Benefits of Synchronous Collaboration Support for an Application-Centered Analysis Team Working on Complex Problems: A Case Study

John M. Linebarger
Sandia National Labs
P.O. Box 5800, MS 1138
Albuquerque, NM 87185
1 505 845-8282
jmlineb@sandia.gov

Andrew J. Scholand
Sandia National Labs
P.O. Box 5800, MS 1138
Albuquerque, NM 87185
1 505 284-9110
ajschol@sandia.gov

Mark A. Ehlen
Sandia National Labs
P.O. Box 5800, MS 1138
Albuquerque, NM 87185
1 505 284-4568
maehlen@sandia.gov

Michael J. Procopio
Sandia National Labs
P.O. Box 5800, MS 1138
Albuquerque, NM 87185
1 505 845-9653
mjproco@sandia.gov

ABSTRACT
A month-long quasi-experiment was conducted using a distributed team responsible for modeling, simulation, and analysis. Six experiments of three different time durations (short, medium, and long) were performed. The primary goal was to discover if synchronous collaboration capability through a particular application improved the ability of the team to form a common mental model of the analysis problem(s) and solution(s). The results indicated that such collaboration capability did improve the formation of common mental models, both in terms of time and quality (*i.e.*, depth of understanding), and that the improvement did not vary by time duration. In addition, common mental models were generally formed by interaction around a shared graphical image, the progress of collaboration was not linear but episodic, and tasks that required drawing and conversing at the same time were difficult to do.

Categories and Subject Descriptors
H.5.3 [**Information Interfaces and Presentation**]: Group and Organizational Interfaces – *collaborative computing, computer-supported cooperative work*

General Terms
Measurement, Design, Experimentation, Human Factors.

Keywords
Benefits of collaboration, synchronous collaboration, collaboration frameworks, collaboration experiments, common mental models.

1. INTRODUCTION
That computer-mediated collaboration capability does indeed benefit group work has been the operating hypothesis of the field of Computer Supported Cooperative Work since its inception. However, not only is demonstrating the truth of this hypothesis still surprisingly tricky to do in practice, but simply answering the basic journalism questions about the benefits of collaboration is also quite slippery. Who precisely benefits from computer-mediated collaboration? What is the exact nature of the benefits? Where (in space, whether geographical or abstract) and when (in time) are the benefits realized? Why should a group choose to use computer-mediated collaboration tools? How does computer-mediation produce these benefits? And how can they be precisely measured? (See [23] and [24] for early summaries of experimental attempts to address these kinds of questions.)

GROUP'05, November 6-9, 2005, Sanibel Island, Florida, USA.
Copyright 2005 ACM 1-59593-223-2/05/0011…$5.00.

Approaches to answering these questions have varied from comparing group performance with collaboration capability to a control group without such collaboration capability, using either subjective or objective metrics (*inter alia*, [10] and [35]), to defining and capturing physiological measures of effective collaboration [6], to measuring collaboration effectiveness against a model of how collaboration occurs ([27] and [28]), either for a particular domain, or independent of domain. A variant of the last approach, experimental measurement against a domain-specific model of collaboration, is the approach taken in the work described in the present paper. This approach was chosen because it serves as an important first step in evaluating and demonstrating the value of collaboration in the context of a customer-sponsored project. The relevant collaboration domain is modeling, simulation, and analysis of the impact and interdependencies of potential critical infrastructure threats.

In the sections that follow, background to the quasi-experiment will be presented, an observed procedure for forming common mental models will be outlined, the software collaboration framework and design rationale will be described, previous work will be surveyed, the quasi-experiment itself will be detailed, and the results will be analyzed, both quantitatively and qualitatively.

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 turnaround analyses (FTAs) of 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) among all members of the analysis team.

3. COMMON ANALYSIS PICTURE

Numerous observations of FTA teams have distilled four stages in forming a common analysis picture (see Figure 1 below). The first stage is **awareness**, and consists of two levels: The identification of other members of the analysis team, and the knowledge of the specific tasks that they are currently working on. The second stage is **specialization**, in which subgroups form to carry out the overall fast analysis task. These subgroups are formed recursively, and reflect the hierarchical structure of the particular fast analysis problem. The third stage is **synchronization**, which also consists of two levels: Ensuring that each member of the subgroup is looking at the same thing (common data), and in the same way (common view of the common data). In practice, achieving this synchronization between members of a geographically distributed collaboration community can require a significant amount of time. The final stage, collaborative **interaction** proper, is only possible once synchronization has been established.

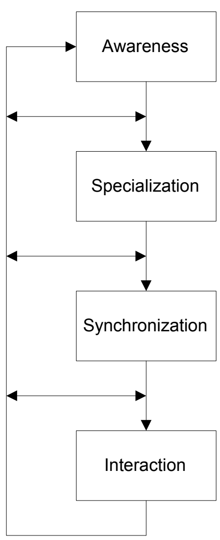

Figure 1: Stages of Forming a Common Analysis Picture

Thus a common mental model (a "common analysis picture") is formed by a flexible process of iteration through the four stages enumerated above. The number of iterations required to form a common mental model is not deterministic, and is itself an interesting research question.

4. COLLABORATION FRAMEWORK

To support such FTA teams, a software framework for synchronous collaboration has been developed. This framework, the Secure Synchronous Collaboration Framework (SSCF), addresses each of the stages of forming a common analysis picture depicted in Figure 1. 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).

The collaboration framework 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 (see [7] for an architectural description), is the first NISAC project to use the library. A screenshot of its use inside the N-ABLE™ simulation application is shown in Figure 2 below.

