CCDC DAC-TR-2020-034
May 2020

Behavioral Observations Logging Toolkit (BOLT): Initial Deployed Prototypes and Usability Evaluations

by Christopher J. Garneau, Blaine E. Hoffman, and Norbou E. Buchler

DISCLAIMER

The findings in this report are not to be construed as an official Department of the Army position unless so specified by other official documentation.

WARNING

Information and data contained in this document are based on the input available at the time of preparation.

TRADE NAMES

The use of trade names in this report does not constitute an official endorsement or approval of the use of such commercial hardware or software. The report may not be cited for purposes of advertisement.

CCDC DAC-TR-2020-034
May 2020

Behavioral Observations Logging Toolkit (BOLT): Initial Deployed Prototypes and Usability Evaluations

by Christopher J. Garneau, Blaine E. Hoffman, and Norbou E. Buchler
CCDC Data & Analysis Center

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED *(From - To)*
May 2020	Technical Report	1 May 2019–1 May 2020

4. TITLE AND SUBTITLE
Behavioral Observations Logging Toolkit (BOLT): Initial Deployed Prototypes and Usability Evaluations

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Christopher J. Garneau, Blaine E. Hoffman, and Norbou E. Buchler*

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Director
U.S. Army CCDC Data & Analysis Center
6896 Mauchly Street
Aberdeen Proving Ground, MD 21005-5071

8. PERFORMING ORGANIZATION REPORT NUMBER
CCDC DAC-TR-2020-034

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT
DISTRIBUTION STATEMENT A. Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES
*correspondence author

14. ABSTRACT
There is a need for valid tools and methods to support the depth/breadth of cyber assessments needed to facilitate and inform 1) training assessments, 2) advanced technology selection, 3) Human Systems Integration, and 4) operational test and evaluation. In such live evaluations, feedback is currently limited to after-action reviews supported by manual methods that often lack a detailed accounting or complete understanding of lengthy, complex, and federated cyber activities. Automated data management and digital collection capabilities are needed to support structured observational assessments and to reduce data collection, aggregation, and analytical bottlenecks. This report presents a software application called the Behavioral Observations Logging Toolkit (BOLT), which assists data collectors and analysis by digitally collecting information about the human–machine interface in close proximity to the users of the system. Specifically presented and discussed in this report are requirements, functionality, use cases, prototypes, and evaluations of usability for several early iterations of the BOLT software.

15. SUBJECT TERMS
activity logging, task analysis, user observations, usability data collection, Human Systems Integration (HSI) analysis tools

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Christopher Garneau
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	SAME AS REPORT	36	19b. TELEPHONE NUMBER *(include area code)* (410) 278-5814

Table of Contents

List of Figures

List of Tables

1. INTRODUCTION

The project outlined here addresses the lack of valid tools and methods to support the depth/breadth of cyber assessments needed to facilitate and inform 1) training assessments, 2) advanced technology selection, 3) Human Systems Integration (HSI), and 4) operational test and evaluation. In such live evaluations, feedback is currently limited to after-action reviews (AARs) supported by manual methods that often lack a detailed accounting or complete understanding of lengthy, complex, and federated cyber activities. As a current baseline, noninstrumented, observational data collection processes during U.S. Department of Defense testing or experimentation typically involve human data collectors recording observations in a notebook or on a data collection form. There is a need for automated data management and digital collection capabilities to support structured observational assessments and to reduce data collection, aggregation, and analytical bottlenecks. The desired project objective is to improve the depth/breadth of assessments and dramatically shorten the delivery time of analytical results.

The Behavioral Observations Logging Toolkit (BOLT) assists data collectors and analysts by digitally collecting information about the human–machine interface in close proximity to the users of the system. This research and development program is part of an effort led by the U.S. Army Combat Capabilities Development Command (CCDC) Army Research Laboratory (ARL) and subsequently the CCDC Data & Analysis Center (DAC) to develop and deploy automation support to the collection and metric assessment of human task performance in complex cybersecurity environments. In general, BOLT uses are not limited to cybersecurity, and the tool may apply to any evaluation of HSI and performance. Development of the BOLT tablet is established within the first of three lines of effort in a project entitled "Performance Assessment Suite for the Cyber Mission Force" funded by the Office of the Under Secretary of Defense for Research and Engineering. Specifically, BOLT development is the primary line of effort called "Observer Tablet / Analytics Platform". The project objective is to develop observer tablets that digitally enhance data collection efforts and expert observational assessments. The aggregation of timestamped logs of cyber operator behaviors supports analysis and visualization of task completion, expert observational assessments, and human–computer activity over time. The tool provides process traces that will enable Cyber Mission Forces to foster and better align HSIs, reflect on their performance, identify best practices, identify needed changes to concepts of operations, and increase their agility. Observer tablets have the potential to increase the breadth and depth of cyber team performance assessments and increase inter- and intra-rater reliability.

For the initial prototypes and usability surveys discussed in this report, CCDC Army Research Laboratory collaborated with three external partners to bring the software to fruition: 1) the Massachusetts Institute of Technology Lincoln Laboratory (MIT-LL), 2) the U.S. Cyber Command (USCYBERCOM) Joint Artificial Intelligence Center (JAIC), and 3) Lockheed Martin Advanced Technologies Laboratory (LM-ATL). Collaboration with the MIT-LL group yielded the first prototype of the software developed for the USCYBERCOM Cyber Immersion

Laboratory (CIL). MIT-LL's contributions to the project concluded with this product. In early 2019 JAIC joined the effort and integrated the initial code to develop the first BOLT prototype. In mid-2019 LM-ATL also joined the effort to lend expertise in software development and product delivery. The focus of the latter portion of this report is on results from two usability evaluations of these prototypes that occurred in August and October 2019.

2. OVERVIEW OF TOOL CONCEPT, FUNCTIONALITY, USE CASES, AND METRICS

The purpose of the BOLT software is to provide a digital data collection mechanism for expert observers watching users performing tasks on a computer workstation. Often, but not always, there is a one-to-many mapping between observers and users. It is incumbent upon the observer to determine the tasks the user(s) are performing as well as the users' experience in performing those tasks (are they performing well?, confused?, need support?, etc.). Sometimes it may be difficult for an observer to determine the users' active tasks or current actions. In our specific case additional software was loaded onto the users' workstations that provides visual clues to the observer about the users' active program and the time at which the user switches programs (e.g., a colored frame applied to the edge of the screen). In the use case considered for these prototypes it was assumed that the observer could interact with the user for clarification. A key task for observers was to enter notes that were as detailed as possible for any user tasks or observations.

Given this overall concept, basic requirements of the tool are as follows:

- Tablet computer preferred (with keyboard for note-taking)
- Mechanism to specify scenario, log observer input, and save for later analyses
- Provision for quick input of user tasks based on scenario
- Ability to track user tasks (i.e., add, remove, bring back active task[s])
- Ability to add notes for specific tasks or actions
- Provision for timestamping and logging all observer actions

To support cyber product evaluations, the JAIC developed a specific use-case for BOLT: an observer watching one or two users at a time interacting with novel candidate software as they perform routine computing (cybersecurity) tasks. Note that while most of the discussion in this report focuses on this use case, future development should also consider other use cases. Given the specific product evaluation use case, the JAIC developed a set of metrics for evaluating the candidate software (the software under evaluation augmented traditional analyst procedures with artificial intelligence capabilities). While these metrics do not necessarily apply to BOLT, BOLT should enable evaluation of these factors. The JAIC metrics are shown in Table 1.

Table 1. JAIC Metrics

Assessment Category	Measure	Definition	Method	Measurement
Utility	Completeness and Efficiency	Frequency of high-quality network incidents reported by type and context in Session 2 each day	Post hoc assessment of participant-generated incident report	No. of incidents detected by type (i.e., CAT 0-9); Cyber Security Service Provider (CSSP) expert assessment of incident reports
	Task Sufficiency	CSSP scenario tasks supported by product	Observations	No. of subtasks completed using product, as scored by observers
	Data Sharing	Support for sharing data with others	Vendor reported	Assessment of data export/sharing types
	Accuracy	Accuracy of events identified in labelled data	Post hoc assessment of incident reports	Incident reports compared against ground truth in labeled data
Explainability	Transparency	Ability to display an audit trail, or data that are used to make inferences: Complete, Intelligible, Consistent	Post hoc assessment of incident reports	Scoring of Usability Questionnaire and CSSP expert assessment of incident reports
	Trust	User's assessment of their understanding of system function and ability to rely on the system to complete key tasks	Trust Questionnaire	Scoring of Trust Questionnaire
	User Model of Automation	Congruence of user's mental model with automated tool's process/model	Debrief	Qualitative coding of debrief comments
Usability	Directability	Ease of use and ability for operator to direct product elements such as filtering, sorting, user-defined prioritization, and threshold tuning.	Usability Questionnaire	Scoring of Usability Questionnaire
	Prioritization	Ability of product to direct user's attention to priority events	Usability Questionnaire	Identification of specific tool mechanisms to support prioritization and Usability Questionnaire scoring for those mechanisms

3. PROTOTYPES

This section provides screenshots for and discussion of the user interface for four prototypes of BOLT, showing its evolution from the CIL Observer/CIL Observation Recorder (v0), the BOLT JAIC prototype (v1), the first refinement of BOLT by LM-ATL in use for the August 2019 usability evaluation (v2), to the second LM ATL refinement of BOLT for the October 2019 event (v3). Discussion of added or refined functionality accompanies the screenshots.

