Representations and Metaphors for the Structure of Synchronous Multimedia Collaboration within Task-Oriented, Time-Constrained Distributed Teams

John M. Linebarger, Andrew J. Scholand, Mark A. Ehlen
Sandia National Laboratories, Albuquerque, New Mexico, USA
{jmlineb, ajschol, maehlen}@sandia.gov

Abstract

Based primarily on the results of a month-long experiment and a crisis management exercise, synchronous multimedia collaboration within a task-oriented, time-constrained distributed team appears to exhibit three layers of structure. The first layer is episodic, and results in collections of related multimedia collaboration artifacts that can be called "chapters" or "scenes" in the collaboration. The second layer is the multivalent nature of collaboration, in which collaboration conversations at multiple subgroup levels take place at the same time. The third, top-level, layer is the agenda that drives the collaboration. The implications for the design of synchronous collaboration systems are that multiple views, representations, and metaphors for this conversation structure are needed. Chapter views, subgroup views, and agenda views are presented as alternative packaging mechanisms and entry points into the collaboration data. Other metaphors and presentations include the collaboration tree and infinitely recursive conference room, as well as network graphs of subgroup structure and agenda-based group awareness.

1. Introduction

James Watson was a young man in a hurry. The year was 1953, and Watson and his scientific partner, Francis Crick, were in a race to decipher the fundamental building block of life, deoxyribonucleic acid (DNA). But they were not alone. Fresh from his discovery of the alpha-helix structure of protein, the famed chemist Linus Pauling had also turned his attention to DNA. At stake was nothing less than the Nobel prize.

But Pauling took an uncharacteristically wrong turn in his chemical analysis of DNA, and Watson and Crick redoubled their efforts. Aided by X-ray data that seemed to suggest a helical structure, Watson built numerous cardboard models of potential structures to fit against known results and constraints. New data suggested a double structure of some sort. On the morning of February 28, Watson created a cardboard model that finally worked. The structure of the data of life was a double helix, which Watson described as "too pretty not to be true" [14].

The discovery of underlying structure can have enormous benefits, from presentation to prediction to pedagogy. In the case of DNA, it was quickly realized that each strand of the double helix can function as a template for copying the genetic code for the entire organism, and that this replication mechanism formed the engine of life itself. The thesis of this paper is that synchronous multimedia collaboration within task-oriented, time constrained distributed teams also exhibits a rich structure, and that explicit representations and metaphors for that structure are needed. In the sections that follow, background to several collaboration experiments will be presented, the software collaboration framework will be described, the experiments themselves will be detailed and the results analyzed, the observed structure of collaboration will be delineated, the implications for the design of synchronous collaboration systems will be addressed, and previous work will be surveyed.

2. Background

The National Infrastructure Simulation and Analysis Center (NISAC), a program under the United States Department of Homeland Security's Information Analysis and Infrastructure Protection (IAIP) directorate, provides advanced modeling and simulation capabilities for the analysis of critical infrastructures, their interdependencies, vulnerabilities, and complexities. These capabilities help improve the robustness of critical infrastructures of the United States by aiding decision makers in the areas of policy analysis, investment and mitigation planning, education and training, and near real-time assistance to crisis response mobilizations.

NISAC and related programs are frequently called upon for Fast Analysis and Simulation Team (FAST) exercises to assess the impact of a potential event on critical infrastructures. The primary metrics for this high-pressure, time-constrained collaboration (which can be characterized as "collaboration in a crisis") are time to solution and quality of solution. A primary time consumer is the information exchange required to establish a common mental model (also called a "common analysis picture") of the problem(s) and solutions(s) between all participants in the exercise.

