, widgets, or dialogs are necessary, these may have to be re-implemented to allow more than one window to be in focus at a given time.

#### 2.3.2 Adding Multiple Inputs

In NodeTrix, since mouse interaction is the most common, we decided to give each collaborator their own. On the other hand, keyboards were only used for three relatively rare interactions (labeling, triggering a global re-layout or undo) and take up a lot of physical space on the table, we decided to provide one shared keyboard.

To capture independent input from any attached mouse, we used the JInput library [6] and added a GlassPane, a transparent panel, on top of the application to render the additional mouse cursors and dispatch modified mouse events to the application. We derived a new mouse event class that carried individual mouse ids in addition to the traditional mouse event data. These ids were necessary to be able to react to user specific input. For instance, user-specific data structures were put in place to keep track of which items were being drawn or dragged by which mouse. For example, the lasso gesture was used to select multiple nodes. To capture this gesture it was necessary to save a mouse path per user.

In keeping with the spirit of making as few changes as possible in our retrofitting and because it has been suggested that social protocols are often the preferred conflict resolution method [11], we chose to leave the resolution of conflicts to these social protocols.

#### 2.3.3 Changing Representation and Interaction

We made three changes to visual representation and interaction: (i) We provided additional visual feedback. To differentiate the available mice, each cursor was enlarged and received an individual color. Click or drag interaction from these mice created a similarly colored glow effect on each clicked node or matrix. We extended the rendering code for both objects and rendered a colored semi-transparent rectangle on top of them to achieve this effect. (ii) We changed keyboard input for matrix labels. Previously, labels were created by selecting a matrix and typing the desired text. In a multi-user case several matrices can be in focus and, thus, it is unclear to which one a label should be added once a user starts typing. To circumvent this problem we created a new label object, representing the label text. This object was added to the visualization after a user finished entering text. It could then be dropped on a matrix to create a label. (iii) Using mice input we provided functionality that was previously available through selection and control panel interaction. To allow zooming in and out of rendered matrices, we mapped the resizing action to the mouse wheel, a simple fix to address the previously mentioned mouse focus problem. All the interactions are implemented using “Interactor” objects decoupling the interaction from the visualization rendering and from the logic of the application. This feature of the toolkit made the retrofitting easier.

#### 2.3.4 Retrofitting Cost

Estimating the retrofitting cost is difficult as it relies on the developers’s knowledge of the underlying code and the number of places to edit. As an indication, we created ten classes and wrote less than a thousand lines of code to retrofit NodeTrix.

## 3 Study

The goal of our study was to determine whether our retrofitted version of NodeTrix could support collaborative social network analysis in realistic settings and examine how users viewed our cost-effective design decisions. We strove to provide a study environment as close as possible to (a) real environments, (b) real data, (c) with domain experts who are (d) performing real social network analysis tasks.

We studied groups of four experts performing social network analysis using data from their own organization. Our participants were experts in the data, not social network analysis experts. To ensure that our collaboration setup was effective in different realistic settings, the study was conducted in three organizations (Org A, an educational institution, Org B and Org C, research organizations) using existing technical facilities.

### 3.1 Social Network Data

Our three organizations have an interest in determining how their internal research groups collaborate and how effective these collaborations are. We, therefore, decided to use research collaboration social networks as data for our study. Given that research publications are a good indication of collaboration, the co-authorship network of each organizations was used as a dataset. Authors in the dataset became nodes of the network, and co-authorship relationships became links. Each institution had a high number of authors (exceeding 800 in all three), making the analysis difficult to complete in less than one hour. To ensure a whole experimental session could be concluded in approx. 1.5 hours, thus making it easier to recruit knowledgeable experts with limitied available time, we filtered out authors with a low number of publications. This resulted in 423 authors for Org A, 327 for Org B, and 430 for Org C.

| Org. | Screen Size | Resolution | Projectors | Distance | Figure |
|---|---|---|---|---|---|
| A | 1.46 m × 1.1 m | 2048 × 1536 | 2 × 2 | 1 m | 3-left |
| B | 4 m × 1.5 m | 2560 × 1024 | 2 × 1 | 1.5 m | 3-right |
| C | 2 m × 0.8 m | 2560 × 960 | 2 × 1 | 2 m | 1 |