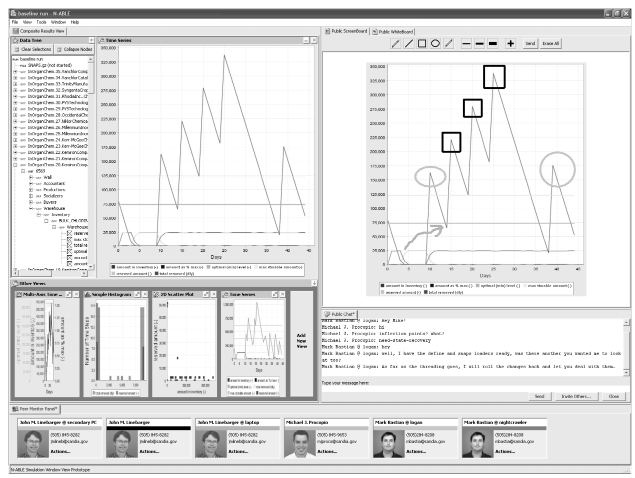

Figure 2: N-ABLE™ snapshot with collaboration enabled, showing peer awareness, group chat, and screenboard

The collaboration capabilities provided by the framework were influenced by the work presented in [29], and include:

- Pictorial awareness of other members of the virtual team that are currently using the application
- 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).

The framework was developed in the Java programming language, and uses RMI over IIOP (Remote Method Invocation over the

Internet Inter-ORB Protocol) as the distributed communication mechanism. The use of Java provides cross-platform portability—SSCF currently runs on Windows, Macintosh, and Linux computers. The framework is deployed as a set of Java packages in a single JAR (Java ARchive) file. The Java drag-and-drop API is used to drag a simulation graph or OpenGL (Open Graphics Language) image onto the screenboard panel. Each collaborator is both a client of and a server to all the other collaborators in the session, so the network topology is truly peer-to-peer. The communication functions are multithreaded, so reader-writer locks are used to protect shared data structures. An instance of the CORBA (Common Object Request Broker Architecture) Naming Service is used to keep track of all the participants in the collaborative session as well as their current subgroup structure. Subgroups can be nested to an arbitrary depth. This CORBA-based, multithreaded, peer-to-peer, subgroup-aware collaboration architecture is similar to the one pioneered in [21]. A limitation of this architecture is that all computers must be on the same network or security domain; collaboration transactions cannot currently traverse a firewall.

Two optimizations are performed to reduce network traffic. Shared screen images are compressed in JPEG (Joint Photographic Experts Group) format prior to transmission and uncompressed at the receiving end. And annotations are handled by collecting the coordinates of all mouse-button-down events in a serializable Java object which is sent as soon as the mouse button is released; the annotations are then redrawn from the coordinates by the receiver.

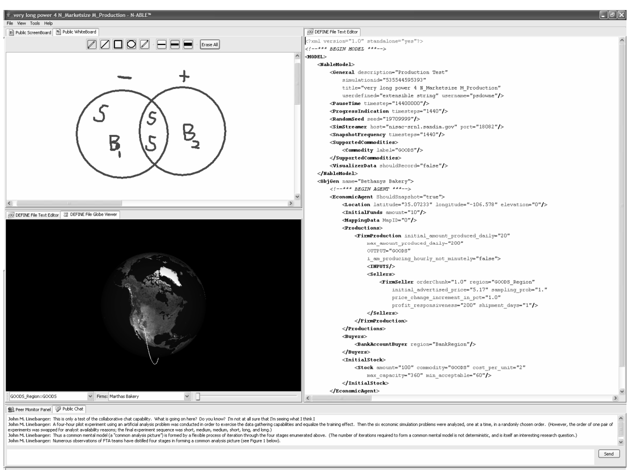

Figure 3: N-ABLE™ snapshot showing group chat and shared whiteboard with a globe visualizer and model editor

5. DESIGN RATIONALE

Two fundamental design decisions presented themselves immediately when collaboration support for FTA teams was considered. The first was whether the collaboration would be synchronous or asynchronous. Much, if not most, collaboration software in the scientific domain is asynchronous in nature; see [4] for a recent example. IBM Lotus Team Workspace (a.k.a. QuickPlace) [13] is a commercial asynchronous collaboration application with which several FTA teams were familiar. The now discontinued Habanero project [3] is a notable exception; Habanero supports synchronous collaboration. Of all the collaboration tools considered, Habanero is the closest in spirit to SSCF. Observations in several FTA exercises of the devastating impact of the failure to form a common mental model in a timely fashion led to the choice of synchronous collaboration; such collaboration was deemed the best mechanism to form common mental models within a distributed simulation analysis team, as well as between simulation analysis teams.

The second design decision was whether to use a generic collaboration tool (or suite of collaboration tools) or to use programmable collaboration capability to embed collaboration services directly into a simulation application. The first approach can be labeled "collaboration across applications" and the second approach "collaboration through an application." In the second approach, the application itself provides the context for the collaboration and forms a collaborative community. Again, much, if not most, of the synchronous collaboration that does exist in the scientific domain appears to use generic collaboration applications. On Windows computers, Microsoft's NetMeeting [15] is a popular choice. Systems like IBM Lotus Instant Messaging and Web Conferencing (a.k.a. Sametime) [13] and Habanero provide a software framework with a programmable API. So do systems for mobile device collaboration, such as YCab [1] and YCab.NET [29]. However, anecdotal evidence suggests that most users of these systems use them as standalone applications. A recent exception is an on-going research project (the "Jazz Bar") [16] which provides synchronous collaboration capability to software developers using an Eclipse plug-in. The authors call this approach by a different name, "contextual collaboration." Web-based conferencing systems such as Microsoft's LiveMeeting [15] and WebEx [14] were considered, but deemed more appropriate for collaborative creation of documents and presentations at the *end* of an FTA than collaborative analysis *during* an FTA.