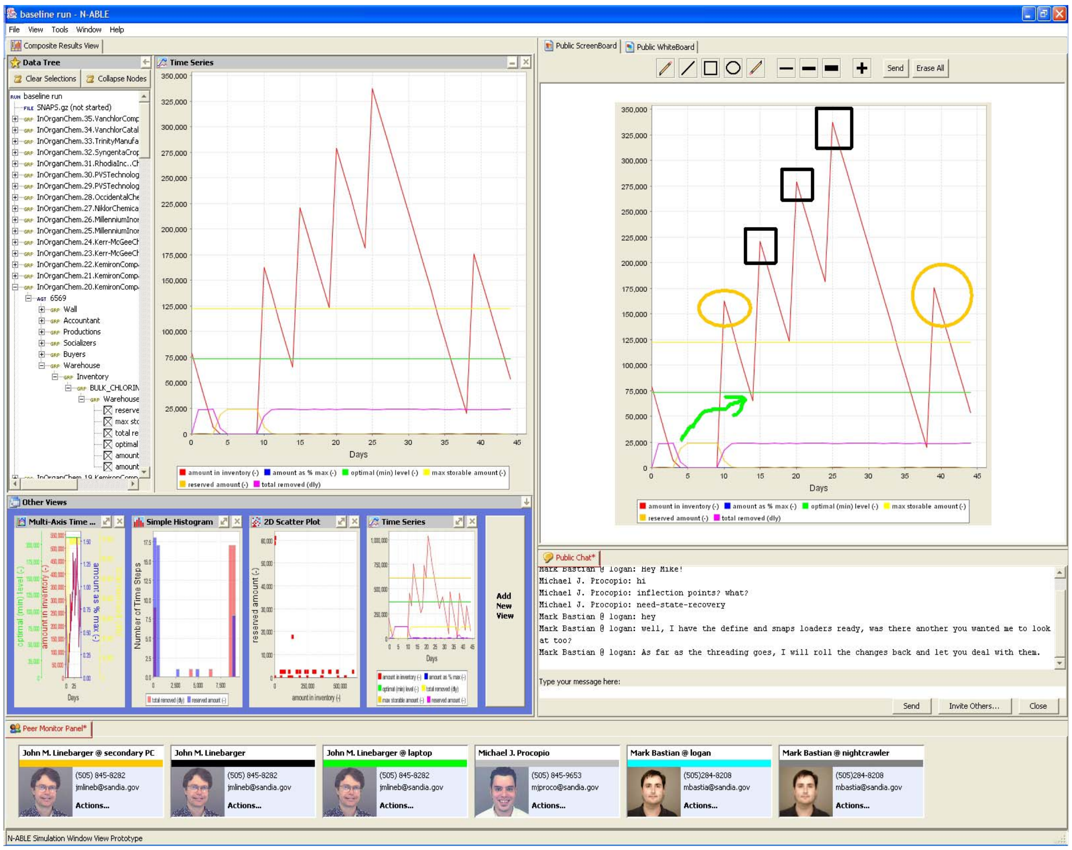

Figure 1. N-ABLE™ collaboration, showing group awareness, chat, and screenboard

3. Collaboration Software Framework

To support such FAST exercises, the GroupMeld™ software framework for synchronous collaboration has been developed. The goal of this framework is to facilitate real-time collaborative interaction, in order to allow geographically-distributed analysis teams to integrate multiple perspectives and quickly converge on a shared view of the problem(s) and potential solution(s).

GroupMeld™ was developed using the Java programming language, and has been deployed as a programmable collaboration library with an application programming interface (API). The library enables collaboration through a particular software application that uses the library, thus forming an application-centered collaboration community. The NISAC Agent-Based Laboratory for Economics (N-ABLE™) tool, an agent-based economic modeling and simulation package [4], is the first NISAC project to use the library. A screenshot of its use inside the N-ABLE™ simulation application is shown in Figure 1 above.

The collaboration capabilities provided by GroupMeld™ include:

- Pictorial awareness of other members of the virtual team, with visual status change indicators
- Real-time chat

- Shared screen images with collaborative annotation capability (a.k.a. “screenboard”)
- Shared whiteboard
- File transfer
- Audible paging capability (to get someone’s attention in case they are working on something else).

The collaboration scope of each capability is chosen from three levels, which can co-exist simultaneously:

- Full group (“public” collaboration)
- Subgroup (“restricted” collaboration)
- Person-to-person (“private” collaboration).

4. Description of the Experiments

To test the efficacy of the use of the collaboration framework within a task-oriented, time-constrained distributed team, a month-long series of experiments were conducted with the N-ABLE™ analysis group. Since FAST exercise problems can vary in time duration from several hours to several days (based on the time-to-answer specified by the customer), the factor level that was varied in the experiments was the time duration of the analysis problem. Three time durations were investigated: short (four hours within a single day), medium (eight hours, spread over two calendar days) and long (twenty-four hours, spread over five calendar days). Two replications of each factor level were conducted. Each problem was reasonably independent of the others, and ranged from “Is the simulated supply chain in balance? If not, why not?” to “Analyze the causes of the bullwhip effect in a multi-level commodity supply chain.” A four-hour pilot experiment was performed in order to exercise the data gathering capabilities and equalize any training effect.

The subject pool consisted of six N-ABLE™ analysts who already had experience with the N-ABLE™ application and its collaboration capabilities. The team contained a mix of economists and software developers with expertise in economics; not only does the team use N-ABLE™ for its modeling, simulation, and analysis activities, but it also develops the tool itself. Each of these four kinds of activities was performed during the experiment. Between four and six analysts participated every day an experiment was scheduled; four was considered a quorum. However, the composition of the team was not constant for each experiment because of real-world scheduling constraints. Most of the analysis team was co-located in the same hall. But one of the participants was located downstairs in the same building, and half of the time another member of the analysis team was located in a satellite office almost three hundred miles away.

The data collected by the experiment included a transcript of the group chat conversations; a timestamped transaction log of every collaboration transaction performed by each group member; and a post-experiment questionnaire.

Although the GroupMeld™ framework supports the formation of collaborative subgroups, the N-ABLE™ interface at the time of the experiments did not utilize that capability. As a result, collaboration during the experiments took place at only two levels: full-group (“public”) and person-to-person (“private”).

