Low-Level Automation as a Pathway to Appropriate Trust in the PED Enterprise: Design of a Collaborative Work Environment

Authors: Michael P. Jenkins, Arthur Wollocco, Michael Farry, Martin Voshell

Abstract

In Military Intelligence, Processing, Exploitation, and Dissemination (PED) functions are critical to success, providing capabilities that support intelligence request lifecycles. PED capabilities are becoming increasingly available to smaller, more centralized teams that support multiple battlespace operators. Lessons learned reveal that while existing PED processes provide detailed intelligence, operators are burdened with continual coordination, communication, and interpretation tasks that, coupled with the data volume generated by multi-INr ISR platforms, cause breakdowns in coordination and communication. These breakdowns result in failures to share, attend to, and interpret information critical to mission success. Key contributors to these breakdowns are the lack of automated support and cognitive incongruence between existing automated solutions and the support required by analysts, which reduce or destroy trust. Developers must recognize that system interactions that establish a human-machine dialogue are necessary to develop the trust required for automation’s successful adoption and employment. To overcome these challenges, we present a Collaborative Work Environment for PED operations. It includes decision-centered analytics automation and low-level task automation designed to establish the human-machine dialogue that engenders trust. This approach facilitated appropriate attitudes of trust, leading analysts to request that we develop and deploy higher-level, planned automation capabilities to further offload PED tasks.
1. Introduction

Maintaining military superiority in the 21st century is of utmost importance to the United States armed forces, but this preservation does not come without cost and significant changes to doctrine, ideology, and process. In order to dominate the 21st century battlefield, the Army has the need to transform from the premier land force of the past, and enhance itself for the evolving conflicts ahead. Warfare and battlefield operations are evolving at a previously unseen rate, causing an increased emphasis on decision dominance and speed. This emphasis is appropriate, because it drives and informs the decision making process at every echelon. Decision management is enabled by information superiority, which can be defined as the speedily generation, collection, and effective use of information to inform Commander’s battlefield intent. The battle rhythm is shaped by this intent, which is passed through various echelons and different units through “information requirements” (IRs), or intelligent, directed, and relevant questions about the environment and enemy as a whole. Commanders guide and direct the battlespace flow through these IRs, with Military Intelligence (MI) units conducting direct intelligence support for the Commander’s intent. Therefore, it is of critical importance to the war effort to generate and answer these Information Requirements as quickly as possible, achieving an optimal, persistent, and common battlespace view that supports rapid decision making and execution.

Due to the importance placed on this decision-making process in the evolving battlespace, it is evident why recent Army focus has shifted attention to the proliferation of new Intelligence, Surveillance, and Reconnaissance (ISR) capabilities. These capabilities have advanced significantly in recent years, with the introduction and increasing dominance of unmanned aircraft and enhanced spending in sensor technology across all military domains. To directly support the rapid decision-making and execution process, new advances in ISR technologies have focused on making as much data and information available to Commanders, their staff, and anyone assisting with the current mission. Information collection is provided by processing IRs and other essential elements of information relative to the mission and available through common data sources. This collected information, and, more importantly, the information generated from its analysis, drives the entire flow of tactical and strategic operations by answering and addressing the IRs, and giving all stakeholders a transparent view of the battlespace.

However, for system developers looking to provide technology-based support with automated capabilities, it is still unclear how this information collection is conducted, converted into useful and manageable intelligence, and distributed throughout the tiers of military command. Processing, Exploitation, and Dissemination (PED) is defined as the conversion of collected information into forms suitable for the production of intelligence (TAPED 15 JP-01-02). It is the process where analysts receive Commander’s intent from IRs, then collect and analyze raw intelligence, converting it a form usable by the Commander. Analysts conducting PED are required not only to collect (e.g., through tasking INT assets) information, but also must facilitate the understanding of collected information, including factors such as the intent and capability of hostile forces, effects of terrain and weather, and behavior of civilian populations in a particular operating environment.

PED has been conducted for decades in continually evolving formats, but its key functional components are getting increasingly difficult to define as new technologies and missions continue to outpace force structure changes and blur the lines between individual roles, responsibilities, and authorities. As a result, system designers looking to provide relief for PED operations with shrinking manpower and rapidly expanding volumes of MultiINT data need guidance to ensure their developed capabilities will succeed. This guidance must be grounded in a robust and deep understanding of the PED force structure; capability gaps; existing tactics, techniques, and procedures (TTPs); and the future of PED evolution to ensure novel systems succeed and thrive within the challenging PED landscape.