**Table 1:** *The physical study setup in the three organizations.*

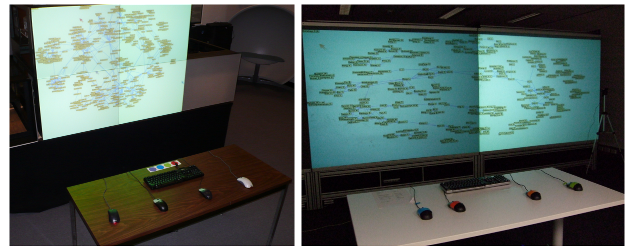

**Figure 3:** *Study setup in Org A (left) and Org B (right) using display and computer resources available at each organization.*

### 3.2 Participants

44 participants (14 female) took part in our study. All had been with their organization for at least 6 months and were experts in either parts or the entire social network they were asked to analyze. Their positions included senior professors/researchers, group and project leaders, administration, human resources personnel, technical personnel, and few graduate students. We recruited 4 groups (16 participants) in Org A and Org C, 3 groups (12 participants) in Org B. To ensure a realistic and comfortable collaborative setting, participants were either work collaborators or friends. With one exception, all participants reported to be familiar with their group.

### 3.3 Apparatus

Resources in the organizations differed slightly, but an effort was made to keep the settings as similar as possible. The same visualization software ran on a dual core 3GHz CPU, with 2G RAM, running Windows Vista. In each setting, the 4 physical mice were color-coded to match their respective cursors on the screen. The details of our physical setup can be found in Table 1.

### 3.4 Task

Participants were presented with a visual representation of a social network that they had intimate knowledge of in terms of: actors (researchers), their roles and positions in the organizations, and their working relations. Participants were asked to create a representative view of the researchers in the organization that could later be printed in poster form. They were provided with a single shared network representation using a force-directed layout (LinLog [8]). For this task they were asked to identify and name the different communities, defining their own criteria. This type of open-ended task of identifying communities and examining their connections is commonly performed in social network analysis [12].

### 3.5 Procedure

Each study session lasted 1.5h. Participants were asked to complete a brief demographic questionnaire eliciting their background, their familiarity with the rest of the group, the dataset, and their experience in using social network software. They were then introduced to the NodeTrix collaborative system and were allowed to experiment with it for 15–20min on a training dataset. After reporting feeling comfortable using the system, they proceeded into the main task of organizing and labeling the co-authorship social network of their organization. The task ended when they completed their labeling and grouping of the network, or when they reached the 40min mark. After a short break, the entire group took part in a semi-structured group interview eliciting their opinions on the task and the system. An experimenter was present for the duration of the study to answer any questions.

### 3.6 Data Collection and Analysis

Apart from the pre-trial questionnaire, observations, and interview, a number of other data was collected for later analysis. All sessions were video-captured from two distinct locations focusing on both the participants and the screen. Moreover, detailed system logs were stored for each session. Finally, a note taker was present making detailed observations on the use of the system and interaction of the participants. For the analysis we combined our information from transcribed interview data, notes, and observations and created affinity diagrams to reveal patterns in the data.

## 4 Results

In this section we present how our retrofitted collaboration environment provided collaboration support and assess whether this support was effective. We group our results according to the mechanics of collaboration [2], reporting on these low-level actions and interactions that a collaborative system must support in order for group members to be able to complete a task in a shared manner, as well as findings relating to our understanding of effective collaborative data analysis. Similar to [2], we consider the collaboration to have been effective when activities could be completed successfully, and no major errors or conflicts arose.

### 4.1 Explicit Communication

In face-to-face settings like ours, the majority of explicit communication is verbal and is the main means to establish a common understanding of the task at hand.

*Observations:*
We observed frequent verbal communication: in 9/11 groups lively communication arose around the content of the data. We observed two types of explicit communication: running commentary and direct discussions. Running commentary was common when participants wanted to quickly inform others of an action performed or planned without an intent to start a conversation. Direct discussions were used to directly contribute to social knowledge building: groups exchanged rational and argumentation regarding actor placement or grouping choices, group members would agree, disagree, and negotiate, building a shared understanding of the network they analyzed.