5. Results of the Experiments

Both quantitative and qualitative results came out of this set of experiments. A thorough analysis of the quantitative results is provided in [10]. Of the qualitative results, perhaps the most important is that group insight (*i.e.,* formation of common mental model of problems and solutions) often occurred while the group was discussing and annotating a shared screen image. These screen images generally contained a graphical presentation of the output of the simulation. Closely related is the observation that collaboration generally did not proceed linearly but instead proceeded episodically, in chunks or chapters or cinematic “scenes.” The line of demarcation between episodes or scenes was often the transmission of a shared screen image, around which subsequent collaboration coalesced. At times *several* screen images, annotations, transferred files, and chat transcripts formed a “conversation package,” a collection of related collaborative interactions that constituted a collaboration chapter. And frequently the series of collaboration chapters exhibited a hierarchical structure, such that the chapters were really subchapters of a larger chapter, which often constituted one of the tasks in the implied task list or agenda that drove the analysis for the experiment.

====== TRANSMISSION OF SCREEN IMAGE =====
Sue: OK so what is on the screenboard is the outstanding order amount for the first supermarket in the list
Verne: That's Natrona
Sue: this to me indicates a stable ordering pattern...so he is not frantic
Andy: I'm sorry to draw the conversation back to an earlier comment, did you all figure out why the supermarkets were not happy very early in the sim? Looks like day 8?
Sue: I haven't checked yet Andy
Andy: On screenboard now makes sense
Andy: You get 1 pallet in transit for 3 days
Andy: every time you order
Sue: right
Andy: which is infrequently, since you consume much less than order size
Andy: looks right
Sue: yes
Sue: would I confuse everyone if I put up a new graph now
Andy: go ahead
====== TRANSMISSION OF SCREEN IMAGE =====
Sue: this graph is the frequency of calls he makes to find butter
Verne: I would say he's frantic in the first 10 days or so
Andy: Deb: intransit does not count against amount in inventory until received at location of firm
Sue: right
Sue: Verne: agreed
Andy: wow
Andy: does market structure change at day 11 somehow?
Verne: That's why the supermarkets weren't happy in the first 12 days.
Sue: Verne: correct
Andy: So lets see if this hypothesis works for you guys
Deb: so you are cross checking call against utility?
Sue: Andy: I think this is an artifact of the initial inventories not all being the same at each butter producer
Andy: although aggregate demand is balanced, order chunks are very large multiples of individual demand
Sue: Deb: remember you did that earlier with your graph of 4?
Andy: therefore, first supermarkets to place orders suck large quantities out of market
Deb: yes
Andy: causing starvation for other supermarkets until they can get 1 order in
Sue: Andy: agreed
Verne: sounds plausible so far
Andy: eventually, since ordering is infrequent after you get 1 pallet in, system settles down
Andy: its an interesting consequence of having a high order qty
Sue: this behavior is very similar to what we saw the packagers in chlorine do
Andy: similar behavior would be expected after every disruption
Sue: also true
Andy: it suggests that one mitigation strategy would be to offer small, frequent shipments
Andy: ?
Sue: that is easy enough
Sue: to do I think

Figure 2. A Chapter from the Chat Transcript Showing Collaborative Creation of Common Mental Model of Problem and Solution

An illustration of some of the observations above is provided in Figure 2, which contains a chapter from the chat transcript for one of the experiments. Each section of the chapter begins with the transmission of a screen image, and subsequent interaction is based on that shared screen image. The second section demonstrates how collaboration around a shared screen image triggered the creation of a common mental model of a problem, and then a solution to the problem. Note that the line of demarcation between sections of a collaboration chapter, and even between chapters, was fuzzy at times instead of sharp. The fuzziness was caused by an overlap in the topics that constituted a particular section or chapter, as some collaborators would move ahead to the next topic while others were closing out the previous one.

6. Observed Collaboration Structure

From empirical observation of the experiments described above, as well as a crisis management exercise described below, we propose a collaboration structure taxonomy that applies to synchronous multimedia communication within a task-oriented, time-constrained distributed team. This taxonomy is depicted in Figure 3 on the following page. At the top of the taxonomy is the agenda, which consists of a list of agenda items, nested hierarchically. Since the goal of the agenda in this domain is to accomplish a task, synonyms for “agenda” include “hierarchical task list” or “work breakdown structure” in project management terms. For each agenda item at the lowest level of nesting, a subgroup is formed by the people who are working on the agenda item. A chapter is a packaging of related collaboration transactions performed by a subgroup. If a subgroup consists of a single person, no collaboration at that level of the agenda occurs.

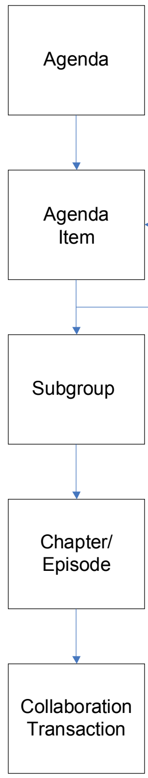

Figure 3. Collaboration Structure Taxonomy

6.1. Episodic

Characterizing collaboration as episodic reflects the observation that it progresses not linearly but in chunks of related interactions that can be called "chapters" or "scenes." Several kinds of chapter delimiters were observed during the experiments. These delimiters can be categorized as either natural (arising from the content of the collaboration itself) or synthetic (artificially induced transitions in the collaboration due to exogenous factors). Natural chapter transitions occurred for several reasons—a shift in chat topic or screen board image content, a time gap in the collaboration conversations, or the introduction of what became recognized as a stereotypical topic. Such topics included an opening or closing summary to an experiment, or a code design interlude. Synthetic chapter breaks occurred either by vote of the team, by the decision of the *de facto* team leader, or when the time allocated to the current agenda item expired.