3.1 CIL Observer (v0)

The CIL Observer software was written in C# with the Windows Presentation Foundation (WPF) user interface (UI) framework. Consequently, this prototype and the subsequent prototypes derived from it run only on Windows. The software uses a configuration file written in JavaScript Object Notation (JSON) format that specifies users, tasks, and groups of tasks for a scenario. The following code snippet shows a sample JSON configuration file:


```
{
  "Scenario": "Splunk",
  "Session": "1",
  "LocalTime": "2019-08-14T15:00:34.4247809-04:00",
  "Participants": [
    {
      "Name": "User 1",
      "Role": "user"
    },
    {
      "Name": "User 2",
      "Role": "user"
    }
  ],
  "TaskFiles": [ ],
  "Tasks": [
    {
      "Group": "Tier 1",
      "Immediate": false,
      "Name": "Configuring Application",
      "Order": 0
    },
    {
      "Group": "Tier 2",
      "Immediate": false,
      "Name": "Exploring Event Internal",
      "Order": 0
    }
  ]
}
```

Figure 1 shows the configuration screen populated from a saved JSON file. While the configuration file provides parameters for many or all of the required fields, these fields may be changed after loading the file. Figure 2 shows the main screen before a scenario has started. Clicking the "Start Session" button and adding some active task results in a log of actions and a pane with "Activities to Be Completed" (Figure 3). Clicking any of these active tasks results in the task disappearing and being marked accordingly (recorded as "stop"). Available activities are shown in the pane to the right of the main screen. CIL Observer saves data from each session locally on the device, in the directory specified in the Configuration view. Figure 4 shows the popup that appears when clicking "edit" for a log item, and Figure 5 shows the popup that appears when clicking "set participant" for a log item.

Figures 6 and 7 show the Survey view of the prototype. Subsequent prototypes of the software have omitted this feature, but some variant of it is expected to return in a future version.

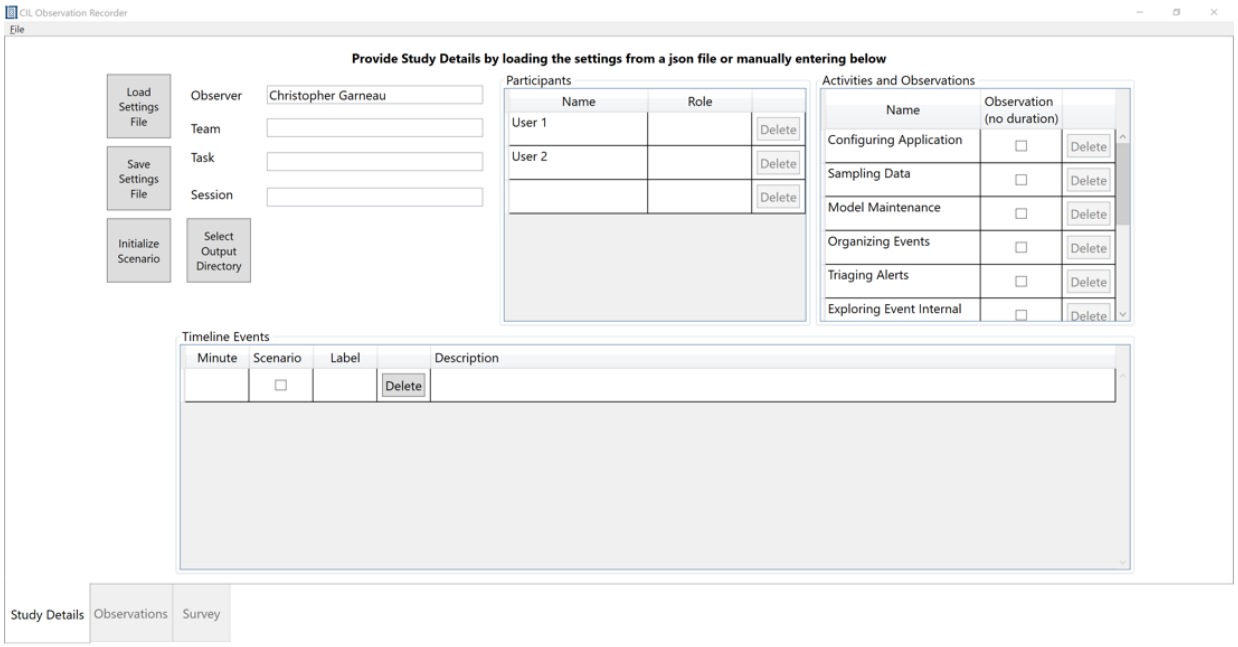

Figure 1. Configuration view with scenario loaded

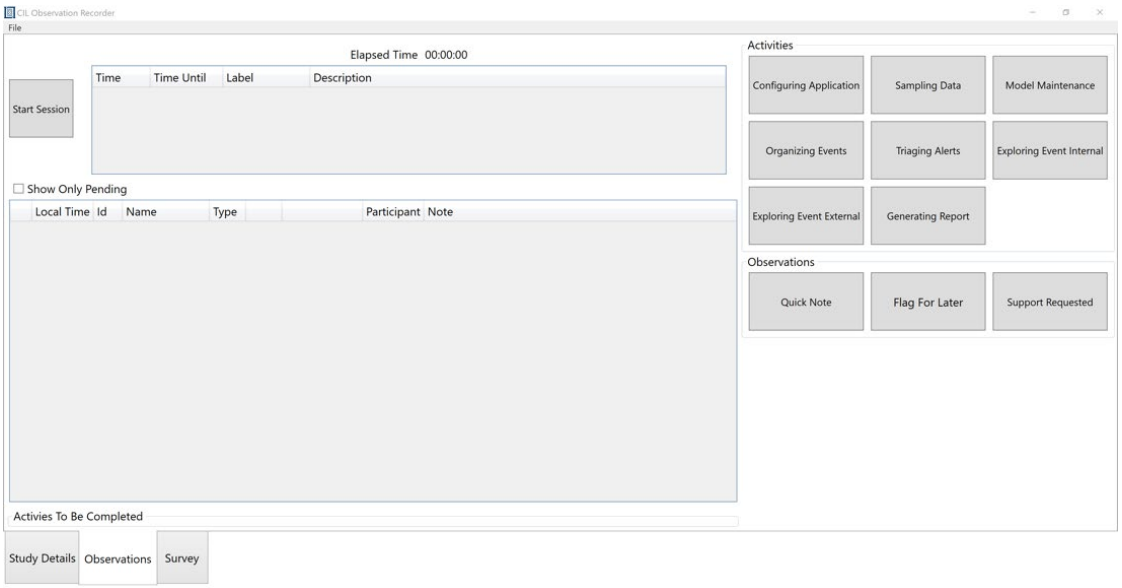

Figure 2. Initial observations view (before starting scenario)