Since participants were not directly interacting with the display, our system needed to facilitate deictic references and gesturing for communication in and with a group. Participants performed deictic references not only by pointing with their hands at the display and making verbal references, but also by gesturing and pointing *indirectly* with their uniquely colored mouse cursors. Moreover, they repurposed the system to their communication needs, for example by enlarging an object to attract attention. During phases of joint visual attention, mice were commonly moved to the joint focus area to show that attention was given to a specific information item that was under discussion.

*Requests for Improvement:*
Participants only requested additional features to support deictic references. Three groups asked for a visual feature, such as a user-controlled glow or animation, that could explicitly draw the visual attention of the group to a particular mouse cursor.

*Summary:*
We observed that our system provided adequate support for intentional verbal communication, facilitated mostly through the face-to-face setting. Participants made creative use of the visual representation to perform deictic referencing, with few participants asking for better support. One of the goals of collaborative information visualization tools is to allow groups to come to a common understanding of the the data through the use of the visualization. Through our observations of instances of explicit communication we are quite confident that this goal was reached.

### 4.2 Consequential communication, monitoring and group awareness

Information in physical collaborative settings is unintentionally given off by collaborators and by artifacts as they are being manipulated, for example seeing hands move in the space or hearing paper being dragged by others. This consequential form of communication is very important in digital collaborative tasks as well, as it is the primary mechanism for gathering awareness information about what is going on, who is working on what, and where others are in the workspace.

*Observations:*
We observed four main visual features with which the representation mediated consequential communication and enhanced awareness within the group:

**Color Coding:** Our environment provided a single explicit awareness mechanism in the form of uniquely colored cursors and matching coloring of selected artifacts. This color coding indirectly indicated to participants areas of the display and specific artifacts that others were focusing on.

**Labeling:** Participants labeled communities to indicate that they had been analyzed or needed further work, implicitly informing the group of the work to be done. For example in 9/11 groups participants would only give a community a name once they felt it was reasonable finalized, while in 2/11 groups unknown or not finalized communities would be given a predefined default name (e. g. ”unknown 1”).

**Location:** Participants implicitly communicated their decisions regarding communities by placing them at predefined areas of the display. Some groups (2/11) used the periphery of the display to place finalized communities, while others used a predefined area of the screen for ”unknown” or ”draft” communities (2/11). Although in most cases this placement started out unintentionally, it often became an explicit work practice (e. g. “I am putting unknowns to the right”).

**Scale:** In 6/11 groups, matrices representing finished groups were scaled down in size to communicate that they should not be edited further.

Participants generally reported to have been aware of group process on the visualization. Yet, we observed several participants stop their interaction for moments at a time and watch engaged at the representation. When asked about this behavior in the interview, they reported to have done so to gain an overview of what had changed in the dataset, what the group strategy was, and what areas they could work on next.

*Requests for Improvement:*
One known issue that pertains to awareness is that users easily lose their mouse cursors on large displays [1]. Participants in 6 groups reported to have lost their cursors occasionally, even though we had increased the mouse cursor size to four times that of the standard Windows desktop and given each cursor a distinct bright color.

During the interview some groups (5/11) also asked for more explicit ways of labeling and annotating their work to assure that decisions would not get lost in the work process (e. g. changing colors of communities to indicate they are completed, giving matrices specific descriptions like "do not merge!", etc).

Only participants in 4/11 groups requested a feature for viewing the interaction history of the group, to see each other's actions and the history of a specific area of the network.

*Summary:*
Although our participants were able to collaborate on the retrofitted setup, half of them felt the colored cursors did not provide enough awareness of other users' actions. Annotation functionality was also requested to mark the state of communities. However, most felt that although detailed actions were missed, they were globally aware of the group process and progress. Interaction history was not frequently requested maybe due to the task and length of our study. We generally saw the visualization itself being used as the medium to indirectly capture, represent, and communicate the group understanding and knowledge of the communities in the dataset.

### 4.3 Action coordination, assistance, and protection

An important part of effective and fluid collaboration, is how collaborators mediate their actions and share common workspace resources. People organize their actions to avoid conflict with others and efficiently complete their task.