6.2. Multivalent

The second structural characteristic of collaboration conversation that was noticed during the collaboration experiments was its multivalent nature. Private conversations were taking place at the same time as public conversations. Because intermediate collaborative subgroups were not supported by the N-ABLE™ interface at the time of the experiments, the incidence of these non-public conversations was quite low. However, another exercise involving a task-oriented, time-constrained distributed team can serve to bring the multivalent nature of such collaboration into sharp relief.

On 14 December 2004, the Critical Decision Institute organized a training event in the city of Portland, Oregon. The purpose was to simulate a series of critical infrastructure disasters and mass casualty events, beginning with an explosion on the Steel Bridge, in order to give public officials and first responders some hands-on experience with crisis management. The simulation was driven by a software system named DEMA (for Deus Ex MAchina) and lasted for four hours. During the exercise two Emergency Operations Centers (EOCs) were simulated in two adjacent rooms, one representing the City of Portland, the other Multnomah County. Approximately ten people participated in the exercise, in roles ranging from Mayor, Fire Chief, Police Chief, FBI, and Coast Guard (in the City of Portland EOC) to County Commissioner and Sheriff (in the Multnomah County EOC). The two EOCs were linked by phone, and each had video walls that displayed simulated feed of the disaster events from cable news services.

What was remarkable to several observers of the exercise was the large number of simultaneous conversations, at several different levels, that persisted for long periods of time during the simulation. At the highest level was the phone link between the two EOCs; that phone link was active for almost the entire duration of the simulation. At the next level was the scope of conversation defined by the EOC itself. A third level was the collaborative subgroups within the EOC which were formed by conversations between adjacent participants; such proximity occurred either accidentally (due to the seating positions taken at the beginning of the exercise) or intentionally (for example, when someone got up to move next to someone they wanted to talk to). Such subgroups could be nested, and were fluidly formed and dissolved during the exercise. And the lowest possible level occurred when someone chose to work alone for a period of time.

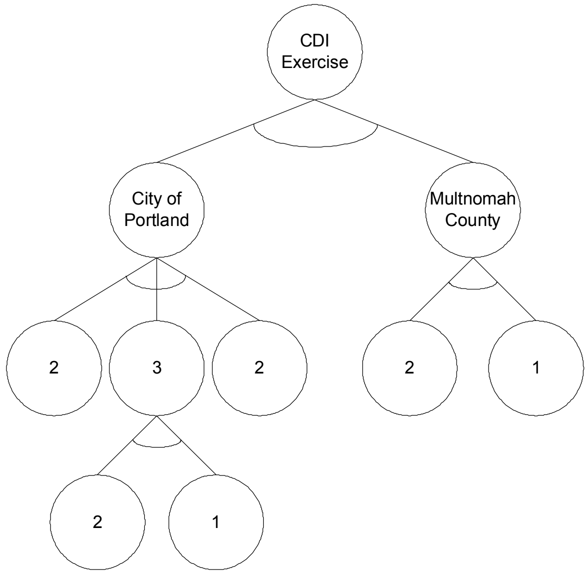

Figure 4. Collaboration Tree Snapshot for Critical Decision Institute Exercise

Figure 4 displays a collaboration tree representation of a snapshot of the CDI exercise. A collaboration tree is an "and/or" tree that provides a graphical portrayal of the subgroup structure at an instant in time ([8] and [9]). The "and" branches indicate complementary subgroups, and are denoted by arcs between them; the "or" branches indicate competitive subgroups, and lack such an arc. The number of subgroup members in that branch of the collaboration tree is displayed inside each node. At the top level in Figure 4 is the CDI exercise itself; the two EOCs formed the second level; and recursively nested subgroups within the EOCs formed the lower level subgroups.

Previous research has indicated that the structure formed by nested collaborative subgroups generally reflects the hierarchical structure of the task ([8] and [9]). Since completing a task generally requires the accomplishment of a recursive set of subtasks, our operating hypothesis, which is reflected in the taxonomy in Figure 3, is that the agenda or task list forms the highest level structure of the collaboration, followed by subgroup level, then chapter within subgroup. However, a future experiment with explicit support for agenda items and fully recursive subgroups is necessary to validate this hypothesis.

6.3. Agenda-Driven

The collaboration observed in these experiments and exercises did not occur in a vacuum; instead, it was goal-oriented and task-driven. As such, it generally followed an agenda, either explicit or implied. A key observation is that the chapters in the collaboration generally tracked the progress of the agenda. The default or implied agenda (which was the one most frequently used) was a four-step iterative process that corresponded to the four primary activities of the team—modeling, simulation, analysis, and software development. First the simulation model was created using XML (eXtensible Markup Language). Next, the model was submitted to the simulation engine. Then the results of the simulation were validated via a sampling procedure—key outputs of various agent types for representative firms at each level of the supply chain were displayed graphically and shared with others using the screenboard. This validation process was called a "deep dive." Finally, if anomalous results were discovered, a review of the simulation software code was performed, which often resulted in code changes. This would trigger a new iteration of the agenda, in which the simulation model would be resubmitted to the simulation engine to run against the updated code, and the results revalidated.