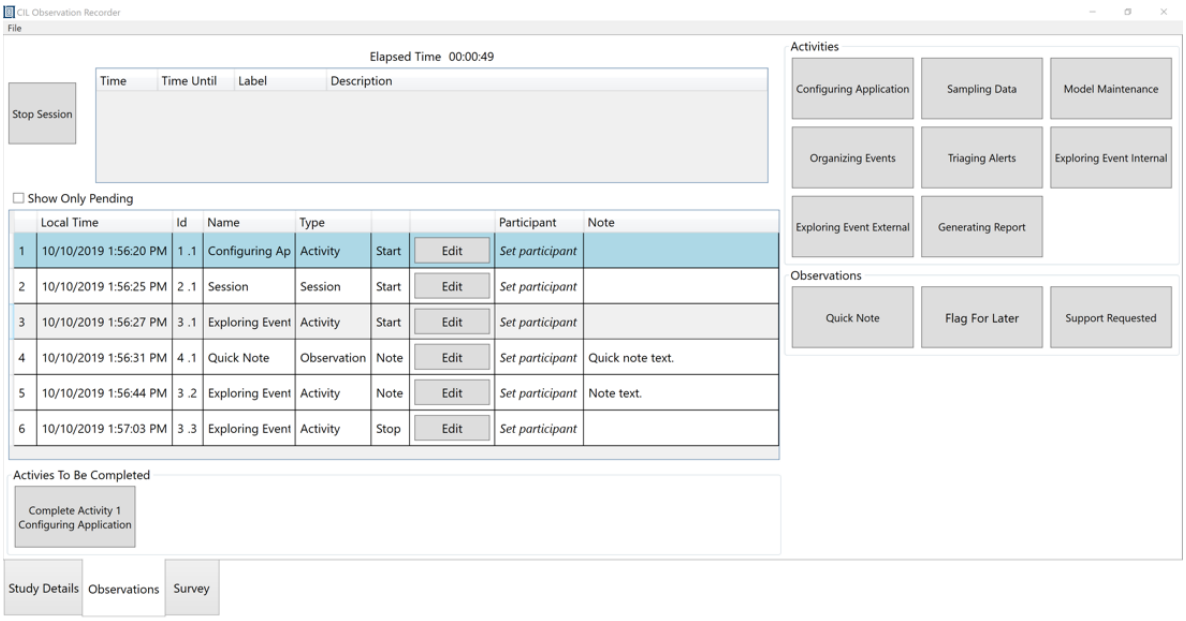

Figure 3. Observations view with tasks and notes added

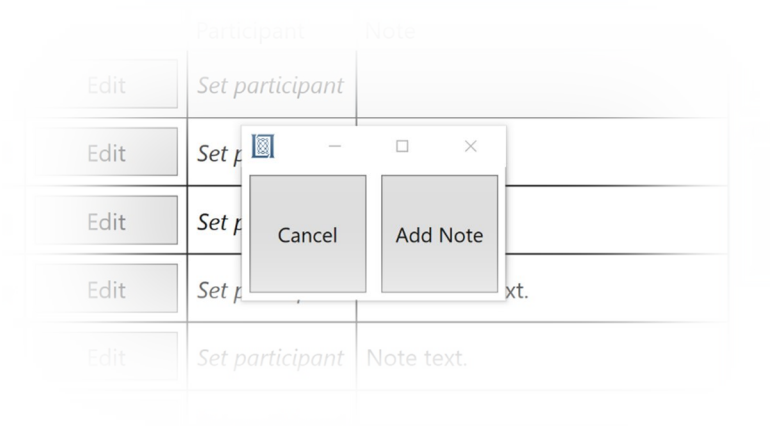

Figure 4. Popup that appears when clicking "edit" for a log item

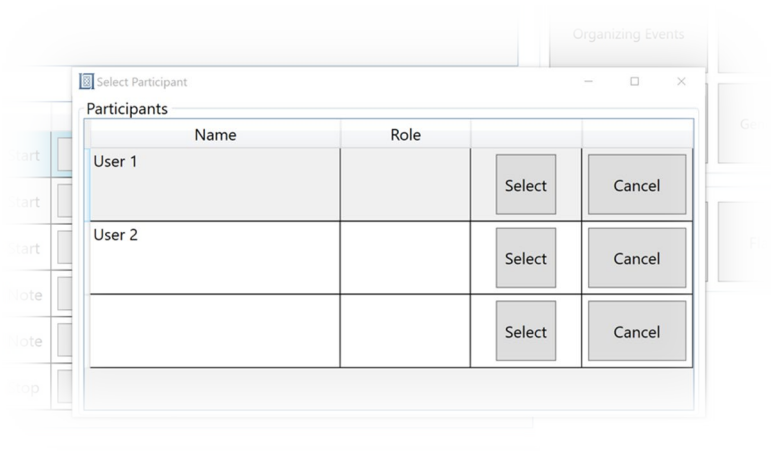

Figure 5. Popup that appears when clicking "Set participant" for a log item

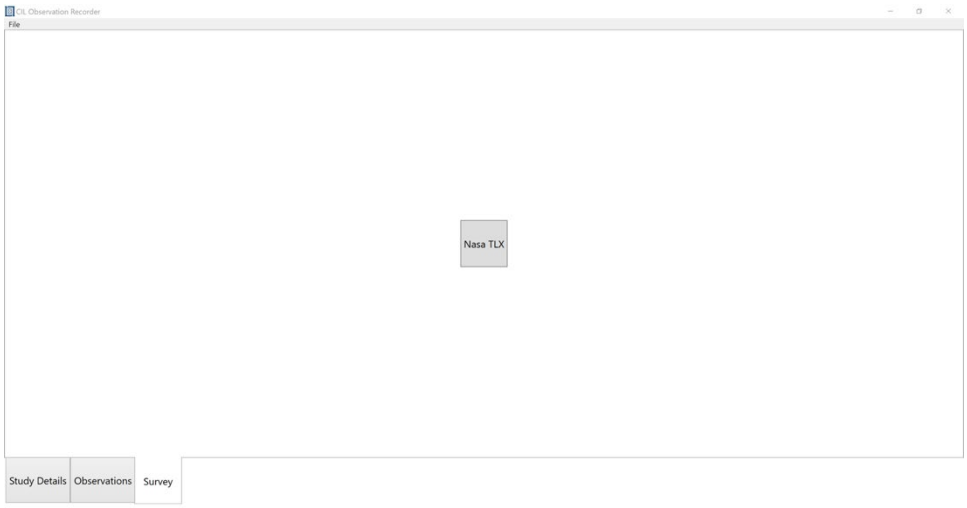

Figure 6. Survey view (only NASA Task Load Index [TLX] available in this prototype)

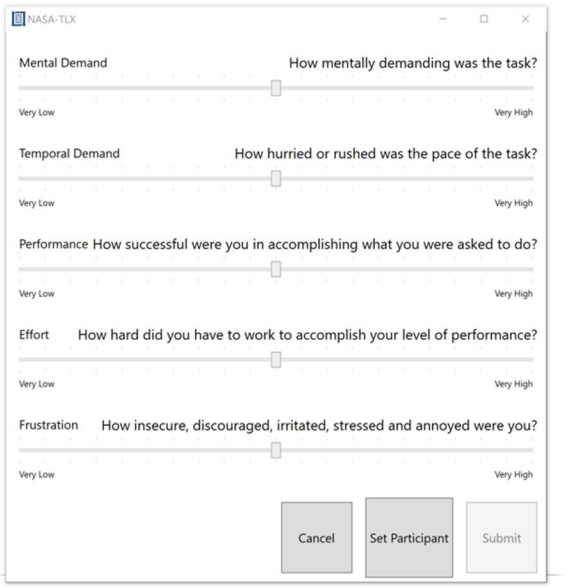

Figure 7. NASA TLX survey that appears when clicking the NASA TLX button

3.2 BOLT v1

The JAIC BOLT prototype (v1) follows the basic structure of the v0 prototype with two key UI modifications: 1) the interface was redeveloped as touchscreen-compatible (within the limitations of the WPF framework), and 2) in addition to being locally stored on the device, the results of the observer logging actions are streamed live over a local area network connection to a server. In turn, the server provides an administrative view of all observer inputs and interactions. The focus of this report and much of the initial BOLT development is the UI, designed for touchscreen interactions on tablet-style devices. However, BOLT as a comprehensive toolkit includes both the front-end and back-end servers. The assessment dashboard prototype on the server aggregated and visualized the observer inputs as task frequency charts, tabular displays, and survey completion percentages. Unlike prior evaluations, wherein observers recorded their findings in their own notebooks during the event and had to spend a lot of time annotating, editing, and aggregating data to produce the overall report, BOLT cut the time requirements significantly by automatically labeling, timestamping, and collecting data as they were transmitted from each front-end server. Leadership was able to immediately observe the utility and usability of the tools being evaluated in real time by checking the dashboard visualizations and monitoring the quality of the provided data live. Prior to using BOLT, the only indications of evaluation status and quality required interrupting observers to obtain their impressions. Likewise, there would have been significant time expended to collect, process, enter data, and clean up data to generate meaningful reports. The interconnection of each observer's tablet also enables the addition of a live chat function to BOLT, facilitating more-rapid notification of issues and sharing updates between observers, also potentially increasing inter-rater reliability.

Figures 8 and 9 show the Configuration view. Figure 10 shows the main screen before a scenario has started. Clicking the Start button and adding some active tasks results in a log of actions and a pane with Pending Tasks (Figure 11). Clicking any of these active tasks results in the task disappearing and being marked as having stopped. Available activities are shown in the pane to the right of the main screen. Figure 12 shows a touch-friendly popup that appears when editing a log item. Compared with v0, adding a note results in a new log item, but editing a note simply edits the existing text (versus creating another log item). New options also allow the observer to quickly rate a user's experience (the camera functionality did not work in this prototype). Figure 13 shows a popup that appears when an observer adds a task; this forces the observer to assign a participant to a task when making it active.