*Observations:*
Our participants clearly organized their actions in order not to conflict with others. This was achieved by either explicitly dividing the task and working areas through verbal communication, or by observing where others were working. Collaborators worked predominantly individually or in pairs in different areas of the workspace, moving fluidly between closely and loosely-coupled work styles. When questions arose or global changes had to be negotiated, all groups came together and evaluated a solution, performing coordinated actions on the workspace. Coordinated actions were also common when participants helped each other out. Such peer aid would either be requested (e. g."Could you remove X from that community while I ..." or would be voluntarily offered by observing the actions of others (e. g."Let me do that").

In groupware systems accidental conflicts of concurrent input can be disruptive and special control mechanisms have been suggested [2, 7]. Since we chose not to provide any conflict control mechanism, we logged potential sources of interaction conflicts to validate our choice. These included two or more participants grabbing the same node or matrix, or trying to lasso select an item that was currently worked on by another person. These conflicts occurred rarely. In 10/11 groups a maximum of two conflicts were logged with concurrent dragging actions being the most common one (4×). One group had 7 such conflicts, mostly caused by two people interacting with the same matrix concurrently. When discussed in the interview, none of participants perceived the logged conflicts as problems. Outside of the logging, we observed conflicts dealing with inadvertent dropping of elements in matrices or a participant editing matrices after others considered it finished. All these conflicts were solved socially, and some groups even established rules (e. g. "ask before editing a reduced size matrix" "if you see labels don't touch it, that's the rule"). When interviewed, participants felt these conflicts were easily solved and did not interfere much with the task.

*Requests for Improvement:*
Participants perceived little conflicts of interaction. When asked if they would have wished for a mechanism to lock control or indicate ownership of items, all but one group responded negatively.

*Summary:*
Our participants coordinated their actions very fluidly. We feel that our choice of not to include specific protection mechanisms was further justified as conflicts were resolved socially and mistakes could be easily reverted through local or global undo.

### 4.4 Analysis Strategy and Group Insight

One of our original goals, was to determine if our discount environment supported successful collaboration with the visualization. An indicator for successful collaborative visualization use is the establishment of an effective strategy leading to group insight. Group insight is difficult to measure, but can be visible in interactions between participants and with the visual representation, or interview comments like "we found out that ...".

*Observations:*
Although no explicit planning support was given in our environment, most of our participants verbally negotiated their strategies. Almost all groups (9/11) started the task with a short group exploration phase in which initial obvious clusters were identified. The establishment of an analysis strategy seemed to evolve naturally from conversation and participants observing each others' actions.

When asked, all 11 groups reported to have gained new insight from working with the dataset and reported several surprising or confirmatory findings, such as close collaboration patterns between research groups previously thought unconnected, and even findings about their close working environment "I had no idea that many people collaborating in our lab, I even learned things about my own team!". Peer-learning and teaching of these insights occurred often in groups that had an imbalance of shared knowledge. In one group, for example, a participant helped to identify the initial communities and taught others about parts of the dataset they were unfamiliar with, so the work could then commence in parallel.

*Summary:*
We observed participants smoothly establishing an analysis strategy and did not request any additional features for activity planning. Observations and comments showed that our tool helped the group gain insight, teach each other facts about the data, and support knowledge building in the group. We see this as an important part of a successful collaborative data analysis environment.

### 4.5 Work preference

As an indication of successful collaboration, we asked participants whether they preferred conducting this analysis task as a group rather than individually.

*Observations:* The majority (40/44) of participants preferred group work and 4 preferred to do the task alone. Three of the latter were among the most knowledgeable members of their group and felt that they could have done a reasonable job on their own, although they admitted it to be potentially slower. The $4^{th}$ had a completely different opinion than the rest of her group about what criteria to use in forming communities. The participants who preferred group work named as reasons for their preference: shared knowledge (27/44), fun of collaboration (25/44), shared process of forming consensus (6/44), brainstorming (4/44), efficiency (4/44), and shared working styles (1/44). One participant commented that "doing it with 3 people was fun, doing it by myself would be work." In addition, 9/11 groups reported feeling happy with the result of their analysis and the communities they had created.

*Requests for Improvement:*
Most participants stated that additional time and meta-information would have helped to resolve questions about unknown people and improve the visual presentation of the analysis.