7. Implications for Design

Since synchronous multimedia collaboration conversations within task-oriented, time-constrained distributed teams appear to exhibit the structure proposed in Figure 3 (at least in our experiments and exercises), then new representations, metaphors, and presentation mechanisms for such conversations are needed that take these structures into account. In particular, chapter views, subgroup views, and agenda views are proposed as different entry points into the same transaction data during a collaboration session.

7.1. Episodic

The view proposed for the episodic layer of collaboration structure is the chapter view. As depicted in Figure 5 on the following page, the chapter view is a collection of all multimedia artifacts related to that chapter of the collaboration. A VCR interface could be provided to step through the sequence of chapters. The motivation for this view is the need to organize and store the record of a collaborative analysis session for later retrieval and review. This need was frequently observed during the month-long collaboration experiments, especially during the later stages of a long duration analysis. It is tempting to envision an automatic chapter or episode detection mechanism. However, because of the fuzzy nature of chapter boundaries discussed above, we are skeptical of the actual utility of such an automatic mechanism. The involvement of humans in determining chapter boundaries, either by vote or by decree, would seem to be required. One approach might be to assign the role of "film director" to one of the collaborators, whose responsibility would be to "wrap a scene" by fixing the boundaries of a chapter.

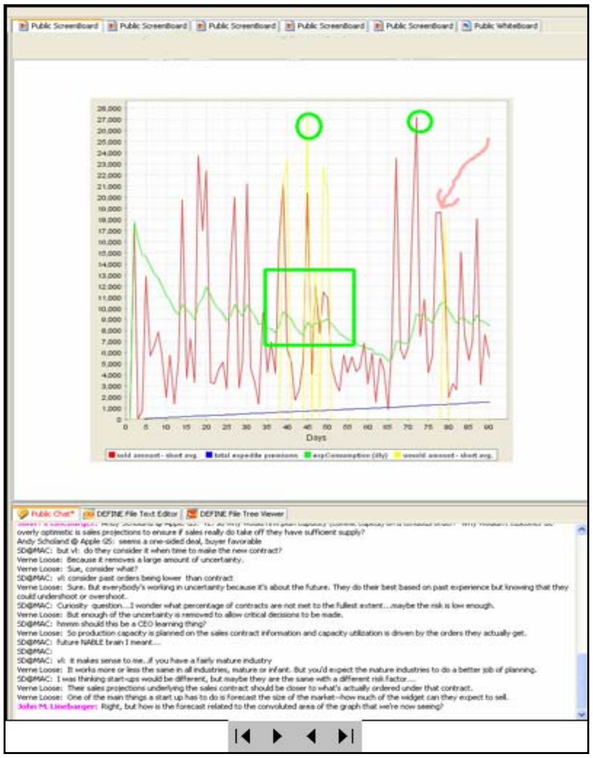

Figure 5. Chapter View with VCR Interface

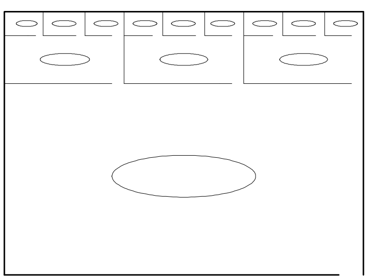

Figure 6. Infinitely Recursive Conference Room

7.2. Multivalent

Multiple metaphors and representations are suggested for this layer of structure, which refers to the multiple levels of subgroup conversations that exist simultaneously during task-oriented collaboration. Essentially, each subgroup forms its own collaboration context. The first metaphor is the collaboration tree, which was depicted in Figure 4 on the previous page. A second is the infinitely recursive conference room, which in many ways is the inverse of the collaboration tree [8]. As shown in Figure 6, an infinitely recursive conference room consists of a table in a conference room surrounded by breakout rooms, which themselves consist of a table in a conference room surrounded by breakout rooms, and so on.

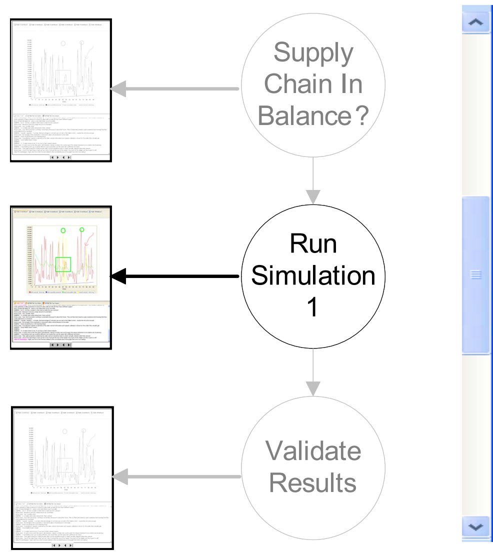

Figure 7. Vertical Slider which Foregrounds Subgroup Chapters where Membership is Based on Task Hierarchy