The decision was made to embed collaboration services inside of a simulation application using a programmable API, for several reasons. The first was to better focus the collaboration around the tasks defined by the application and to avoid the context switch between applications necessitated by the use of standalone collaboration tools. The second was the ability to transmit application-specific objects and data structures between collaborators. The third was the ability to make the collaboration have the same look and feel as the application itself. Figure 3 shows another configuration of collaboration services in the N-ABLE™ interface, thus demonstrating the power and flexibility of bringing collaboration under the control of the application.

6. PREVIOUS WORK

Several areas of previous work apply to different aspects of this set of experiments. The concept of a shared mental model has been variously applauded ([30] and [31]) and critiqued [18], the definition differs by domain [32], and measurement is tricky [19]. However, some recent experiments (such as [22]) have demonstrated a positive connection between a shared mental model and team performance. The domain of team software development has seen much work on the impact of shared mental models (*inter alia*, [8]), where the experimental results are mixed (*e.g.,* compare [9] to [20]).

The closest empirically-based, domain-specific model of collaboration to the one presented in Figure 1 was described by

Terry Disz in [5]. The stages in the Disz model are Awareness, Interaction, Cooperation, Collaboration, and Virtual Organization. The model applies to the domain of research collaboration. In contrast, the stages of the model in Figure 1 were influenced by previous work as well as by observation of FTA teams. The first stage of the model in Figure 1, Awareness, has triggered a large body of research. In particular, the work by Saul Greenberg and Carl Gutwin at the University of Calgary, as embodied in their GroupKit toolkit, is relevant to this stage; see [12] for a representative example. The second stage, Specialization, is thoroughly discussed in [21], as is the support for fluid creation and dissolution of collaborative subgroups to reflect the hierarchical structure of the task. The third stage, Synchronization, draws on the concept of WYSIWIS ("What You See Is What I See"), which can be found in [11] and [34]. These three stages are prerequisites for the fourth stage, collaborative interaction.

7. DESCRIPTION OF EXPERIMENTS

7.1 Experiment Structure and Factor Levels

Although fast turnaround analyses generally involve groups of simulation applications, this initial quasi-experiment was designed to measure the benefits of collaboration capability for a specific application-centered analysis team, one that formed around the N-ABLE™ agent-based economic simulation tool. Since FTA 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. The six analysis problems that were investigated consisted of real N-ABLE™ economic analysis questions, not hypothetical problems for the purposes of the experiment. Stated another way, the experiment problems were not just similar to the type of work that would be performed for a real customer, they were in fact instances of real work being performed for a real customer. 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." The flow of an experiment was identical to the canonical N-ABLE™ problem analysis cycle. First, a simulation model was created on one of the client desktop machines in XML (eXtensible Markup Language) format. Next, this model was sent to a central server or cluster of servers running in parallel for execution. Finally, the results were streamed out to each of the client desktop computers for analysis by the distributed team. This process was often repeated many times during the course of each experiment. The N-ABLE™ application was used not only to perform the analysis but also, through the use of the collaboration framework described above, to perform the collaboration between the analysis team members.

A four-hour pilot experiment using an artificial analysis problem was conducted in order to exercise the data gathering capabilities and equalize the training effect. Then the six real economic simulation problems were analyzed, one at a time, in a randomly chosen order. (However, the order of one pair of experiments was swapped for analyst availability reasons; the final experiment sequence was short, medium, medium, short, long, and long.) The entire set of experiments occupied a full calendar month. Following each experiment a questionnaire was filled out by each participant, and the collaboration transaction log files that were automatically stored on each participant's machine were collected. In addition, transcripts of the group chat messages were saved and analyzed.

Because each of the participants experienced each of the three time duration factor levels, the experimental design was within-subjects (also known as repeated-measures). One-way ANOVA (ANalysis Of VAriance) was the statistical analysis method used [25]. The response variables came from the quantitative questions on the post-experiment questionnaire—which contained both Likert-scaled and short answer questions—as well as from counts derived from the automated collaboration transaction logs. Note that although the experiment was performed in a *group*, the data collected came from *individuals* in the group.

Although the SSCF 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").

7.2 Operating Hypotheses

The primary operating hypothesis was that the synchronous collaboration framework does indeed improve the ability of an application-centered collaboration community to form a common mental model of both the problem(s) and potential solutions(s). The primary benefit was hypothesized to be time, not necessarily the quality of the understanding of the problem(s) or of the solution(s) discovered. A secondary hypothesis was that the benefit of synchronous collaboration would diminish as the time duration of the analysis increased. The goal of this secondary hypothesis was to explore and identify the boundary between the synchronous and asynchronous collaboration mechanisms used by the team.

7.3 Experiment Participants

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 and enhances the tool itself. Each of these kinds of tasks 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. And it must be stressed that the software developers on the experiment team were developers of the N-ABLE™ simulation tool, not of the collaboration software used by N-ABLE™.

Each of the subjects had also participated in a long-running N-ABLE™ analysis project ("the Chlorine project") conducted the previous Fall, which did not use the built-in synchronous collaboration capabilities. Because of this, the Likert-scaled questions on the post-experiment questionnaire asked if the collaboration capabilities improved the performance of the team in a particular way relative to their experience on the Chlorine project. ("The basis of comparison is your experience on the Chlorine project, which did not use the collaboration capabilities of the new version of N-ABLE™.") In effect, the Chlorine project served as the implied control group.