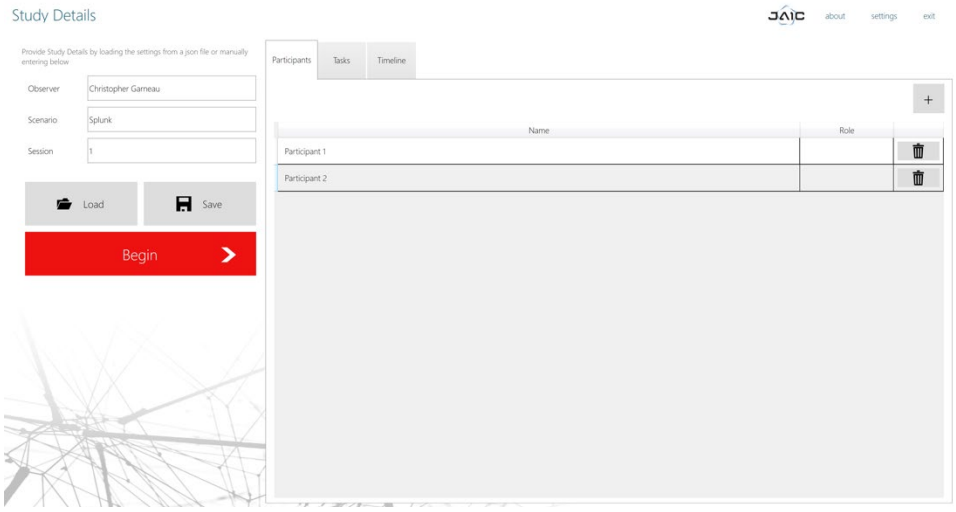

Figure 8. Configuration view showing participants

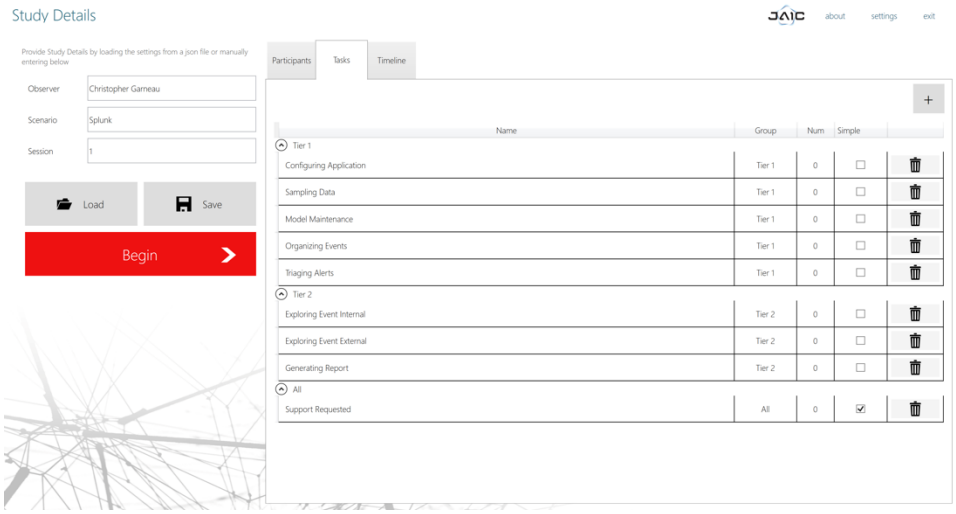

Figure 9. Configuration view showing tasks

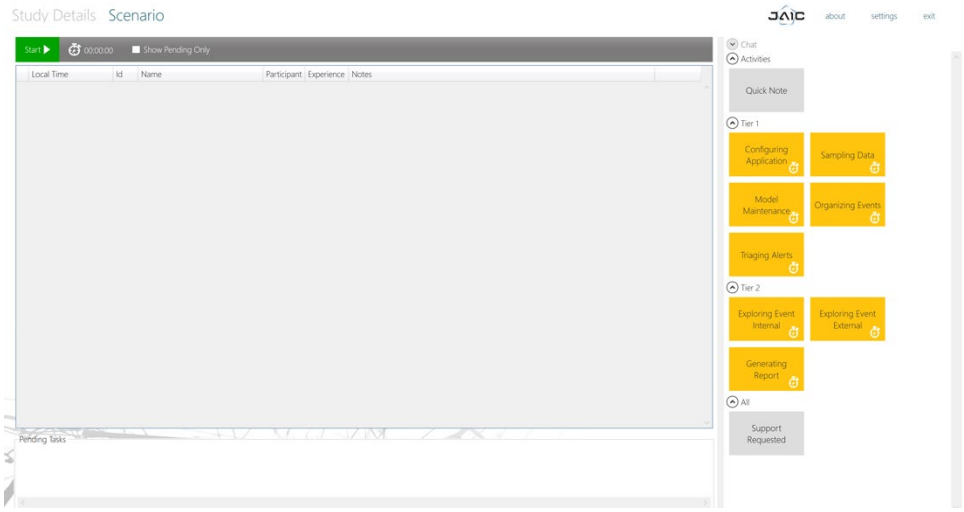

Figure 10. Initial observations view (before starting scenario)

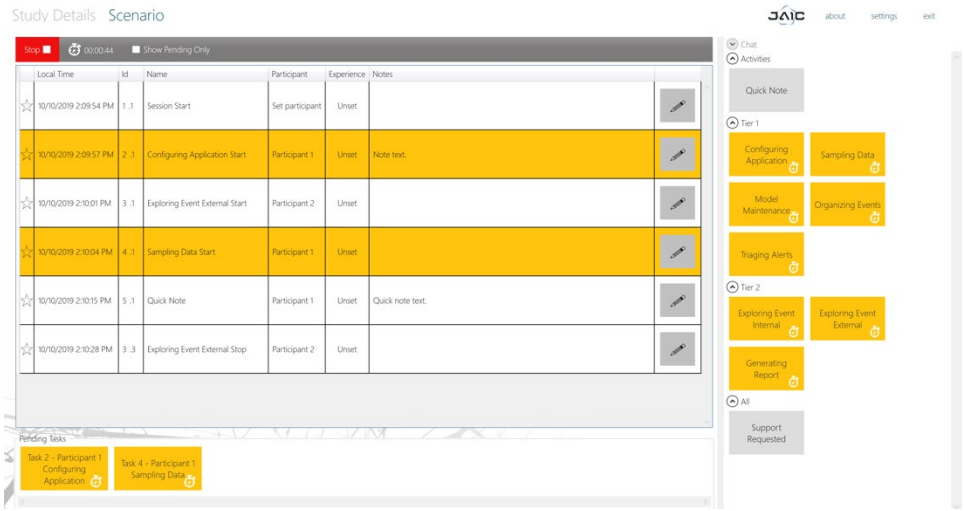

Figure 11. Observations view with tasks and notes added

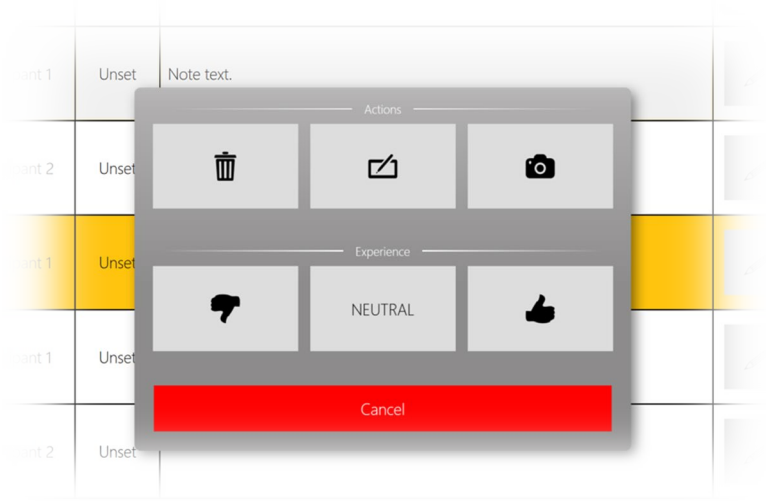

Figure 12. Popup that appears when clicking the pencil (edit) icon for a log item

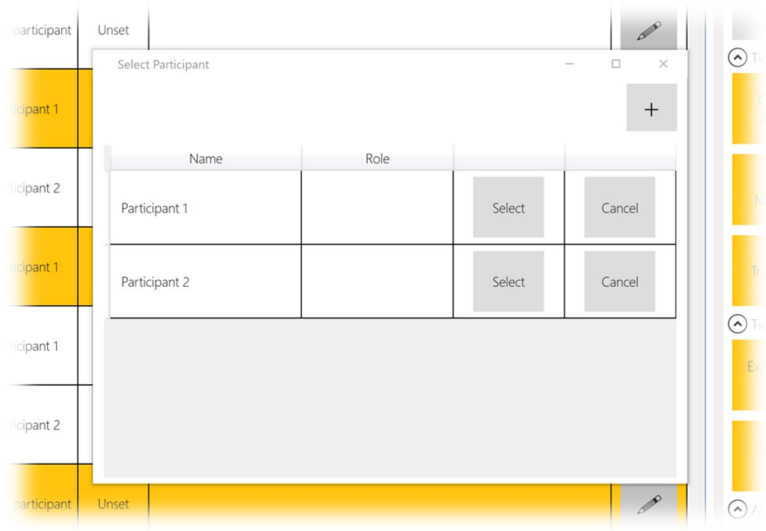

Figure 13. Popup that appears when adding a new task to assign a participant

3.3 BOLT v2

The JAIC- and LM-ATL-developed BOLT August 2019 prototype (BOLT v2) refines the v1 prototype with one key modification: separating user actions into panes above the log items. The active-user toggle (upper right) determines the task assignment.