Several interface representations are proposed. The first is a slider-based interface that foregrounds the collaboration context and chapter sequence for the subgroup selected by the slider. Two examples of this representation are provided. Figure 7 depicts a vertical slider that foregrounds the subgroup context for subgroups whose membership is based on task hierarchy, while Figure 8 on the following page portrays a horizontal slider for subgroups whose membership is based on role. In role-based subgroup membership a given collaborator may be a member of multiple subgroups at the same level of the collaboration tree, which is generally not the case with subgroups based solely on task hierarchy. The motivation for this kind of representation is the realization that—as we observed during the crisis management exercise—people exist in multiple collaboration contexts at the same time, some of which are organized hierarchically (such as by organization or task breakdown structure) and some of which are not. These slider representations not only assist in the awareness of which collaboration contexts are currently active, but also enable the foregrounding or backgrounding of collaboration contexts as focus and attention shift.

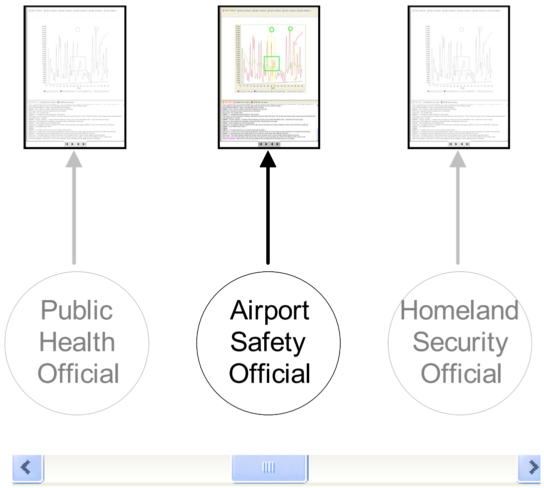

Figure 8. Horizontal Slider which Foregrounds Subgroup Chapters where Membership is Based on Role

A network graph representation is also possible. Figure 9 shows one such example. The collaboration media services are modeled as nodes because they serve as persistent mediums of exchange. The graph data was represented in Pajek format [2] and rendered using the JUNG (Java Universal Network/Graph Framework) library [7] with a Kamada-Kawai layout. Note that three subgroup levels are clearly visible—the full group, the subgroup linked by subgroup chat, and the troika at the bottom who are engaged in person-to-person collaboration. In addition, the directionality of the communication is marked.

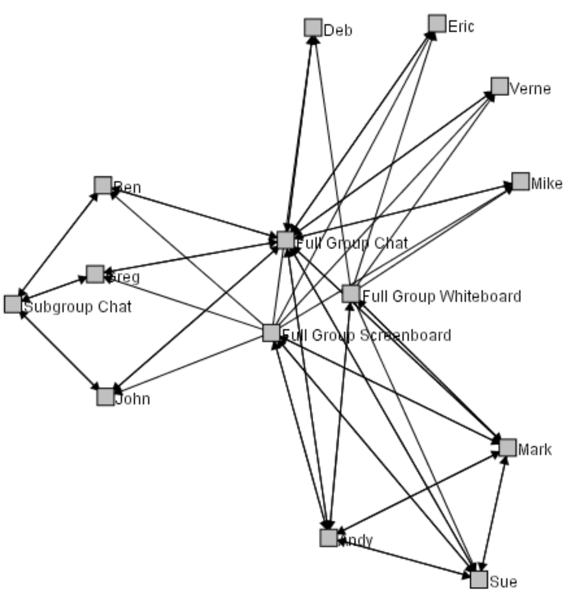

Figure 9. Network Graph of Collaboration Activity and Multivalent Group Structure

Network graphs lend themselves particularly well to dynamic presentations of the structure and progress of collaboration because such the graph can summarize transactions within a time window. Slider controls could be placed on the network graph view that would allow the beginning and ending of the time window to be specified easily. An animated replay of the entire collaboration session as a constantly shifting network graph would also be possible, and could serve not only as a nice dramatic touch but also as a rich source of insight into group dynamics.

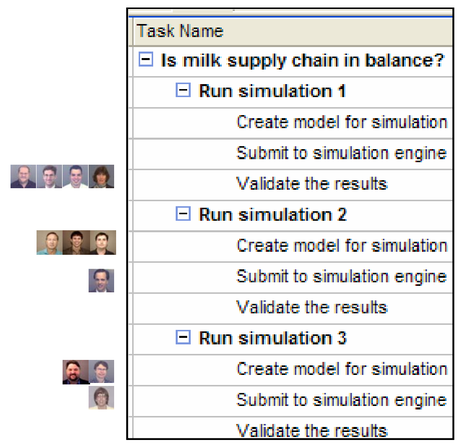

Figure 10. Agenda-Based Group Awareness

7.3. Agenda-Driven

The first representation suggested for this layer of collaboration structure is shown in Figure 10. The agenda itself is used as an awareness mechanism, with avatar-like images of collaborators displayed next to the agenda items on which they are currently working. Note that the current subgroup structure of the collaboration is also indicated by this collection of images. Figure 11 on the following page displays a network graph representation of the same state of the agenda depicted in Figure 10. This time the ISOM (Incremental Self-Organizing Map) layout was used to create the graph. Explicitly representing the agenda for the accomplishment of the task is useful not only for awareness of who is working on which part of the agenda, but also for which agenda items remain to be done before the allocated time expires. This need was observed during the collaboration experiments, particularly during short duration analyses but also toward the end of longer duration analyses. And Figure 12 shows how the three main structure views—chapter, subgroup, and agenda—could be linked. For example, double clicking on an agenda item could bring up a chapter view of the subgroup(s) responsible for that agenda item.