Most of the analysis team was co-located in the same hall. However, 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 (but connected to the same network). The desktop computers used in this cross-platform experiment were almost evenly split between Macintosh and Windows machines.

The principal investigator was an on-line observer of each of the experiments, and sat in on the pre-experiment coordination meetings as well. The investigator also conducted a post-mortem review of the results of the experiments in order to get feedback and gain insight into the causes of the results.

7.4 Quasi-Experimental Design Rationale

Ideally, multiple analysis teams would have participated in multiple replications of each of the time durations of the experiment, using independent, artificially-constructed analysis problems. This experimental design would ensure the widest generality of the results. However, the reality of research on a complex customer-funded project mitigates against the ideal in several ways. First is the expertise required to use the N-ABLE™ software tool itself. Not only is an academic background in economic analysis necessary, but also many hours (even days) of training and familiarity with the tool. Second, the funding requirement of the customer is that all analysis efforts be directed towards real economic questions of direct value to the project. In other words, investing a month of team analysis effort on artificial problems for the purpose of an experiment would not have been approved by the customer. In practical terms, only the existing N-ABLE™ team had the expertise to work on real problems using N-ABLE™.

Quasi-experimental designs (see [2] and [33]) are ideally suited for the conditions described above—investigations in a field setting of complex, long-duration tasks requiring specialized expertise performed by members of a single group. In situations where multiple randomized experimental groups are not possible, well-constructed quasi-experiments can control internal validity such that inadequate hypotheses are properly rejected [2]. The tradeoff, of course, is a lower level of external validity than a randomized multi-group experiment. Thus the design of the experiment described in this paper can best be characterized as an exploratory case study that consists of a quasi-experiment. As a quasi-experiment it is an instance of "instrumented real work" using a single group performing complex tasks in an industrial setting, instead of a randomized experiment using multiple groups performing simple tasks in an academic setting. As an exploratory case study, we believe that it lays the foundation for future studies in addition to providing results that will guide future software development of benefit to the funding customer. We also believe the results will be generalizable at least to other distributed simulation analysis teams in the NISAC program, and quite possibly to similar analysis teams in other situations.

8. QUANTITATIVE RESULTS

As context for interpreting the results, it should be noted that the series of experiments required 72 contact hours spread out over an entire calendar month. A total of 11,477 collaboration transactions were performed, where a transaction is defined as the transmission of a multimedia artifact (text, graphic, or generic object) to every other member of the distributed team. Based on the results of the post-experiment questionnaire, the primary operating hypothesis—that synchronous collaboration capability would improve the formation of a common mental model of the problem(s) and solution(s)—was consistently supported. As Table 1 below indicates, in all cases the average response was between "Agree" and "Strongly Agree." Contrary to expectations, with collaboration the quality of the common mental model of the problem improved more than the time it took to form the common mental model. This finding is consistent with the results of the research reported in [24].

Table 1. Results from Questionnaire about Benefits of Collaboration in Forming Common Mental Models (Scale is 1 [Strongly Disagree] to 5 [Strongly Agree])

	Improved Overall with Collaboration		**Improved Time with Collaboration**		**Improved Quality with Collaboration**	
	Mean	**Std Dev**	**Mean**	**Std Dev**	**Mean**	**Std Dev**
Common Mental Model of Problem	4.45	0.71	4.29	0.77	4.48	0.62
Common Mental Model of Solution	4.23	0.67	4.13	0.83	4.13	0.81

Note that the average responses to the second suite of questions centering on the common mental model of the solution(s) were somewhat lower than the responses centered on the common mental model of the problem(s). This may point to a limitation in the collaboration capabilities provided. However, it may also point to the particular analysis tasks chosen for each time duration of the experiment. Several questionnaires contained comments to the effect that "We didn't have enough time to actually get to the solution" of the analysis task, and "N/A" (Not Applicable) was chosen more often for this second set of questions than for any other set.

There was some evidence that the positive impression of the collaboration capabilities grew over time; stated another way, the collaboration capabilities "wore well" with the analysis team as they got accustomed to using them to solve problems as a group. One example is that humor sprang up as the experiments progressed. Another is that the average response to the questions about the *overall* contribution of collaboration to forming a common mental model of the solution, and about the contribution of collaboration to the *time* it took to form a common mental model of the solution, was monotonically increasing over the course of the set of experiments. This is more than just a training effect; the entire team had used the collaboration capabilities well in advance of the pilot experiment. Instead, it was more of an "adoption" or "integration" effect as the team wove the collaboration capabilities of the software into their group problem solving practice.

Similar positive results were obtained from questions that probed how well the synchronous collaboration capability supported the stages of forming a common analysis picture presented in Figure 1. The average response to each question was between "Agree" and "Strongly Agree." Table 2 below summarizes the relevant responses to the questionnaire.

