Position Paper for Cognition and Collaboration Workshop: Analyzing Distributed Community Practices for Design

Jean Scholtz, Michelle Steves, and Emile Morse
National Institute of Standards and Technology
100 Bureau Drive, MS 8940
Gaithersburg, MD 20899
Jean.scholtz@nist.gov, msteves@nist.gov, emile.morse@nist.gov
+1 301 975 2520, +1 301 975 3537, +1 301 975 8239

Evaluation of Collaboration Support for the Intelligence Community

In the United State, the events of 9/11 have triggered analysis of the intelligence community. Among the recommendations from the National Commission on Terrorist Attacks Upon the United States are the recommendations on unity of effort on the challenge of counterterrorism, including:

> "unifying the many participants in the counterterrorism effort and their knowledge in a network-based information sharing system that transcends traditional government boundaries" [1]

Although the work presented in this paper, has been carried out specifically in the United States, there are almost certain to be identical requirements in other countries. In the paper, we focus primarily on US aspects and leave to international readers to draw any parallels in their home countries.

The problem

US Government agencies today face a number of tensions. First, each agency needs to carryout their particular mission, while interacting appropriately with other agencies. An agency may rely on other agencies for input and may produce output for other agencies. They need to collaborate to ensure a high quality of work, but they need to retain their identity and mission focus. To that extent, the various agencies within the United States that produce intelligence focus on different aspects of counter terrorism, ranging from different types of data to different types of questions for different purposes. Today we can collect much more data than can be analyzed. Even sorting massive amounts of data to locate "relevant" data is a huge task. To make the task even more difficult, some of the information is meant to be misleading and deceptive. To work more effectively, we need internal collaboration (within agencies) and external collaboration (between agencies). Collaborating with people who we know is difficult enough. Locating and collaborating with people we don't know is extraordinarily difficult.

During the past several years, we have been working as consultants advising on methodologies and metrics for evaluation of tools for the analytic community. One area that we have been evaluating is the use of collaborative tools and the impact they have on various processes. During the course of this work, we have installed collaboration tools on an experimental network available to a number of agencies. This work is both exciting and challenging as we have been given the opportunity to evaluate collaborative software use in complex real-world environments. With this, however, come a number of restrictions. We have used both qualitative and quantitative data to assess the use of the collaboration software and the impact on the various processes where it has been used. There are several issues: first, we are trying to evaluate this software in an operational environment. There are security restrictions on what we can collect and what we can report. There are constraints on analyst time. At times we are able to collect baseline data about processes and at other times, the software is being used in a new process for which no baseline exists. We are opportunistic about the experiments that we conduct, and that sometimes implies that there is no opportunity to collect baseline information.

We, however, need to be able to draw some conclusions about collaboration software based on experiments with different users, for different purposes, and at different times. Currently, we have only one

[1] Page 21, Executive Summary, The 9/11 Commission Report, National Commission of Terrorist Attacks Upon the United States, 2004, http://www.9-11commission.gov/

collaboration tool available, but we should be installing another soon. How can we compare collaboration tools used in different agencies by different people? To address these questions, we began by developing a metrics model.

The Metrics Model

Our model consists of five elements or levels [3,4]. The levels are: system goals, evaluation objectives, conceptual metrics, and measures, both conceptual and implementation-specific. Each successive element in the model is a refinement in scope. This refinement can also be thought of as a level of abstraction or sphere of concern. For example, a system goal may have one or more evaluation goals. Each evaluation goal will have one or more conceptual metrics, which when assessed with its associated measures, will contribute to an assessment of if, and potentially how well, a particular system goal was met.

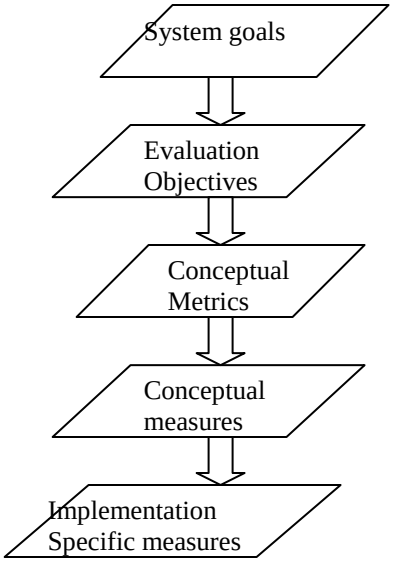
The ***system goal*** is the intended benefit or functionality the software system will provide. This goal may reflect a technical performance goal for the software system or an organizational goal, e.g., process or product improvement. The system goal element provides two critical ingredients of system evaluation design: determining which aspect(s) of the system are of primary importance in the work domain and providing high-level questions the evaluation must answer – whether the stated goals were met.