Figure 14 shows the Configuration view. Figure 15 shows the main screen before a scenario has started. Clicking the Start button and adding some active tasks results in a log of actions and panes for each user filled with their active tasks (Figure 16). Clicking any of these active tasks results in the task disappearing and it being marked as having stopped. Available activities are

shown in the pane to the right of the main screen. Figure 17 shows the popup that appears when editing a log item.

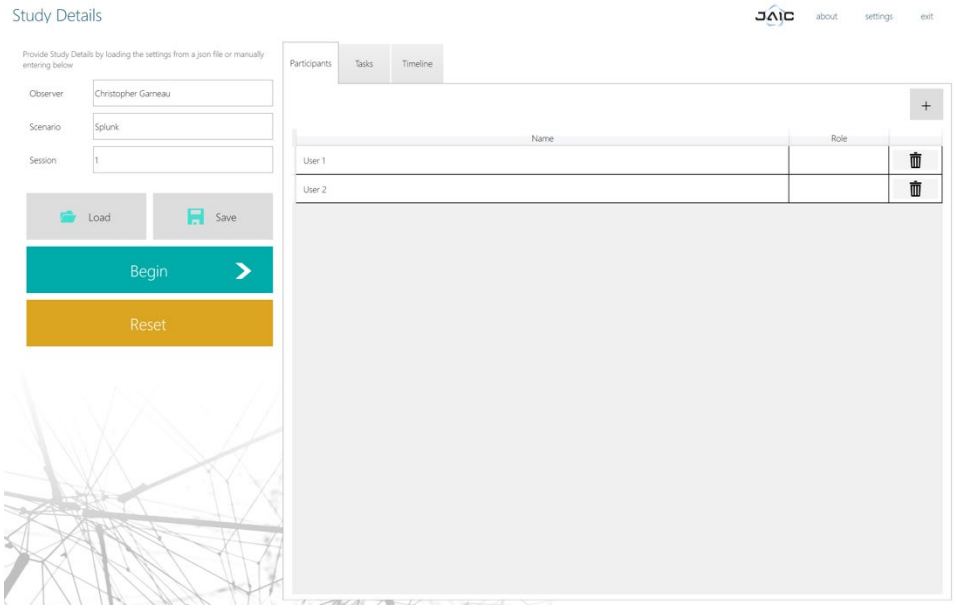

Figure 14. Configuration view showing participants

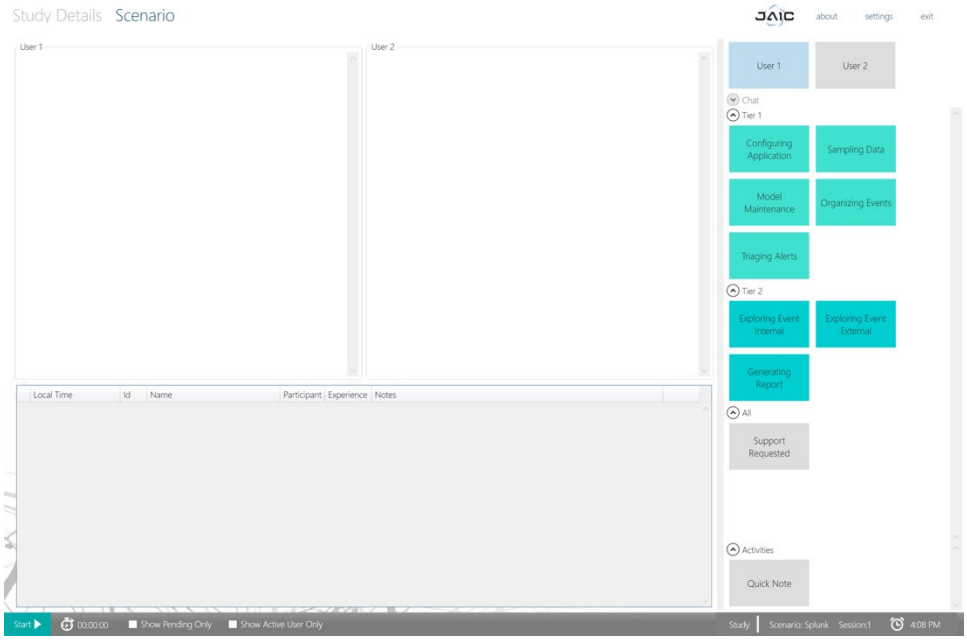

Figure 15. Initial observations view (before starting scenario)

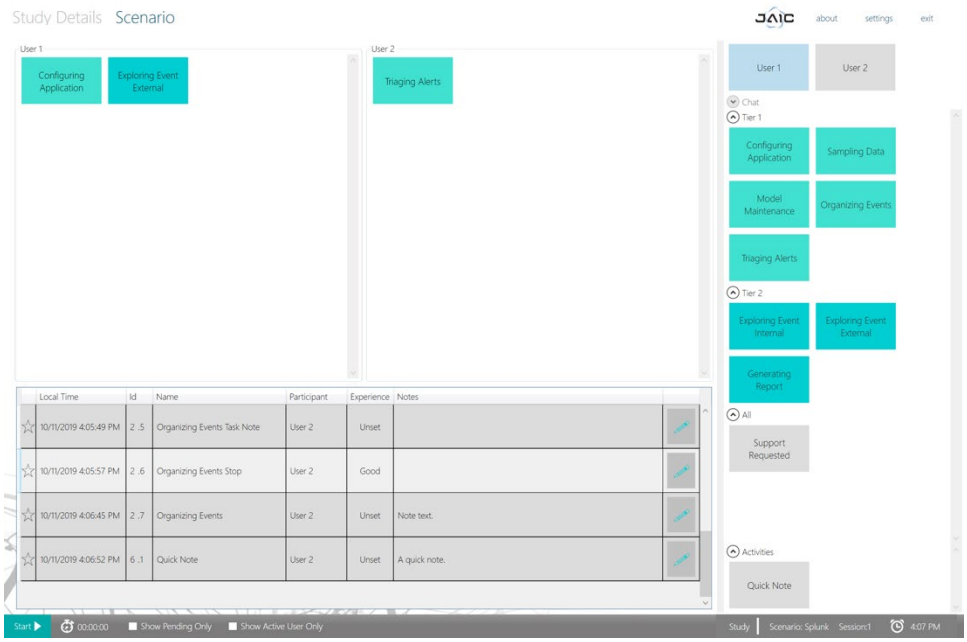

Figure 16. Observations view with tasks and notes added

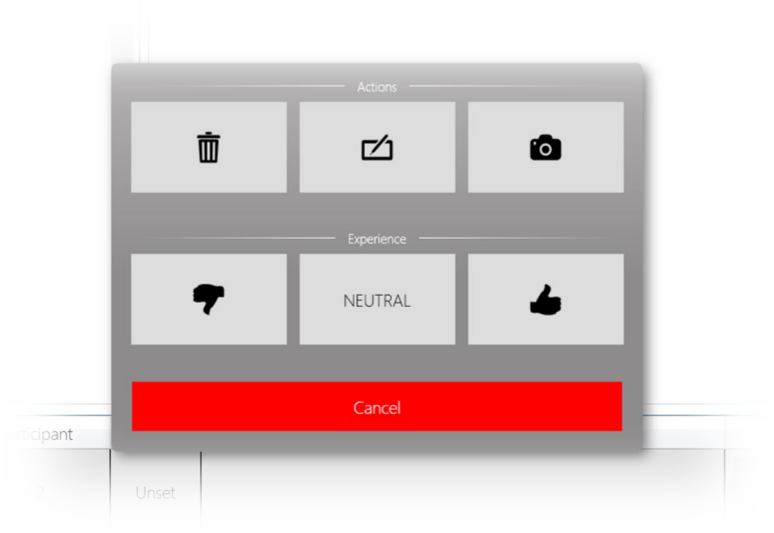

Figure 17. Popup that appears when clicking the pencil (edit) icon for a log item

4. BOLT V3

The JAIC- and LM-ATL-developed BOLT October 2019 prototype (BOLT v3) largely retained the v2 prototype UI but made the following significant modifications:

- Elimination of participant selection: when recording tasks, observers now just tap the pane corresponding with their participant name in the main view (Figure 18), eliminating the extra step of selecting participant buttons in the upper right panel prior to data recording
- Note history: a new note history view appears when an observer enters a note (Figure 19)
- New notes functionality: additional functionality to interact with the tabular note display at the bottom and edit existing notes
- Various other minor UI changes, including color scheme, task bar layout, and more

In addition to UI improvements, v1, v2, and v3 have each improved the structure and efficiency of the previous version's code.