Table 2. Results from Questionnaire about Support for Stages of Forming a Common Analysis Picture (Scale is 1 [Strongly Disagree] to 5 [Strongly Agree])

Stage	Mean	Std. Dev.
Awareness of Team	4.65	0.48
Awareness of Task	4.23	0.76
Synchronization of Data	4.4	0.76
Synchronization of View of Data	4.32	0.69

However, the secondary operating hypothesis—that the benefits of collaboration would vary by time duration of the analysis task—was not supported by the ANOVA results. The p-values of all but one of the response variables were above the significance threshold of 0.05. However, one response variable—agreement or disagreement that the common mental model of the problem improved overall with collaboration—had a p-value of just under 0.05 ($F_{2,27}$ = 3.38, p = 0.0491). Further analysis led to the conclusion that this response variable was probably not significant either. Examination of the data using a box-and-whisker plot revealed the presence of outliers, which may have influenced the ANOVA. Running a Kruskal-Wallis test against the data instead, which utilizes the median instead of the mean and is thus less sensitive to outliers, yielded a p-value of 0.053. This value is just outside the range of statistical significance at the 95% confidence level.

So why might duration not have been a significant factor? One reason might have been due to the small number of replications of each time duration. But another might have been due to the "adoption" effect mentioned earlier, coupled with the fact that the long duration experiments were conducted last.

One of the goals of the experiment was to explore the boundary line between synchronous and asynchronous collaboration, and to capture the mechanisms of collaboration that were used outside of the collaboration features provided by the software framework. Toward that end, the post-experiment questionnaire asked four questions: The percentage of collaboration on the analysis task done synchronously as opposed to asynchronously; the percentage of synchronous collaboration done in (or through) the N-ABLE™ application as opposed to outside (or around) N-ABLE™; the asynchronous collaboration mechanisms used; and the non-N-ABLE™ synchronous collaboration mechanisms used. Based on the responses there was reason to believe that the two percentage-of-collaboration questions were variously interpreted, so the validity of the responses is questionable. Nonetheless, the mean percentage of synchronous collaboration for each of the experiments was 97%. Contrary to expectations, that percentage did not decline as the time duration of the experiments increased. The percent of synchronous collaboration performed inside the application (instead of outside the application) was 92%. This percentage was lower for long duration experiments, as expected, but it was higher for medium duration experiments. E-mail was by far (75%) the most common form of asynchronous collaboration; it was primarily used to get around the current limitations of the collaboration framework, such as the inability to display two shared images side-by-side. The most frequent methods of synchronous collaboration outside of the application were face-to-face conversation (84%) and phone calls (16%). Face-to-face interaction was used to avoid the limitations of the software collaboration framework for certain kinds of collaboration (see the discussion in the Qualitative Results section below).

The questionnaire also asked which collaboration capabilities were most and least useful. The most useful collaboration capability listed was chat (52%), followed by screenboard (36%) and whiteboard (12%). However, the usefulness of the whiteboard appeared to be dependent on the particular analysis task being performed—it was also deemed the least useful capability 71% of the time. File transfer was noted as least useful in 19% of the responses.

Finally, raw collaboration transactions from the automated collaboration transaction log were counted and aggregated. Figure 4 displays the results as rounded percentages. Public chat dominated the collaboration transactions, followed by annotations, private chat, and screen images.

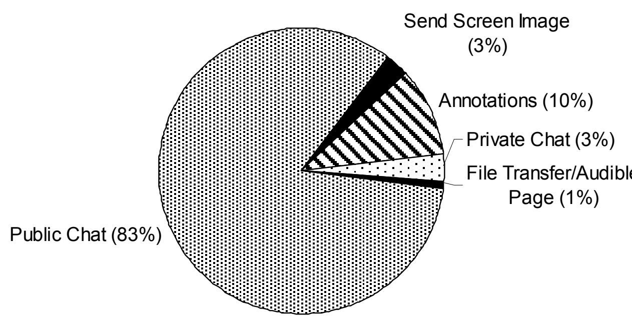

Figure 4: Measured Collaboration Transaction Percents

Table 3. Measured Collaboration Transaction Percents by Collaboration Media Type

Collaboration Media Type	Percent of Transactions
Text	86.5
Graphics	12.75
Generic Objects	0.75

Another presentation of this data is possible. Aggregating screen images and annotations into the category of graphical collaboration, and noting that both file transfers and audible pages were implemented by the transmission of generic (but serializable) Java objects, Table 3 above presents collaboration transaction percents by media type.

====== 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
Deb: so intransit is part of max storable amt and amt in inventory?
Deb: no
Deb: go ahead
====== TRANSMISSION OF SCREEN IMAGE ======
Sue: this graph is the frequency of calls he makes to find butter
[RECOGNITION OF PROBLEM(S)]
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
[COMMON MENTAL MODEL OF PROBLEM(S)]
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
[COMMON MENTAL MODEL OF SOLUTION(S)]
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
Deb: what would that cost?
Andy: Sue: I'm sorry I stepped on your comment to Deb about graph of 4
Deb: if that were a mitigation policy
Andy: Deb: in the real world?
Andy: or for us to sim
Sue: Andy: no worries
Andy: easy to simulate
[REFINEMENT OF COMMON MODEL OF SOLUTION(S)]
Deb: now that we have a potential policy suggestion, can we determine the cost of that policy?
Deb: in simulation
Andy: real world cost, you would know better than I, having spoken with truck firms
Sue: we just need to turn on the pricing component of transportation to model I think...
Andy: one way to think of pipelines (in chlorine) is they remove the cost of discrete quantization
Deb: yes that is what I was inferring, Sue
Andy: We may need a second market
Andy: For example, for concrete, you can have a truck come and pour ready mix
Andy: or you can buy a package of dry mix at Lowes yourself
Andy: those are 2 different attempts to serve the same market
Sue: in chlorine we solved this quandary by having bulk and bottled delivery which were separate markets
Sue: I should say modeled as separate markets
Andy: Sue: exactly
Sue: hmm this gives me lots to think about...
Sue: and it is three
Verne: I'm thrashed
Andy: it seems there should be a second tier of distributors serving supermarkets, maybe chains like Smiths, Krogers, Safeway...