*Summary:*
Groups were generally very happy with their collaboration and result of their work. We take this as an indication that the retrofitting was successful for this setting and task and could effectively support collaborative data analysis as perceived by these participants.

### 4.6 Reaction to low-cost environment choices

While observations on collaboration and group insight can establish whether collaboration in our low-cost setup was effective, observations on the usability of the environment can further inform the effectiveness of the retrofitted tool in use.

*Observations:*
One observed strength of the CoCoNutrix visualization was its intuitiveness of interaction. All participants were at some point interacting with the information items and over longer periods of time all mice in a group were in movement concurrently. Participants were comfortable interacting anywhere on the screen. Even though the screen sizes were slightly different, this observation was unaffected. The keyboard as a shared device was typically used by one dedicated scribe who would type in the labels for communities as they were requested. Groups rarely used features that would have created global view changes (undo, redo and a re-layout of the graph), and when they did it was generally after negotiating and obtaining group approval. Five groups never made use of these functions, two groups used them 6×, and the remaining groups used it 2-3×. Participants commented that our low-cost setup of mouse input and large screens supported well their group work.

*Suggestions for Improvement:*
Three groups expressed the need for a second keyboard to avoid interrupting others' work process by asking for a label, or handover of the keyboard. There were 15 requests (from all 44 participants) for functionality that was originally part of NodeTrix and was removed during the retrofitting. These requests were mostly for visual features mentioned earlier, such as highlights, more meta-data, or for additional interactions (such as sorting) on matrices. Participants reported they did not feel the sitting configuration influenced their collaboration, but to further improve communication some would have preferred a slightly curved seating arrangement to be able talk to each other better. In Org C, dealing with a larger network on a slightly smaller display, participants would have preferred a larger screen display or functionality to "push nodes to get more space". Thus, the ratio between the display and network size used in Org C was perceived as a threshold condition for comfortable use.

*Summary:*
While participants requested additional functionality for the system and physical setup, they generally reported to have been well supported in their global task. Lack of interaction capability and the lack of meta-data affected their work efficiency, but the work quality was not generally compromised. We see this fact as proof that our discounted interface was a good compromise for this task.

The requested additional visual and interaction features are difficult problems to solve when multiple users interact with the system. Selection actions can induce input conflicts and parameterizing actions requires consensus as they affect the entire representation. This is the reason we removed them originally in our retrofitting, but further research is necessary to reduce global changes in visualizations or make them less disruptive. While the actual sitting position did not seem to interfere with the collaboration, we found that the display size was very important. Finding the optimal screen size for visualization tasks requires further research attention.

## 5 Discussion

To summarize our findings we return to our initial questions in regards to the utilization of our retrofitted collaborative software.

### 5.1 Assessment of the Results

*Communication:*
We observed frequent interaction between analysts, with the data and with the visualization. Analysts slipped in and out of interaction with the full group and with varying subgroups as work progressed. This confirms previous CSCW studies on information visualization in other settings where frequent switching between loosely and closely-coupled work was observed (e. g., [5, 10]). Active data interpretations, discussion and negotiations occurred throughout the collaboration while participants interacted on all areas of the display. This finding is important as information visualization analysis requires seeing and interacting with all parts of the representation to explore all available data and avoid misleading or incomplete data analysis.

*Conflicts:*
Control mechanisms to avoid interaction conflicts have been studied and suggested [2, 7] for co-located collaboration. Even though we included no specific control mechanisms, we observed and logged few interaction conflicts between participants, echoing previous findings [11] that people naturally avoid interfering with each other by spatially separating their actions in the workspace. Moreover, participants did not request any additional control mechanism features, so our decision to leave them out was further justified.

*Awareness:*
The visualization mediated the awareness of decisions made about the data and helped group members to build on each others' work. Factors like labeling were used to help the group coordinate which data aspects were decided upon and which were still in flux. Yet, several additional awareness features were asked for and this is a promising direction for further work in collaborative visualization.