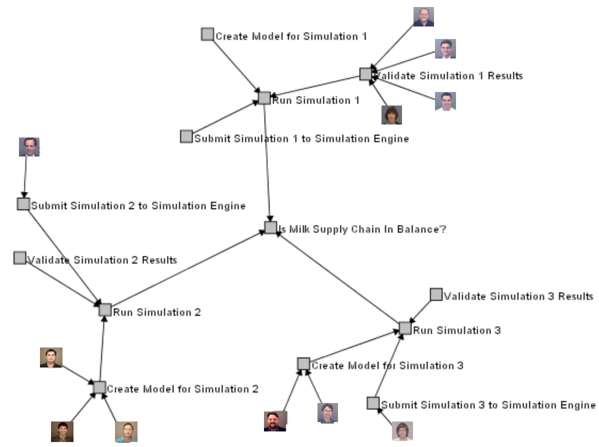

Figure 11. Network Graph View of Agenda-Based Group Awareness

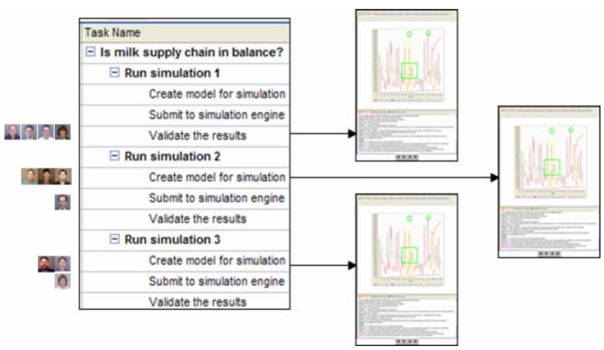

Figure 12. Linking Agenda, Subgroup, and Chapter Views

7.4 Integrated Use Case

It is anticipated that these three levels of representations and metaphors would be manipulated independently but related hierarchically. The agenda would serve as the point of departure for the collaboration, and changes in the progress of the agenda would cascade downward and affect subgroup composition and chapter boundaries as well. Though we hesitate to provide a canonical guide to the use of these representations and metaphors, because we are frequently surprised at the unexpected ways in which collaboration capabilities are actually used, nevertheless we would like to offer a use case that integrates these representations in the context of a FAST exercise.

Ideally, the agenda for the exercise would itself be created collaboratively (see [12] for an example of such capability). Once the agenda was determined and rough time limits allocated, the team would divide into subgroups and begin to work on the agenda items for which they are responsible. Several mechanisms are possible for subgroup formation—moving avatar images to a breakout room in a user interface widget like Figure 6 that represents an infinitely recursive conference room; manipulating a 3D collaboration tree widget like Figure 4 to create a subgroup node and move avatar images into it [9]; or (perhaps most simply) by dragging avatar images next to the appropriate agenda item in an agenda awareness widget like Figure 10. Chapter boundaries for the chapter view in Figure 5 could be determined either by vote of the subgroup members or by decree of the subgroup leader. A vertical subgroup chapter slider like Figure 7 could be used to participate in multiple levels of the collaboration by foregrounding the active collaboration context of each nested subgroup in which a collaborator participates. When time expired for a particular agenda item, or a vote was taken to declare it done, it would be marked as complete, the current subgroup collaboration chapter would be closed, and new subgroups formed around subsequent items on the agenda.

At any time the exercise leader could get a snapshot of the progress of the collaboration by using network graph views like Figures 9 and 11. In addition, at the end of the exercise the network graph views—and the VCR controls of the chapter views—could be used to play back the history of the of the collaboration to support post mortem analysis.

8. Previous Work

Several areas of previous work apply to different aspects of this set of experiments. The domain of team software development has seen much work on the impact of shared mental models (*inter alia*, [5]). Some recent experiments in other domains (such as [11]) have demonstrated a positive connection between a shared mental model and team performance. Much, if not most, collaboration software in the scientific domain is asynchronous in nature; however, the now discontinued Habanero project [1] is a notable exception. Habanero supports both synchronous collaboration and a programmable API. And the use of graphs to represent social networks is just one instance of a more general analysis approach to that domain known as blockmodeling [3].

We are not aware of other work that has directly addressed the structure of synchronous multimedia collaboration within task-oriented, time-constrained, distributed teams in this particular way—as multiple levels of packaging of multimedia collaboration artifacts. However, another approach to structuring synchronous conversation is semantic or topic analysis; [6] is a good example. And the notion that a digital photo can act as a conversational anchor [13] is at least superficially similar to our observation above that collaborative conversations often coalesced around shared screen images.