Figure 5: Two Contiguous Chapters from Transcript Showing Collaborative Creation of Common Model of Problem and Solution

9. QUALITATIVE RESULTS

Several qualitative results also came out of the quasi-experiment. Perhaps the most important is that group insight (*i.e.,* formation of a 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. The critical importance of shared images to the collaboration is the best explanation for the apparent disparity in the quantitative data above—the screenboard was listed as the most useful collaboration capability 36% of the time, yet it was actually used for less than 13% of all collaboration transactions. Closely related is the observation that the 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 usually the transmission of a shared screen image, around which subsequent collaboration coalesced. Sometimes several screen images, transferred files, and chat transcripts formed a "conversation package," a collection of related collaborative interactions. And sometimes the series of collaboration chapters or episodes 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. The implications of these observations for the design of synchronous collaboration software will be explored in a subsequent paper.

An illustration of the observations above is provided in Figure 5, which consists of two contiguous chapters from the chat transcript from one of the experiments. Each chapter begins with the transmission of a screen image, and subsequent interaction is based on that shared screen image. The second chapter is annotated by bracketed tags that delineate the points at which collaboration around a shared screen image triggered the recognition of a problem by a particular individual, then the creation of a common mental model within the group of the problem, followed by a common mental model of a solution, and finally a refinement of the solution.

However, not all chapters were bounded by the transmission of a shared screen image. Some floated free of the analysis context, and were often triggered by other stimuli in the environment. "Humor" chapters occasionally occurred, especially toward the end of the experiments as the use of the collaboration capabilities became second nature and functioned as just another mode of expression. But almost all of the collaborative interaction was work-related, consistent with the results reported in [17].

Another qualitative result relates to the type of work that was appropriate for the collaborative environment. There were two types of complex collaboration tasks for which the synchronous collaboration capabilities provided by the framework proved inadequate—the characterization of the problem space of the analysis, and the design of the simulation software itself. This inadequacy was recognized during the pilot experiment, and all subsequent experiments began with a face-to-face meeting (geographically expanded via teleconference when necessary) in a room with a whiteboard, in order to characterize the problem space of the upcoming experiment and to perform an initial division of labor. The need to perform these kinds of complex tasks was one of the main causes of switching from the computer-based collaboration tools to other forms of synchronous collaboration, which is consistent with the results reported in [26] but not with the results of the follow-on study in [17]. The primary reason expressed for why the software framework was inadequate for these types of tasks was that it serialized textual and graphical communication instead of allowing both to be done simultaneously by the same person. For example, it was physically impossible for the same person to type a chat message and to annotate a shared screen image at the same time. During the face-to-face meetings the use of a whiteboard allowed simultaneous conversation and annotation, which was crucial for the rich interaction required for problem space characterization and software design. Conducting a phone teleconference by speakerphone while using the collaboration tool, or supporting voice-over-IP (Internet Protocol) through the collaboration framework itself, may resolve this inadequacy.

Like other studies (see [24] for a summary), participants reported a more egalitarian consideration of individual contributions. The geographic separation fostered independent thinking and mitigated the influence of dominant personalities. One team member commented that the integrated collaboration capability created an entirely new communication pattern for the team, especially because it involved text, images, and drawings.

A final observation illustrates a possible benefit of programmable collaboration capabilities, which allow collaboration services to be deployed *through* a particular application instead of by generic collaboration tools that work *across* applications (and are themselves applications). Embedding collaboration inside the application itself appeared to impart a task focus and people awareness to the communication, such that the collaboration generally remained on-point. This was contrary to the previous experience of the software developers on the analysis team; they had installed a collaboration system from Groove Networks to support the development of the N-ABLE™ simulation tool, but were forced to discontinue its use because of the flame wars over technical issues that kept breaking out. One theory was that the lack of an explicit task focus in a generic collaboration tool such as Groove fostered the unbounded creation of massive messages that quickly escalated into flames. Such flame wars were simply not encountered during these experiments.

10. FUTURE WORK

To improve the generality of the results, and to better justify the benefits of the synchronous collaboration framework, a follow-on experiment that includes multiple analysis teams and a larger number of replications should be conducted. These teams would either use the collaboration-enabled N-ABLE™ simulation tool, or (more realistically) would use the SSCF framework in the context of their own particular simulation tool. Ideally, a further experiment should be designed that moves away from the realm of subjective perception into more objective measures of the benefits of collaboration. One suggestion is to measure the ability of an economic analysis team to solve a randomly generated problem that has a closed form solution (such as finding the Nash equilibrium), both with and without collaboration capability.