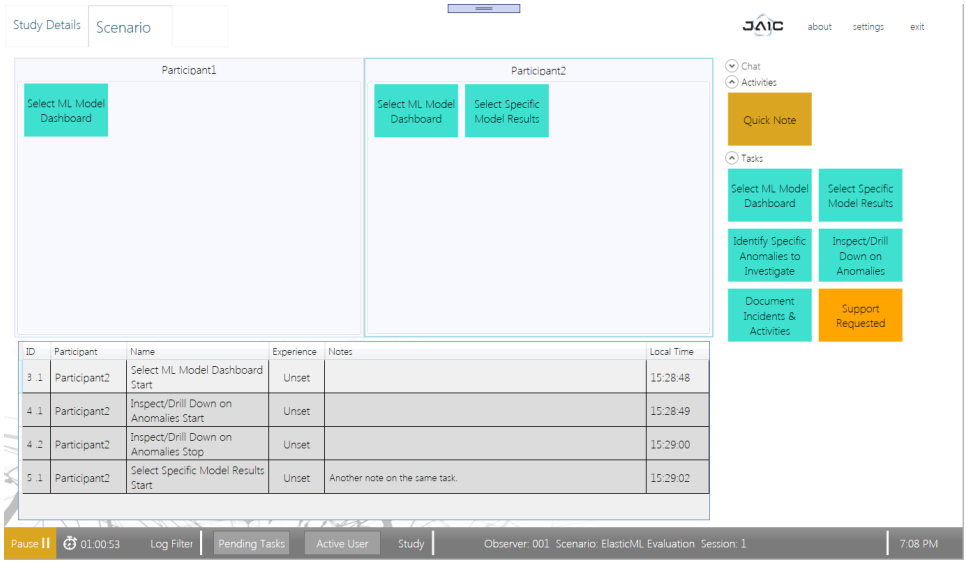

Figure 18. Observations view with tasks and notes added

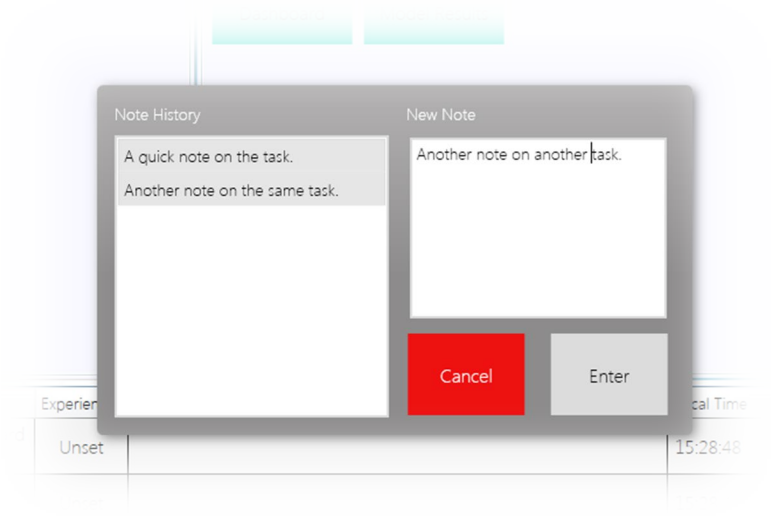

Figure 19. Popup that appears when adding or editing an item's note

5. USABILITY EVALUATION (BOLT V2)

To gain feedback on the progress of development for BOLT, an informal usability evaluation was held in August 2019 at JAIC, using a sample of convenience with developers and other program stakeholders. The total sample size was seven individuals. Potential participants that were already highly familiar with the tool were excluded from the sample.

5.1 Context and Assigned Tasks

A simple script was read to participants to provide some context for their evaluation. The group was asked to evaluate BOLT v1 and v2 to accomplish the tasks (v1 preceded v2 for all evaluators). The script included the following context and prompts:

- **Introduction**: In this exercise, the example evaluation event is the use of a tool like Splunk to parse and manage alerts, sample data and evidence, create and use models to explain the data observed, and appropriately report on incidents from the data set.
- **Task 1**: Take a look at BOLT and load a scenario. There may only be one available.
- **Task 2**: Now that your scenario is loaded, please start it.
- **Task 3**: One of your users is making use of the data-modeling features of Splunk to observe their effects on the data presented. However, you notice that the user seems to be going between the model and the results, which indicates to you potential uncertainty in its use or utility. Later you watch as this user collates information provided by Splunk and the queries generated as evidence for the submission of an incident report.
- **Task 4**: Another user grabs your attention, and you take notice to record the activity. This user has been ordering and re-ordering the alerts presented and appears to be capitalizing on this to manage their priority. At another time this user has pulled up more details on an individual event logged and, from what you can tell, identified a potentially malicious actor by highlighting and making note of the IP address and the context of the event and associated alert(s).
- **Task 5**: Mark one of the tasks you have recorded as active.
- **Task 6**: Consider the tasks you entered and what notes may be applicable for them. Add a note to one or more of these tasks.
- **Task 7**: Remove a task from a user.

These tasks were read sequentially to the test group, but not all evaluators followed each instruction. Some instead preferred to read the prompts and explore the tool at their own pace.

5.2 System Usability Scale Questionnaire

The seven participants were asked to complete the following System Usability Scale (SUS) questionnaire after using each tool. The SUS asks respondents to rate each of the following statements on a five-point Likert scale from 1 (strongly disagree) to 5 (strongly agree).

1. I think that I would like to use this system frequently.
2. I found the system unnecessarily complex.
3. I thought the system was easy to use.
4. I think that I would need the support of a technical person to be able to use this system.
5. I found the various functions in this system were well-integrated.
6. I thought there was too much inconsistency in this system.
7. I would imagine that most people would learn to use this system very quickly.
8. I found the system very cumbersome to use.
9. I felt very confident using the system.
10. I needed to learn a lot of things before I could get going with this system.

Figures 20 and 21 summarize participant responses for each statement. Note that while the SUS has five points, Figures. 20 and 21 compress the results to three: agree, neutral, and disagree. For both figures, questions are grouped into positive and negative phrasing (i.e., agreeing with the statement means a positive reaction or feeling about BOLT vs. a negative one). Totals are then graphed as stacked rows. BOLT v2 was clearly better-received, as Figures. 20 and 21 are almost mirrors of one another at first glance. The new version was not a total win, however; the individual notes and commentary reveal and explore that further.