*Group Insight:*
The hybrid nature of the visualization helped in facilitating, and hence observing group insight, as it captured the evolving construction of knowledge within the group. We noticed that participants did not simply view a matrix as a different representation of a group of researchers in the dataset—a matrix expressed a particular research group and together with a label became the result or artifact of choices made by one or several participants during the collaboration. This artifact was then visible to others and facilitated the emergence of a common understanding of the data within the group. Thus the visualization evolved and became an archive of the par-

ticipants' process, what work was completed or needed discussion, and of the participants' insight, the interpretations and meaning that they had given together to specific information in the dataset. Similar observations have been made for collaborative communication and learning in online communities [9]).

*Qualitative Feedback:*
Both the chosen physical environments (use of a large back projected display and sitting arrangements) and the use of multiple mice for interaction was positively received by our participants. Together with other positive responses and feedback regarding the usability of the system, we feel confident that NodeTrix was sufficiently retrofitted to enable effective collaboration.

### 5.2 Impact for Other InfoVis Systems

The study results have implications for other information visualization researchers or designers considering how to adapt their own single-user applications for co-located collaborative work settings. Our study looked specifically at supporting participants interacting at the same time in a low-cost setup with mouse input. We think that this is a feasible collaborative setup for many different visualization systems, yet there is an important interplay between the success of this type of low-cost retrofitting and the types of interaction that already exist in the system.

One very important aspect that a retrofitted tool needs to support is the awareness of what has been looked at, analyzed, and about which data items decisions have been made. In our case, this was mostly facilitated through the hybrid nature of the visualization. We, therefore, hypothesize that information visualizations in which group members can give the data meaning by either transforming data items into different representations (as in our case), or by annotating and marking them (e. g. through spatial positioning or graphical markers), will not require much additional functionality to be added. It is likely that a large number of other 2D network and graph visualizations will benefit from our collaborative retrofitting. Most such tools already allow for free spatial repositioning, which could be used to annotate or mark data by changing their position. Coupled with user-specified visual clustering group insight could be captured and group coordination and communication helped.

In observing participants using CoCoNutrix, we saw fluid transitions from parallel individual work to group work or negotiation. The ability to display an overview of a dataset, as well as allow for detailed work in different areas of this overview contributed to better parallel work, another crucial aspect to support in retrofitted tools. Contrary, visualizations where views or global changes need to be discussed and negotiated often (e. g. 3D visualizations with a single data view, where navigation is crucial) may benefit less from this type of collaborative setup.

Finally, one of the attributes of NodeTrix that enabled participants to collaborate effectively was the reversibility of all actions (e.g. adding/removing community actors, creating/splitting matrices, etc). This way, our groups avoided using functions that have a global effect in the layout and disturb the work of others (like undo and graph layout). It is thus important that the retrofitted information visualization tool provides alternative lightweight mechanisms for correcting errors.

## 6 Conclusions

In this paper we have reported on challenges and results in extending a single-user information visualization tool for multiple-users, for use in a realistic low-cost collaborative environment. Specifically, we extended NodeTrix, a social network visualization tool, to enable multi-mouse interaction. Our research indicates that retrofitting of existing information visualization software to include discount multi-input interaction is very doable (given appropriate open source software). The resulting low cost collaborative environments can be reasonably functional and well worth assembling. However, we caution that the overall success of retrofitted collaborative software is very dependent on an identified set of interaction capabilities of the existing software. To refine our results and to be able to make further recommendations for low-cost retrofitting, it needs to be studied how other types of visualizations fair in a retrofitted scenario, and how they are used in real-life situations where the outcome of the analysis has a big impact on participants' everyday work. Our work and existing guidelines for collaborative information visualization can be a useful starting point. Future studies may refine this knowledge.

## References


[1] P. Baudisch, E. Cutrell, and G. Robertson. High-Density Cursor: A Visualization Technique that Helps Users Keep Track of Fast-Moving Mouse Cursors. In *Proceedings of Interact*, pages 236–243, Amsterdam, The Netherlands, 2003. IOS Press.

[2] C. Gutwin and S. Greenberg. The Mechanics of Collaboration: Developing Low Cost Usability Evaluation Methods for Shared Workspaces. In *Proceedings of WETICE*, pages 98–103, Los Alamitos, 2000. IEEE Computer Society.

[3] N. Henry, J.-D. Fekete, and M. J. McGuffin. NodeTrix: A Hybrid Visualization of Social Networks. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1302–1309, Nov./Dec. 2007.