With regard to the capabilities of the collaboration framework itself, several items of future work were suggested in the answers to the last question of the post-experiment questionnaire and in the post-mortem meeting with the simulation analysis team. A basic request was to change the architecture of the framework to allow synchronous collaboration between security domains instead of just within a single network security domain (*i.e.*, collaboration across firewalls). Another was to explore the use of voice-over-IP through the collaboration framework to allow someone to converse and make graphical gestures at the same time. This could allow the collaboration framework to be used during problem space characterization and software design, two forms of collaboration where the analysis team had to resort to face-to-face meetings because of the rich interaction required. A third request was to drive the collaboration by a publicly-modifiable agenda, a hierarchical task list of the progression of the analysis to which time limits could be attached. Perhaps the task list could also function as an awareness mechanism of which people were currently working on which tasks, and of the number of tasks remaining until the completion of the analysis. Closely related was the request for support of a moderator capability, a "director" in cinematic terms, who could monitor the progress of the collaboration relative to the task list, move the team to the next item on the agenda, and close the chapters or episodes in the collaboration (*i.e.,* "wrap a scene"). A fifth request was the ability to vote (à la Habanero), which could be used during problem and solution determination, and in moving from one agenda item to another. Sixth was the ability to display multiple shared graphical images side-by-side, at the same time, for comparison purposes. A seventh was to make the file transfer capability more robust, improve its performance with large files, and add a progress bar. Eighth was to provide a rich set of awareness glyphs on the peer awareness panel, which could indicate the status of a particular team member (*i.e.,* away from keyboard, on the phone, etc.) and perhaps even the particular task they are working on. This feature is similar to what the Eclipse "Jazz Bar" [16] provides in a team software development environment. And a final request was to timestamp and add threading capability to chat conversations.

11. CONCLUSION

The perception of the N-ABLE™ modeling, simulation, and analysis team was that the use of the built-in synchronous collaboration framework improved the ability of their team to form common mental models of both problem(s) and solution(s), compared to a previous project in which synchronous collaboration capability was not available. The collaboration capabilities improved not only the time it took to form a common mental model, but also the quality (*i.e.,* depth of understanding) of the common mental model. The team also agreed that the collaboration framework supported key stages in a model of collaboration appropriate to the domain of rapid simulation and analysis of critical infrastructure threats (depicted in Figure 1). This model was based on the empirical observation of several fast turnaround analysis exercises. However, the operating hypothesis that the benefits of synchronous collaboration capability would decline—or even vary—by the time duration of the analysis was not supported by the data from this quasi-experiment. In addition, three qualitative observations were made: group insight (*i.e.,* formation of a common mental model) often occurred while the group was discussing and annotating a shared screen image; the collaboration generally did not proceed linearly but episodically, in chunks or chapters, with the transmission of a shared screen image forming the line of demarcation between episodes; and two types of collaboration—characterization of the analysis problem space and the design of the simulation software itself—highlighted an inadequacy of the collaboration framework because of the inability to gesture and converse at the same time.

12. 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.

13. REFERENCES

[1] Buszko, D., W. Lee, & A. Helal. (2001) "Decentralized Ad-Hoc Groupware API and Framework for Mobile Collaboration." *Proceedings of the 2001 International ACM SIGGROUP Conference on Supporting Group Work (Group '01)*, 30 Sept. – 3 October, Boulder, CO, USA, 5-14.

[2] Campbell, Donald T., & Julian C. Stanley. (1966). *Experimental and Quasi-Experimental Designs for Research.* Boston: Houghton Mifflin Company.

[3] Chabert, Annie, Ed Grossman, Larry Jackson, Stephen Pietrowicz, & Chris Seguin. (1998). "Java Object-Sharing in Habanero." *Communications of the ACM* 41(6), 69-76.

[4] Chin, George. (2004). "Capturing and Supporting Contexts for Scientific Data Sharing via the Biological Sciences Collaboratory." *Proceedings of the 2004 ACM Conference on Computer Supported Cooperative Work (CSCW 2004)*, 6-10 November, Chicago, IL, USA, 409-418.

[5] Disz, Terry. (2001). "The Access Grid Collaboration Environment." *Syllabus* (April 2001). Available at http://www.campus-technology.com/article.asp?id=3238.

[6] Doser, Adele B., Peter B. Merkle, Curtis Johnson, Wendell Jones, David Warner, & Tim Murphy. (2003). "Enabling Technology for Human Collaboration." Sandia National Laboratories Technical Report SAND 2003-4225. Available at http://www.bodymedia.com/pdf/sandia-4225.pdf.

[7] Eidson, Eric D., & Mark A. Ehlen. (2005). "NISAC Agent-Based Laboratory for Economics (N-ABLE™): Overview of Agent and Simulation Architectures." Sandia National Laboratories Technical Report SAND2005-0263.

[8] Espinosa, J. Alberto, Robert E. Kraut, Sandra A. Slaughter, Javier F. Lerch, James D. Herbsleb, & Audris Mockus. (2001). "Shared Mental Models and Coordination in Large-Scale, Distributed Software Development." *Proceedings of the International Conference on Information Systems (ICIS 2001)*, 16-19 December, New Orleans, LA, USA, 513-518.

[9] Espinosa, J. Alberto, Robert E. Kraut, Sandra A. Slaughter, Javier F. Lerch, James D. Herbsleb, & Audris Mockus. (2002). "Shared Mental Models, Familiarity, and Coordination: A Multi-Method Study of Distributed Software Teams." *Proceedings of the Twenty-Third International Conference on Information Systems (ICIS 2002),* 15-18 December, Barcelona, Spain, 425-433.

[10] Furmanski, Chris, David Payton, & Mike Daily. (2004). "Quantitative Evaluation Methodology for Dynamic, Web-based Collaboration Tools." *Proceedings of the 27th Hawaii International Conference on System Sciences*, 5-8 January, Big Island, HI, USA, 50132a.

[11] Greenberg, Saul, Carl Gutwin, & Andy Cockburn. (1996). "Awareness Through Fisheye Views in Relaxed-WYSIWIS Groupware." *Proceedings of the Conference on Graphics Interface,* 22-24 May, Toronto, ON, Canada, 28-38.

[12] Gutwin, Carl, Saul Greenberg, & Mark Roseman. (1996). "Workspace Awareness in Real-Time Distributed Groupware: Framework, Widgets, and Evaluation." *People and Computers XI: Proceedings of HCI '96.* London: Springer, 281-298.