9. Conclusions and Future Work

Based on the results of a month-long experiment as well as a crisis management exercise, synchronous multimedia collaboration within a task-oriented, time-constrained distributed team appears to exhibit three layers of structure. The first layer is episodic, and results in collections of related multimedia collaboration artifacts that can be called “chapters” or “scenes” in the collaboration. The second layer is the multivalent nature of the collaboration, in which collaboration conversations at multiple subgroup levels take place at the same time. And the third, top-level, layer is the agenda that drives the collaboration.

The implications for the design of synchronous collaboration systems are that multiple views, representations, and metaphors for this conversation structure are needed. Chapter views, subgroup views, and agenda views were proposed as alternative packaging mechanisms and entry points into the collaboration data. Several representations and metaphors were also presented, such as the collaboration tree and infinitely recursive conference room, as well as agenda-based group awareness. Network graph views of subgroup structure and agenda-based awareness were also offered.

A future experiment is needed to evaluate and compare the utility of these representations and metaphors, and to discover if additional structural elements emerge from the collaboration.

10. Acknowledgements

Sandia National Laboratories is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy’s National Nuclear Security Administration under contract DE-AC04-94AL85000.

11. References

[1] Chabert, Annie, Ed Grossman, Larry Jackson, Stephen Pietrowicz, & Chris Seguin. “Java Object-Sharing in Habanero.” *Communications of the ACM* 41.6 (1998): 69-76.
[2] de Nooy, Wouter, Andrej Mrvar, & Vladimir Batagelj. *Exploratory Social Network Analysis with Pajek.* Cambridge University Press, Cambridge, UK, 2005.
[3] Doreian, Patrick, Vladimir Batagelj, & Anuška Ferligoj. *Generalized Blockmodeling.* Cambridge University Press, Cambridge, UK, 2005.
[4] Eidson, Eric D., & Mark A. Ehlen. “NISAC Agent-Based Laboratory for Economics (N-ABLE™): Overview of Agent and Simulation Architectures.” Sandia National Laboratories Technical Report SAND2005-0263, Albuquerque, New Mexico, USA, 2005.
[5] Espinosa, J. Alberto, Robert E. Kraut, Sandra A. Slaughter, Javier F. Lerch, James D. Herbsleb, & Audris Mockus. “Shared Mental Models and Coordination in Large-Scale, Distributed Software Development.” *Proceedings of the International Conference on Information Systems (ICIS 2001)*, 16-19 December 2001, New Orleans, LA, USA, 513-518.
[6] Herring, S. C. “Dynamic topic analysis of synchronous chat.” In *New Research for New Media: Innovative Research Methodologies Symposium Working Papers and Readings*. University of Minnesota School of Journalism and Mass Communication, Minneapolis, MN, USA, 2003.
[7] http://jung.sourceforge.net/
[8] Linebarger, John M., & G. Drew Kessler. “*GroupMorph*: Supporting Group Collaboration Modes for Hierarchically Decomposable Tasks in 3D Environments.” *Conference Supplement to the ACM 2002 Conference on Computer Supported Cooperative Work (CSCW 2002)*, New Orleans, LA, USA, 16-20 November 2002, 179-181.
[9] Linebarger, John M., Christopher D. Janneck, & G. Drew Kessler. “’Leaving the World Behind:’ Supporting Group Collaboration Patterns in a Shared Virtual Environment for Product Design.” Scheduled for publication in *Presence: Teleoperators and Virtual Environments* 14.6 (2005).
[10] Linebarger, John M., Andrew J. Scholand, Mark A. Ehlen, and Michael J. Procopio. “Benefits of Synchronous Collaboration Support for an Application-Centered Analysis Team Working on Complex Problems: A Case Study.” Accepted for publication in *Proceedings of the 2005 International ACM SIGGROUP Conference on Supporting Group Work (Group’05)*, Sanibel Island, FL, USA, 6-9 November 2005.
[11] Mathieu, John E., Gerald F. Goodwin, Tonia S. Heffner, Eduardo Salas, & Janis A. Cannon-Bowers. “The Influence of Shared Mental Models on Team Process and Performance.” *Journal of Applied Psychology* 85.2 (2000), 273-283.
[12] Shum, Simon Buckingham, David De Roure, Marc Eisenstadt, Nigel Shadbolt, & Austin Tate. “CoAKTinG: Collaborative Advanced Knowledge Technologies in the Grid.” *Proc. Second Workshop on Advanced Collaborative Environments*, Eleventh IEEE Int. Symposium on High Performance Distributed Computing (HPDC-11), Edinburgh, Scotland, 24-26 July 2002.
[13] Sit, Ryan Y., James D. Hollan, & William G. Griswold. “Digital Photos as Conversational Anchors.” *Proceedings of the 38th Hawaii International Conference on System Sciences*, 3-6 January 2005, Big Island, HI.
[14] Watson, James D. *The Double Helix: A Personal Account of the Discovery of the Structure of DNA.* Atheneum, New York, 1980. First published in 1968.