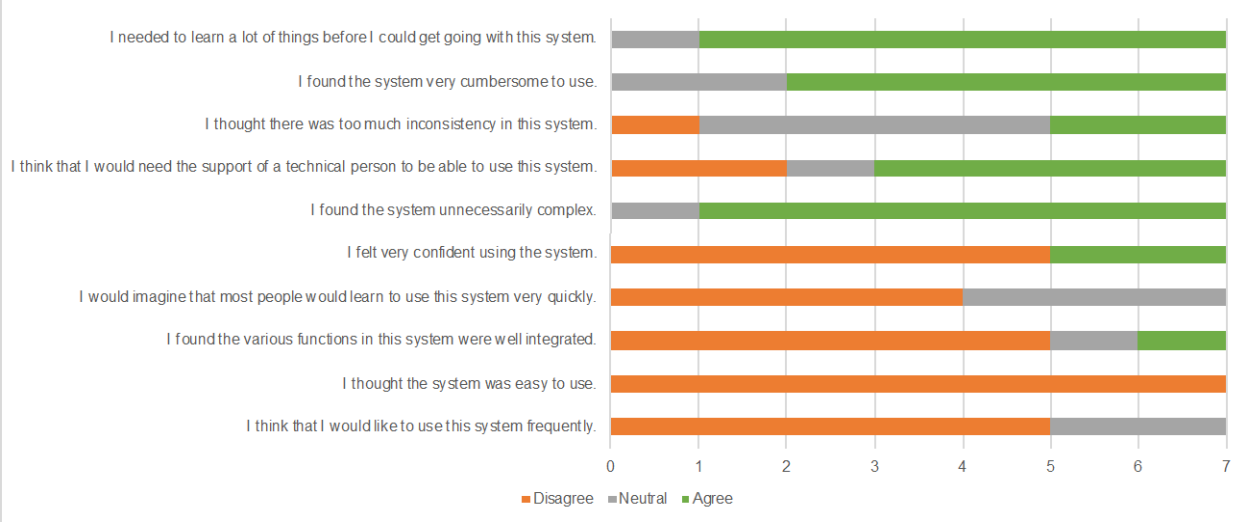

Figure 20. SUS responses, BOLT v1, across the seven participants

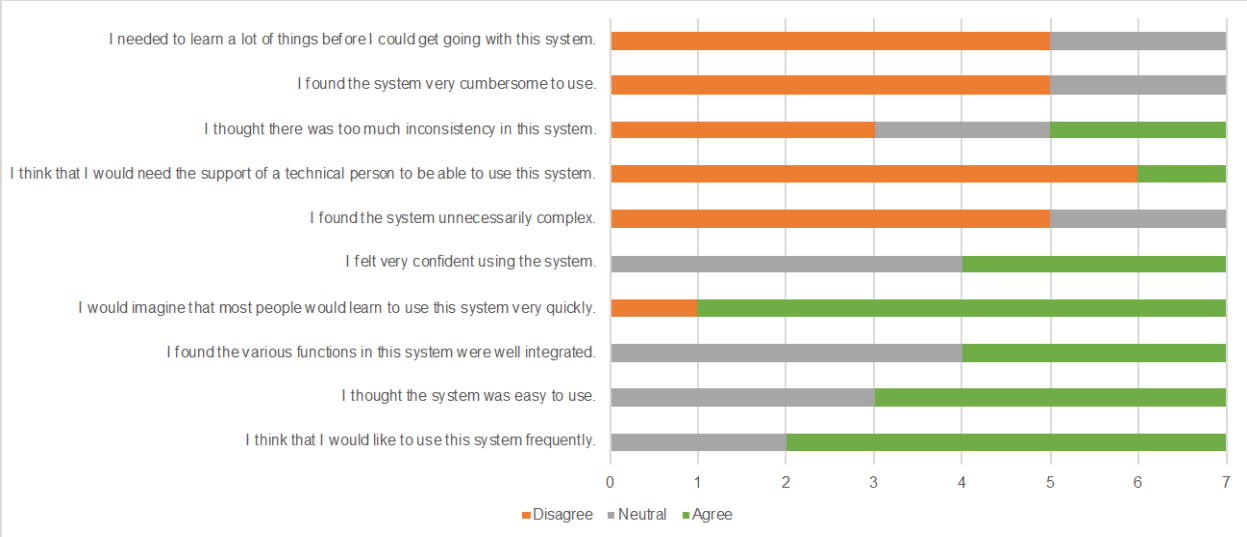

Figure 21. SUS responses, BOLT v2, across the seven participants

6. EVENT FEEDBACK (BOLT V3)

BOLT v3 was used at an evaluation event at JAIC during the week of 21 October 2019. The tool was used by a pool of eight observers to evaluate the performance of eight analysts (with a one-to-one mapping of observers to analysts) using a toolkit for network defense with machine learning capability added. While the performance of the toolkit and details of that evaluation are not described here, an AAR at the end of the evaluation provided useful feedback about the progress of usability improvements in BOLT. A questionnaire was administered with a free-response section and a SUS questionnaire section.

6.1 Free-Response Qualitative Feedback

Observers were asked to "identify 3 aspects of the tool that worked well for the evaluation" and also to "identify 3 aspects of the tool that should be improved for the future". The results of these free-response questions are summarized in Table 2.

Table 2. Free-Response Portion of the AAR. Responses Reported Verbatim as Written on the Questionnaires.

Participant Responses for Three Aspects That Worked Well		
1	**2**	**3**
Size of tablet was great—not too small	Screen/window was displayed for easy use	Able to use touchscreen and keyboard
Ability to quickly drill-down/follow-along	Taking quick notes	…
Selecting, opening, and closing a task button	Writing notes with keyboard and touchscreen	Appending additional information/note to the previous one
Easy to use	Flexibility of use, keyboard, and touchpad	Note taking was easy
Intuitive interface	Touchscreen utility	Pre-built tasks created and selectable
Ability to launch each task easily	Being able to append to previous notes	UI is easy to navigate and intuitive
Keyboard made note typing very easy	The touchscreen made clicking very easy and fast	Tool was simple and didn't include unnecessary functions or data. It was mostly pre-configured.
System included sufficient resources (RAM, battery life, CPU, etc., which made it very stable to use)	Keyboard was an excellent addition; without a keyboard it would have made it much more difficult and time-consuming to use	User experience was simple to use

Notes: RAM = random access memory; CPU = central processing unit.

Table 2. Free-Response Portion of the AAR. Responses Are Reported Verbatim as Written on the Questionnaires.

Participant Responses for Three Aspects That Could Be Improved		
1	**2**	**3**
Processes need to be linked to what an analyst is performing during the task	Spell check	Scrolling through past written comment needs to be easier to locate
Following if the analyst quickly changed tasks	Fix errors if the wrong button was hit	…
Scroll should be added by the side of the tasks to make it easier to see previous notes	Spell-check should be added to help check for observer's errors	Predictive text of what's already in the tab to help with faster writing
Too many task buttons	Need a page for each task button and on each page keep a continuous list of notes each time you click in that task	Keep notes together and organized
Larger tablet	Predictive text input based on previous input	Active focus indication for multiple tasks
Being able to start and stop the timer/clock when switching between task	…	…
Typing multiple notes for the same task showed as separate events; simplifying note additions would help	Adding an undo button for tasks that were started on accident	Having pre-written notes to help start—a.k.a. "common notes"
Larger form factor and keyboard	Add an optional stylus	Add canned note prefixes and context-sensitive task buttons (sort, filter, selected Direct Numerical Simulation model, etc.)

6.2 SUS Questionnaire Results

As with the earlier usability evaluation, observers were asked to complete the SUS questionnaire at the completion of the event. Overall, BOLT v3 scored an 86.9 among the eight observers. Among the three observers who had participated in a previous event, BOLT v3 scored an 87.5. This suggests that the high score was not biased by a rebound effect among users of the tool who had a poor experience with a previous version, though this conclusion cannot be made with any statistical significance due to the small sample size.

Finally, Figure 22 shows a benchmark reference for interpreting overall SUS scores normalized on a 100-point scale: BOLT v1 scored 29.3 overall, BOLT v2 scored 65.7 overall, and BOLT v3 scored 87.5 overall. This agrees with the detailed breakdown in Table 1 and shows a marked improvement of the usability of BOLT v2 and v3 over v1.

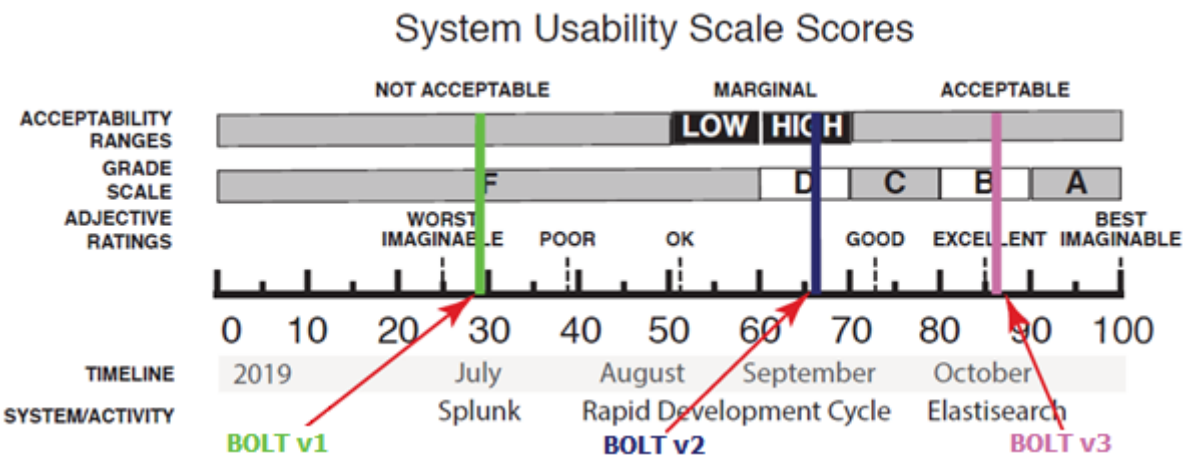

Figure 22. Benchmark normalized SUS scores for the BOLT

7. CONCLUSION AND FUTURE WORK

BOLT has been well-received as a useful data-collection suite for assessing user performance in the completion of complex analytical tasks using a computer workstation. While it has progressed from an initial prototype through multiple revisions, and improved with each iteration, much work is still to be done. Future versions will improve BOLT for cyber security assessments and will focus on the following: 1) reintegrating and enhancing the rapid development and deployment of surveys and checklists (this feature was removed from the BOLT v1), 2) incorporating additional measures of user performance, particularly in the aggregation of inputs for the assessment dashboard, and 3) refining the user interface to best facilitate observational user assessments.

Given the widespread applicability of digitally enhanced performance assessment, it is highly worthwhile to also explore new use cases beyond cyber security and technology assessment, examining the potential application of BOLT in other domains where assessment of user–device interaction is of importance to overall system performance. We have identified three categories or use cases on which to focus our development roadmap for future work, as shown in Figure 23. The cyber security work discussed in this report largely falls within the Technology Assessment category. For further investigation, we have identified the additional categories of Training Assessment and Human Systems Integration (HSI).