[13] http://www.lotus.com/products/product3.nsf/wdocs/ homepage
http://www.lotus.com/products/product3.nsf/wdocs/ltwhome

[14] http://www.webex.com/

[15] http://www.microsoft.com/windows/netmeeting/

http://www.microsoft.com/office/livemeeting/prodinfo/default.mspx

[16] Hupfer, Suzanne, Li-Te Cheng, Steven Ross, & John Patterson. (2004). "Introducing Collaboration into an Application Development Environment." *Proceedings of the 2004 ACM Conference on Computer Supported Cooperative Work (CSCW 2004)*, 6-10 November, Chicago, IL, USA, 21-24.

[17] Isaacs, Ellen, Alan Walendowski, Steve Whittaker, Diane J. Schiano, & Candace Kamm. (2002). "The Character, Functions, and Styles of Instant Messaging in the Workplace." *Proceedings of the 2002 ACM Conference on Computer Supported Cooperative Work (CSCW'02)*, 16-20 November, New Orleans, LA, USA, 11-20.

[18] Klimoski, Richard, & Susan Mohammed. (1994). "Team Mental Model: Construct or Metaphor?" *Journal of Management* 20(2), 403-437.

[19] Langan-Fox, Janice, Sharon Code, & Kim Langfield-Smith. (2000). "Team Mental Models: Techniques, Methods, and Analytic Approaches." *Human Factors* 42(2), 242-271.

[20] Levesque, Laurie L., & Jeanne M. Wilson. (2001). "Cognitive divergence and shared mental models in software development project teams." *Journal of Organizational Behavior* 22(2), 135-144.

[21] Linebarger, John M., Christopher D. Janneck, & G. Drew Kessler. (2005). "'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).

[22] Mathieu, John E., Gerald F. Goodwin, Tonia S. Heffner, Eduardo Salas, & Janis A. Cannon-Bowers. (2000). "The Influence of Shared Mental Models on Team Process and Performance." *Journal of Applied Psychology* 85(2), 273-283.

[23] McGrath, Joseph E., & Andrea B. Hollingshead. (1994). *Groups Interacting With Technology: Ideas, Evidence, Issues, and an Agenda.* Sage Library of Social Research 194. Thousand Oaks, CA: SAGE Publications.

[24] McLeod, Poppy Lauretta. (1992). "An Assessment of the Experimental Literature on the Electronic Support of Group Work: Results of a Meta-Analysis." *Human-Computer Interaction* 7(3), 257-280.

[25] Montgomery, Douglas C. (2001). *Design and Analysis of Experiments*. 5th Edition. New York: John Wiley & Sons.

[26] Nardi, Bonnie A., Steve Whittaker, & Erin Bradner. (2000). "Interaction and Outeraction: Instant Messaging in Action." *Proceedings of the 2000 ACM Conference on Computer Supported Cooperative Work (CSCW'00)*, 2-6 December, Philadelphia, PA, USA, 79-88.

[27] Noble, David, & Michael Letsky. (2001). "Cognitive-Based Metrics to Evaluate Collaboration Effectiveness." Available at http://www.onr.navy.mil/sci_tech/personnel/342/342_ckm/pubs/docs/33_mgc_ebr/cognitive_based_metrics.doc.

[28] Noble, David, Diana Buck, & Jim Yeargain. (2001). "Metrics for Evaluation of Cognitive-Based Collaboration Tools." *Proceedings of the 6th International Command and Control Research and Technology Symposium*, 19-21 June, Annapolis, MD, USA. Available at http://www.dodccrp.org/events/2001/6th_ICCRTS/Cd/Tracks/Papers/Track6/106_tr6.pdf.

[29] Procopio, Michael J. (2002). "YCab.NET: Decentralized Collaboration Groupware for Mobile Devices using the Microsoft .NET Framework." Master's of Science Thesis, University of Florida. Available at http://etd.fcla.edu/UF/UFE1000155/procopio_m.pdf

[30] Rentsch, J. R., & R. J. Hall. (1994). "Members of Great Teams Think Alike: A Model of Team Effectiveness and Schema Similarity Among Team Members." In M. M. Beyerlein & D. A. Johnson, eds., *Advances in Interdisciplinary Studies of Work Teams: Theories of Self-Managing Work Teams*, Vol. 1 (Greenwich, CT, USA: JAI Press), 223-261.

[31] Rentsch, Joan R., & Richard J. Klimoski. (2001). "Why do 'great minds' think alike?: antecedents of team member schema agreement." *Journal of Organizational Behavior* 22(2), 107-120.

[32] Rouse, William B., & Nancy M. Morris. (1986). "On Looking Into the Black Box: Prospects and Limits in the Search for Mental Models." *Psychological Bulletin* 100(3), 349-363.

[33] Shadish, William R., Thomas D. Cook, & Donald T. Campbell. (2002). *Experimental and Quasi-Experimental Designs for Generalized Causal Inference.* Boston: Houghton Mifflin Company.

[34] Stefik, M., D. Bobrow, S. Lanning, D. Tatar, & G. Foster. (1987). "WYSIWIS Revised: Experiences with Multi-user Interfaces." *ACM Transactions on Office Information Systems* 5(2), 147-167.

[35] Wilson, J. D., N. Hoskin, & J. T. Nosek. (1993). "The Benefits of Collaboration for Student Programmers." *Proceedings of the Twenty-fourth SIGCSE Technical Symposium on Computer Science Education*, 28-19 February, Indianapolis, IN, USA, 160-164.