This paper presents a brief overview of the current state of PED and how the domain is trending. This domain information is based on a series of knowledge elicitation (KE) interactions between our team and the PED community, including observing and interacting with traditional PED attachments, more tactical and rapid PED support teams, and the centralized PED reachback operations Cyber Center for Excellence at Ft. Gordon. We then discuss the need for automated support capabilities to meet the challenges of current and future PED. We then shift focus to the importance of establishing and maintaining trust in automation as a critical requirement to succeed given the constraints of the longstanding PED culture, workflows, and TTPs. Next, we provide a series of recommendations for system designers hoping to provide technology-based support to PED analysts. Finally, we give an overview of our ongoing efforts to design, develop, evaluate, and rollout a successful Collaborative Work Environment to meet a number of challenges facing current and future PED (which we refer to as the PED-X).

1.1. Processing

Processing is the initial phase of the PED workflow—the conversion of collected data into usable information by humans and technical systems (e.g., automation, data fusion). As previously mentioned, this collection is directed by the IRs issued by the Commander and staff. “However, effective processing requires that operators do not merely collect data in response to the interpretation of an IR. From a decision support perspective, operators start with
specific operational requirements about current operational needs, but then must re-frame the original information request based on an understanding of the mission context and the intent behind the request (Roth et al., 2010).” The processing phase is complex because the framing and contextualization of IRs is so critical to the evolution of the collection environment. Framing and contextualization is conducted through identification, meta-data tagging, portion marking, and extraction of subsets of streams that may be relevant to the IR they have been issued, or by generating further questions during processing. Essentially, processing functions are a contextually-sensitive filtering operation, taking in massive volumes of data and outputting a relatively small subset of that data deemed relevant to address the IR (TAPED 16). Unfortunately, this filtering is extremely complex, particularly with the volume of information coming from today’s sensor platforms. As General Robert Kehler, Commander USSTRATCOM, mentioned in 2011, “We have to learn to better manage and make sense of the growing volume of data we collect and the intelligence we produce… So the good news is we are collecting 1500% more data than we did just five years ago… The bad news, we’re collecting 1500% more data than we did five years ago. And the worst news is, during the same time, our traditional PED capability has grown by about 30%.” Therefore, a shift from traditional processing techniques is necessary to maintain a high quality output (i.e., high signal-to-noise ratio) and avoid overtaxing and imposing enhanced cognitive burden on our limited analytical resources.

1.2. Exploitation

Exploitation involves analysts or automated algorithms refining processed data. The exploitation phase takes the information generated within the processing phase as input, and extracts or infers mission-specific details to directly address IRs and support the military decision making process (MDMP). Processed information is taken in, and interpretations of data relevance are made by exploiters based on the confirmation, resolution, or discovery of additional contextual information or mission demands. This highly evolving process primarily attempts to establish accurate frames of reference for the processed information (e.g., temporal, geospatial, contextual), and combines this with aspects of the data’s historical record (i.e., its pedigree). When all these factors are considered, a value chain of the data is prepared to be shipped out to consumers for immediate use and action. The exploitation phase is the most analytical part of the entire process, and requires that users are expert at analytics, and expert users of their tools. Exploitation often requires the user to not only think analytically about the supplied data, but also prepare information for understanding and use by others. As mentioned above, this often involves tagging input information as well as sending newly exploited data into the field. For example, during one of the authors’ KE trips to the Cyber Center For Excellence at Ft. Gordon, we observed the painstakingly long process of an analyst conducting Moving Target Identification (MTI), where they had to scrub through multiple hours of video of a single location both before and after the emplacement of a suspected improvised explosive device (IED), then capture relevant still image frames, marking the IED’s emplacement with red boxes. These images were then saved and annotated by the analyst, providing their first pass analysis on what they deemed occurred during this change, and then sent to the customer in this prepared form.

1.3. Dissemination

The dissemination phase completes the PED process. It packages the processed and exploited information into a format that is available and actionable for Commanders, operational teams, and other consumers. While this may seem trivial, it is actually one of the more complex tasks handled by PED analysts due to the rapidly evolving battlefield and strict time constraints. Consumers of information have constant shifts in location and information desires, and have limited cognitive resources, so they require actionable intelligence presented in a manner that is easily understood and not simply dumped into their workspace. Consumers may not know that they require a specific piece of information because it is outside of the scope they initially considered as relevant. This responsibility falls to the PED analyst—they must identify this information and frame it so that it integrates with consumers’ requested information and highlights its perceived relevance to a given mission objective. This task further burdens already overloaded PED analysts. In the dissemination phase, analysts fine-tune and tailor exports and reports for each mission and to each individual consumer. This is a currently a manual, labor-intensive process. For example, exporting chat logs between PED analysts, asset controllers, and mission stakeholders was one example provided to us by an overtasked PED analyst working in a stateside reachback facility. Exporting these logs is required to provide context and exploited information that is collected through collaboration between different PED contributors. For reachback analysts (and others) it remains a large part of the dissemination process, consuming significant resources. In some observed cases, analysts manually exported over twenty chat logs generated by a single mission analysis, consuming time that could have been spent crafting reports to ensure they contained the most reliable and recent information in a format that best supports action and efficiency in the MDMP.
2. **Current and Future PED**