[4] P. Isenberg and S. Carpendale. Interactive Tree Comparison for Co-located Collaborative Information Visualization. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):1232–1239, Nov./Dec. 2007.

[5] P. Isenberg, A. Tang, and S. Carpendale. An Exploratory Study of Visual Information Analysis. In *Proceedings of CHI*, pages 1217–1226, New York, 2008. ACM Press.

[6] Jinput. Java Game Controller API. Website (`https://jinput.dev.java.net/`), 2008. Accessed: May.

[7] M. R. Morris, K. Ryall, C. Shen, C. Forlines, and F. Vernier. Beyond Social Protocols: Multi-User Coordination Policies for Co-located Groupware. In *Proceedings of CSCW*, pages 262–265, New York, 2004. ACM Press.

[8] A. Noack. Energy-Based Clustering of Graphs with Nonuniform Degrees. In *Proceedings of GD*, pages 309–320, Berlin, 2005. Springer-Verlag.

[9] G. Stahl. *Group Cognition*. MIT Press, 2006.

[10] A. Tang, M. Tory, B. Po, P. Neumann, and S. Carpendale. Collaborative Coupling over Tabletop Displays. In *Proceedings of CHI*, pages 1181–1190, New York, 2006. ACM Press.

[11] E. Tse, J. Histon, S. D. Scott, and S. Greenberg. Avoiding Interference: How People Use Spatial Separation and Partitioning in SDG Workspaces. In *Proceedings of CSCW*, pages 252–261, New York, 2004. ACM Press.

[12] S. Wassermann and K. Faust. *Social Network Analysis*. Cambridge University Press, 1994.

# A Sidebox: Related Work

## A.1 Collaborative Retrofitting for Information Visualization

Connecting several mice, keyboards or other input devices to one desktop computer is limited due to support issues at four levels:

1. Operating systems: some systems such as Windows explicitly limit the support for multiple-mice and keyboards due to security issues. Others (including Linux, most flavors of Unix and MacOs) allow the management of extraneous input devices but with a different level of support than the standard input devices. For example, these systems do not provide any cursor feedback for extraneous positional devices so this capability has to be done by applications or window-manager extensions.

2. Low-level libraries for access to USB devices or game devices allow the reading of input devices in system-dependent ways. In the recent years, there has been some progress in trying to standardize access to these libraries with projects such as JInput for Java.[1] There are issues raised by these libraries because the window manager applies many hidden operations to the standard input devices (acceleration management for relative positional devices, key mappings for keyboard devices). These are difficult or impossible to emulate through external libraries, except when integrated with the window systems (e. g. the X Input Extension.[2])

3. Graphical Toolkits such as Swing for Java or Qt for C++ provide support for GUI components (Widgets) and input managements. Like most of the toolkits, they only manage a limited set of input devices through typed events. Even for well supported devices, like the mouse, they usually don't support more than one predictably. Only recently have there been attempts at supporting multiple input devices at this level [10, 3, 11].

4. Applications: applications, like MMM [2], supporting co-located collaboration have been built from scratch due to the lack of toolkit and library support. However, newer generations of co-located applications have been trying to build toolkits or rely on special toolkits to simplify the design of this type of applications.

Some researchers have described their process of retrofitting single-user applications for collaborative use; however, only few have specifically studied this in the co-located information visualization context and considered the implications of offering multiple independent inputs.

Forelines describes collaborative retrofitting for Jmol for molecular visualization [6] and Google Earth [5]. Both tools were adapted for a multi-user and multi-display environment. Their research focuses on describing how the visualization was adapted to be shown and interacted with in a co-located scenario using different views on different display configurations. Problems of concurrent input are solved through a single-user floor control policy that allows only one person to interact with a single display at a time.

Comparing distributed and co-located information visualization work, Mark and Kobsa [12] studied collaborative use of pre-existing information visualization tools and found that group performance increased with the transparency of the system. Collaborative retrofitting for this study was minimal. While a large shared display was used in the co-located setting, participants also shared a single input.

Some graphical toolkits managing scene graphs (e. g. [1]) or information visualization [4, 7], use the Interactor abstraction to implement modular interaction techniques. They decouple display management and interaction, simplifying the retrofitting for multiple inputs. Moreover, they provide support for a layering mechanism on which to draw additional cursors and highlights without interfering with the standard display management. We implemented our extensions with the Infovis Toolkit [4] on which NodeTrix is implemented.