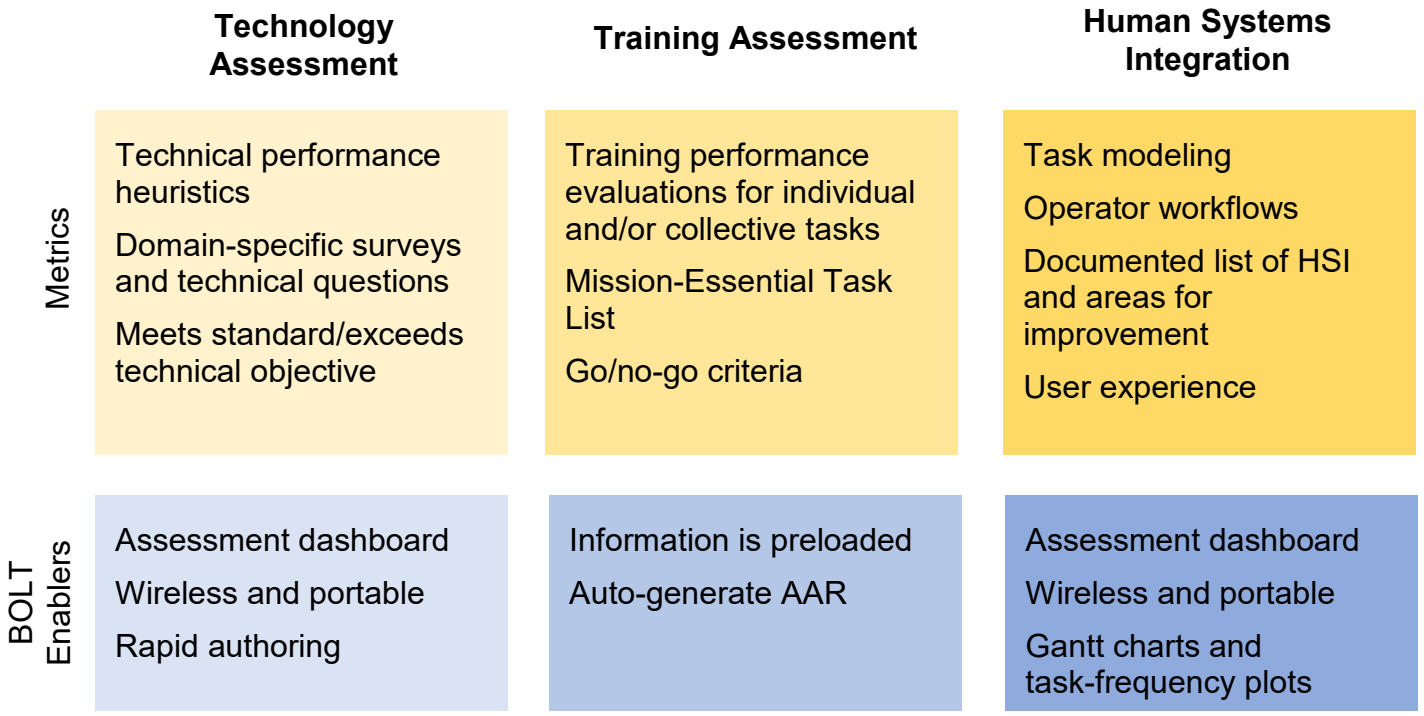

Figure 23. Use cases/categories for future BOLT development

As we consider these new use cases it is essential to understand the strengths of the behavioral observational method represented by BOLT and how it will enable better assessments. Often assessments are conducted with pen and paper techniques (or their computerized analogues using word processors and/or spreadsheets). We see BOLT as a means to support the use cases in Figure 23 in a way that improves upon what is possible with pen and paper methods. In addition to user interface improvements and supporting on-site data collection and observation, BOLT capabilities enable integration and data labeling capabilities that were either not possible or excessively laborious in analog alternatives. As events, exercises, and research scenarios scale in complexity (e.g., time taken, number of participants to track, systems involved, and concrete observables to record and manage), it is critical to minimize or eliminate any addition to the interactions and cognitive workload of observational personnel.

Pen and paper systems have two significant shortcomings for behavioral observation: 1) they do not scale well, and 2) they require manual processes for all stages of data handling. For scaling, each increase in requirements adds to the workload and cognition required by observers to synchronize timing data, differentiate among observed users and/or systems, and keep distinct records separated and labeled appropriately. For manual processes, pen and paper implementations require aggregation, collation, and merging to generate meaningful reports and analysis from the collected data.

The evolution of BOLT integrates the user interface improvements with the data synchronization, management, and analytic capabilities of a digital toolset, allowing for live updates as data are entered, automatic organization and binning of individual data points, and immediate aggregation of data to generate analytics and create reports. Ideally, the observer workload will stay relatively manageable and constant when using BOLT even as scale increases, allowing human observers to focus on providing more meaning and depth to the data collected, thus creating deeper understanding of systems' functioning, user task flow, and subsequent HSI improvements.

As discussed and demonstrated in this report, BOLT already supports the technology assessment use case. Observers can easily gesture and tap to track user actions and system events relevant to the technology under evaluation to provide rich qualitative data as context. However, continued development of BOLT will provide additional methods to assess expanded technology and systems. For example, rather than tracking activities and use, BOLT can be used to complete assessments that focus on the capabilities, functionality, and performance of systems under test (e.g., bandwidth of a system in a congested network, and the quality of communications, performance, and timing of a system scored by a defined requirement threshold). This will provide an evaluator with an interface to rapidly enter data over the course of a test and an immediate means to visualize and otherwise monitor those data. Depending on the evaluation, these data might include Key Performance Parameters, Measures of Effectiveness, and Measures of Performance.

For the use of BOLT for training assessment, there is a need to understand the effectiveness in a training course or system to convey the information to participants and for them to retain that knowledge and recall it appropriately. To evaluate training, BOLT may be used like a technology assessment. Observers can use BOLT to track a user interacting with a system or process prior to training and after training, creating a paired data set enabling the comparison of untrained and trained users. That comparison may reveal performance increases or a better understanding and use of the involved equipment and systems—or it may not, indicating a need to improve or change the training. Alternatively, BOLT can be used directly by users during training to self-identify their workflow, understanding, and knowledge, giving exercise leadership an immediate look into how well training participants are attaining or exceeding standards of performance for the tactics, techniques, and procedures under evaluation.

Many systems and platforms require HSI testing to ensure they function safely and appropriately and satisfy mission requirements. In the HSI use case, BOLT can be used in several ways to 1) track both the required processes relevant to the given mission or operation as well as additional ones that arise in practice or due to limitations, allowing assessments of workflow, 2) build empirical task models from real-time observations and assess cognitive workload (and/or to validate a task network simulation), or 3) gather observations from users interacting with a system to identify issues with usability, performance, safety, and the like. Using BOLT for these tasks allows the analyst to use initial predictions to prepopulate the scenarios in BOLT (thus saving substantial time and effort during observations). However, BOLT's real-time authoring capabilities will also enable the analyst to quickly capture and report on conditions that were not anticipated.

Across all three of these use cases, the addition of survey capabilities will provide research and assessment with the means to easily gather input and data from users relevant to the mission goals, such as usability and utility evaluations of technology, understanding and scoring training effectiveness and situational awareness support, and gauging human factors engineering or effectiveness. The capability to both predefine metrics of performance and survey instruments and rapidly deploy them with BOLT, and to generate performance cut-sheets and survey items on-the-fly for immediate and emerging topics, will give research and engineering personnel the means to easily design and implement studies and field exercises with fairly minimal effort as well as the flexibility to adapt to issues, questions, and topics of interest as they are encountered.

Collectively, developing the front-end UI further and growing the back-end as a modular platform that can support multiple use cases establishes BOLT as a fundamental tool for data collection and analysis. Its digital and networked capabilities mean that data can be collected quickly, labeled automatically, and integrated as it is collected. The flexibility in the style of data collection will allow BOLT to support an increasing number of use cases, providing rapid, human-observation data collection to multiple domains.

Appendix A – List of Acronyms

AAR	after-action review
ARL	Army Research Laboratory
BOLT	Behavioral Observations Logging Toolkit
BOLT v3	BOLT October 2019 prototype
CCDC	U.S. Army Combat Capabilities Development Command
CIL	Cyber Immersion Laboratory
CPU	central processing unit
CSSP	Cyber Security Service Provider
DAC	Data & Analysis Center
HSI	Human Systems Integration
IP	Internet Protocol
JAIC	Joint Artificial Intelligence Center
JSON	JavaScript Object Notation
LM-ATL	Lockheed Martin Advanced Technologies Laboratory
MIT-LL	Massachusetts Institute of Technology Lincoln Labs
NASA	National Air and Space Administration
RAM	random access memory
SUS	System Usability Scale
TLX	Task Load Index
UI	user interface
USCYBERCOM	U.S. Cyber Command
WPF	Windows Presentation Foundation

Appendix B – Distribution List

ORGANIZATION

U.S. Army CCDC Data & Analysis Center
FCDD-DAD-OL/T Resetar-Racine
FCDD-DAH-C/C Garneau
FCDD-DAH-N/B Hoffman
FCDD-DAH-N/N Buchler
6896 Mauchly St.
Aberdeen Proving Ground, MD 21005-5071

U.S. Army CCDC Army Research Laboratory
FCDD-RLD-CL/Tech Library
2800 Powder Mill Rd.
Adelphi, MD 20783

Defense Technical Information Center
ATTN: DTIC-O
8725 John J. Kingman Rd.
Fort Belvoir, VA 22060-6218