The next level is the ***evaluation objective*** level. Each objective drives a particular part of an evaluation, within the context its associated system goal. Evaluation objectives (EO) partition evaluation concerns in the overall assessment of a particular goal. Clear articulation of these objectives provides further refinement in the 'top-down' design of the evaluation.

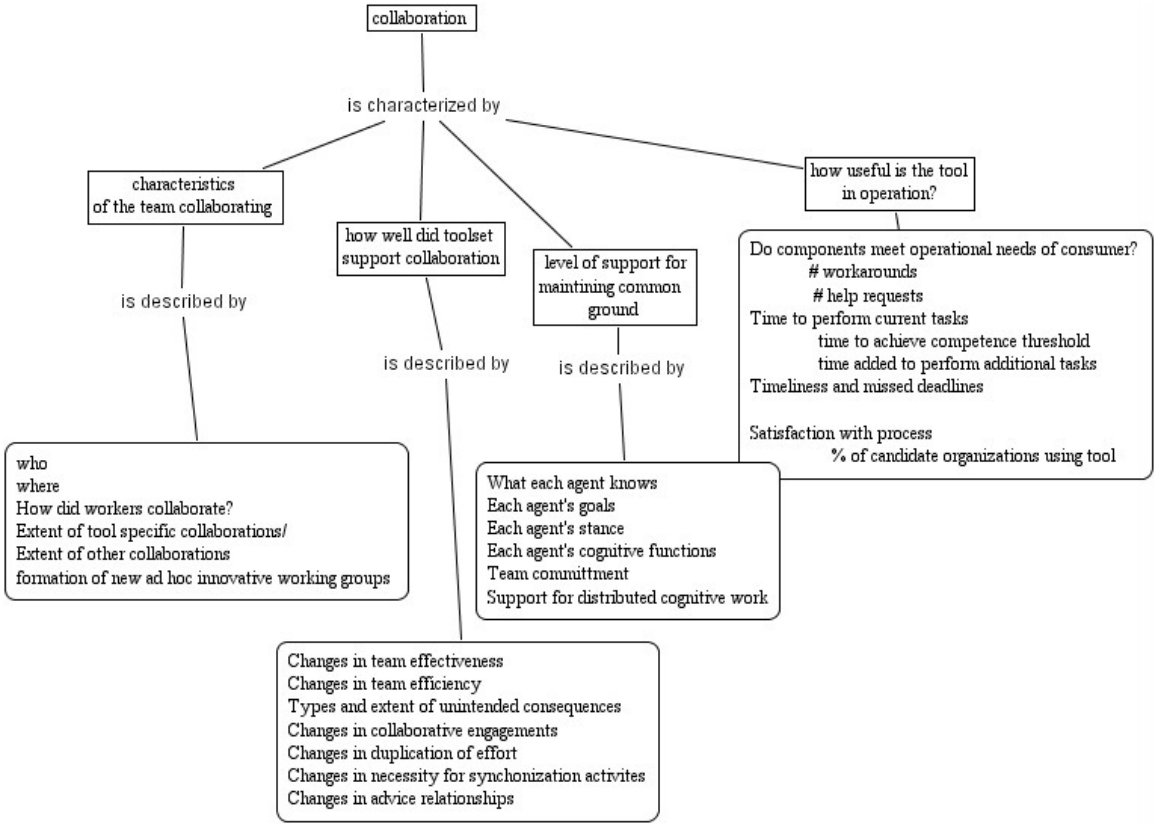
The lowest three levels in the model constitute ***metrics*** and ***measures***. Often these terms are used inter changeably, however, we define the terms in relation to each other and describe how to distinguish them. We define *metric* as the "interpretation of one or more contributing elements, e.g., measures or other metrics, corresponding to the degree to which the set of attribute elements affects its quality." Interpretation can be a computation or human assessment of the contributing elements. The computation can be the summed, weighted values, of the contributing elements. When weights are used, they are assigned to elements according to their importance to the respective attribute with respect to the particular assessment scenario. We define a *measure,* a noun, as a "performance indicator that can be observed singly or collectively, computed, calculated, and may, at times, be automatically collected. A simple way to distinguish between metrics and measures is by the following statement: a measure is an observable value, while a metric associates meaning to that value by applying human judgment, often through a formula based on weighted values using the contributing measures."