## A.2 Social Network Analysis

Our study is situated in the field of social network analysis. Any collection of persons or organizations connected by relations is a social network. In the last decade, the popularity of social networking applications has dramatically increased. Social network analysis tools are used by intelligence agencies to monitor terrorists networks, by epidemiologists to study transmission networks and detect and contain disease outbreaks, or company managers and research institutes to examine the flow of communication between their employees and the strength of their employees' collaboration. In our work we focus on visual analysis of social networks that is more exploratory in nature.

With the increasing popularity of social networking and the progress of Internet technologies, many systems emerged to visualize and analyze social networks.[3] The two most common types of representations are node-link diagrams and matrix-based representations. Node-link diagrams are commonly used to understand the global structure of the network while matrices have been shown to improve readability for detailed community analysis [9].

From trial demonstrations of social network analysis software, we have empirical evidence of spontaneous analysis sessions of collocated colleagues that came together over a small shared display to make sense of, discuss, and explore their data. Similar observations were reported by Heer and boyd [8] in their study of Viszter, a visualization tool for online social networks in a public setting. Social network analysis can benefit highly from collaborative analysis through the combination of knowledge, expertise, and skills as well as the combined cognitive power of several analysts that can tackle larger networks together.

# References


[1] B. B. Bederson, J. Meyer, and L. Good. Jazz: An Extensible Zoomable User Interface Graphics Toolkit in Java. In *Proceedings of UIST*, pages 171–180, New York, 2000. ACM Press.

[2] E. A. Bier, S. Freeman, and K. Pier. MMM: The Multi-device Multi-user Multi-editor. In *Proceedings of CHI*, pages 645–646, New York, 1992. ACM Press.

[3] P. Dragicevic and J.-D. Fekete. Support for Input Adaptability in the ICON Toolkit. In *Proceedings of ICMI*, pages 212–219, New York, 2004. ACM Press.

[4] J.-D. Fekete. The InfoVis Toolkit. In *Proceedings of InfoVis*, pages 167–174, Los Alamitos, 2004. IEEE Computer Society.

[5] C. Forlines, A. Esenther, C. Shen, D. Wigdor, and K. Ryall. Multi-User, Multi-Display Interaction with a Single-User,


[1] https://jinput.dev.java.net/

[2] http://en.wikipedia.org/wiki/DirectInput

[3] See http://www.insna.org/software/index.html for an overview.

Single-Display Geospatial Application. In *Proceedings of UIST*, pages 273–276, New York, 2006. ACM Press.

[6] C. Forlines and R. Lilien. Adapting a Single-User, Single-Display Molecular Visualization Application for Use in a Multi-User, Multi-Display Environment. In *Proceedings of AVI*, pages 367–371, New York, 2008. ACM Press.

[7] J. Heer, S. K. Card, and J. A. Landay. Prefuse: A Toolkit for Interactive Information Visualization. In *Proceedings of CHI*, pages 421–430, New York, 2005. ACM Press.

[8] J. Heer and danah boyd. Vizster: Visualizing Online Social Networks. In *Proceeings of InfoVis*, pages 33–40, Los Alamitos, 2005. IEEE Computer Society.

[9] N. Henry, J.-D. Fekete, and M. J. McGuffin. NodeTrix: A Hybrid Visualization of Social Networks. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1302–1309, Nov./Dec. 2007.

[10] J. P. Hourcade, B. B. Bederson, and A. Druin. Building KidPad: An Application for Children's Collaborative Storytelling. *Software: Practice & Experience*, 34(9):895–914, 2004.

[11] S. Huot, C. Dumas, P. Dragicevic, J.-D. Fekete, and G. Hégron. The MaggLite Post-WIMP Toolkit: Draw it, Connect it and Run it. In *Proceedings of UIST*, pages 257–266, New York, 2004. ACM Press.

[12] G. Mark and A. Kobsa. The Effects of Collaboration and System Transparency on CIVE Usage: An Empirical Study and Model. *Presence*, 14(1):60–80, Feb. 2005.