The lowest two levels in the model constitute 'measures.' These levels represent measures that are required to substantiate the formulation of answers to evaluation questions. There are two levels of abstraction for measures: conceptual and implementation-specific. Conceptual measures identify the type of data to be collected. Implementation-specific measures identify the specifics about a particular collection instance, e.g., data element(s), associated tool, collection method, and so on.

The figure below represents the metrics model



A portion of the metrics model for evaluating collaboration is shown below.



For each of these conceptual metrics that is appropriate for a given experiment, we specify implementation specific ways of obtaining this. The model is also dynamic. As we work on new processes and new software tools, we will certainly uncover new measures to collect.

Case Study One

We are currently analyzing the use of Groove™ software used in a process for the past 1 ½ years. While we will not be able to report the results at this workshop due to the nature of the data, we will be able to discuss the success of the evaluation. To analyze the use of the collaboration software we are currently collecting and analyzing:

- collaboration software logs
- questionnaires
- spreadsheets containing process data

The collaboration software logs will let us examine:

The overall use of the software
The different components used in the software
Usage/ individuals involved in the process
Usage/ roles of individuals involved in the process

The spreadsheets of process data contain dates that input was requested, dates that products were produced, people formally assigned to different requests. This data will give us

Time to perform tasks
Efficiency over time
Effectiveness over time as reflected in number of requests

Questionnaires will help us determine usage and satisfaction by analysts, supervisors, and consumers of analytic products. They will also help us determine the characteristics of the collaboration groups and provide information about user satisfaction and product quality and timeliness.

By combining the software logs with information from the spreadsheets, we hope to see collaborative activities in supporting different requests. There are, however, caveats. Because we are looking at real world data, we must consider what is happening in the real world at the various times frames we are examining. External situations may contribute to increases or decreases in requests, and we must plot this type of information against our trends in order to properly interpret our data.

Case Study Two

We have been analyzing data from four open source analysts, each working on different tasks, for the past year. Prior work [2] has been conducted on two senior open source analysts each doing a single monitoring task over a one year period. We have collected data both through the user of Glass Box software [1] and by observations. While there was some collaboration happening in these activities, it was not a main focus. However, in the experiments the first part of the year, we will be requiring the analysts to use collaborative software and perform analytic tasks that require collaboration. We will be able to collect both Glass Box data and software logs from the collaboration software. We have a number of measures of process including:

- Time spent in various phases of analysis
- Growth of report documents over time
- Number of relevant documents/ task
- Comparisons of above based on length of task/ type of report requested
- Time spent using various software applications
- Ratings of task complexity

We will be able to compare these measures for a non-collaborative analysis with those for collaborative work. This will enable us to determine changes in process. Additionally, we collect the products that each analyst generates. As we have access to a number of senior analysts, we can also evaluate the quality of the products. While the actual logged data will most likely not be analyzed by the time of the CHI workshop, we will have a number of observational studies that we can report on for the workshop.

Summary
We are conducting evaluations of collaborative software for intelligence analysts in operational environments. Because it is the real world, we are able to see how users want to use the software and the different types of processes that need collaboration support. We have been able to provide feedback to developers concerning the usability and utility of their products.

We collect both qualitative and quantitative data. The intelligence analysts have extremely limited time so we use the quantitative data to refine our qualitative data collection. It is also essential that we understand the various processes used in the different agencies so we can focus on collecting information about impacts on those processes due to collaborative software tools.

Because it is the real world, there are limitations to the data we can collect – due to security, access, and time limitations. As we do experiments involving different agencies and different software solutions, we developed a metrics model to attempt to bring some systematic data collection to a sometimes chaotic process. We are currently using this metrics model to analyze 1 ½ years of a collaborative process. By the time of the workshop, we will be able to report on the success of the evaluation methodology and recommendations. Due to the nature of the work, we will only be able to report general results from the actual analysis of the collaboration tool. One of the methodological questions we are exploring is the utility of the metrics model for different stages of software development and deployment.

References
1. Cowley, P., Nowell, L., and Scholtz, J. 2005. Glass Box: An Instrumented Infrastructure for Supporting Human Interaction with Information. HICSS 38, Jan. 3-6. Hawaii.
2. Scholtz, J., Morse, E., and Hewett, T. 2004. In Depth Observational Studies of Professional Intelligence Analysts. Presented at Human Performance, Situation Awareness, and Automation Conference, March 22-25. Daytona Beach, Fl.
3. Steves, M. and Scholtz, J. 2004. A Framework for Real-World Software System Evaluations. CSCW.
4. Steves, M. and Scholtz, J. 2005. A Framework for Evaluating Collaborative Systems in the Real World. HICSS 38, Jan. 3-6. Hawaii.