2.1. **Current State of PED**

PED is arguably one of the essential pillars of intelligence collection, based on the Army’s current focus on developing ISR capabilities in support of PED. As multi-INT and multi-payload platforms are increasingly used in current operations (e.g., the Army’s MQ-1C Gray Eagle, the Air Force’s MQ-9 Reaper), they have not automatically reduced sensor operator workload or reduced manpower requirements. Instead, these technologies have required additional personnel to operate and, because they require operators to work at new intensities and new tempos of activity, they have created new complexities across MI systems. We observed these complexities through our various KE and analyst interaction sessions, with a particular focus on understanding the PED workflow and what factors contribute to situations of analyst overload. After interactions with PED users, we determined that modern PED can be seen as a prototypical illustration of a stretched system (Woods and Dekker, 2000; Woods and Hollnagel, 2006). In stretched systems, every system component is strained to operate at its capacity. When novel capabilities are introduced to stretched systems (e.g., a new automated capability), those improvements will be exploited to achieve a new intensity and tempo of activity. However, PED does not follow this definition of a stretched system. Instead, acceptance of novel capabilities has been slow, delaying any significant enhancements to efficiency or performance at the forefront of the PED process. As a result, analysts are struggling to use antiquated technologies to address increased task demands.

For example, we observed an analyst juggling communications with over twenty individual chat windows on a single screen display using a standard mIRC client (a technology developed in the mid-1990’s based on even older technology). This analyst was trying his best to coordinate across different parties, which led to significant inefficiencies. The result was a further strain on his cognitive and temporal resources as he struggled to keep pace with his current analysis. At the same time, he was being tasked with requirements for his next PED analysis. The analyst was simply unable to keep track of the several references to his mission in his various chat windows, and was completely unaided by any sort of automation that could alert or direct his attention. He spoke of instances where critical information, like his payload’s tail number (used to reference a given collection asset), was referenced and completely missed, missing opportunities for “called out” or limited window of opportunity information collection. Figure 1 is a high-level workflow diagram that shows many of the key tasks in the PED process. For system designers, it is apparent that across these critical tasks there is ample opportunity for automation support; however, for our own efforts, we focused on the need for improved collaboration. This example is just one of many cases of widespread inefficiencies in the PED process that result from a lack of modern collaboration technologies, technologies that should be designed with a deep understanding of the PED domain and the challenges it faces.
2.2. Future of PED

The PED landscape is currently at a crossroads as US forces begin to transition away from isolated attachments of region-specific forces to a globally connected network of force nodes. The vision for the future is one of constant connectivity and collaboration, removing cross-branch and cross-agency barriers that characterized the US military in the past. As with many data-driven domains, we believe that this future state of PED will require a complex balance between manual processing, human-computer interaction, and automation to manage expanding volumes of data. This requirement is combined with force structure changes that are resulting in reduced manpower and a smaller global footprint with command agencies trending towards a preference for a centralized cell that provides global PED reachback capabilities (e.g., the PED Center of Excellence at Fort Gordon). This strategy has been influenced by many factors, among which is the recognition that effective collaboration between different PED stakeholders is essential for the high volume, fast tempo, Multi-INT future landscape of PED. This networked PED is necessary to keep pace with the high-tempo global operations that US forces are now facing. Building a centralized PED infrastructure is critical for these operations, because it will enable collaboration and provide centralized system resources for automated processing, exploitation, and dissemination of the Multi-INT data produced by a global US force network.

While centralizing core PED analysts will facilitate face-to-face communications and enable more efficient resource management by PED supervisors, what we observed in our frequent visits to modern PED facilities is a new collaboration challenge that is arising as analysts and stakeholders are pulled out of region-specific PED cells. Ironically, many of the same collaboration issues now exist in a new format. With a reachback facility, PED analysts and payload operators must collaborate with remotely located payload managers and Commanders. The global distribution of the PED collection network creates collaboration issues that we believe are centered on the need to provide a shared frame of reference or context between different collaborators. Creating this shared frame is especially challenging as the effort to reduce PED footprints from a staffing perspective (resulting from budget cuts and force reallocations) requires PED contributors to manage multiple PED workflows in parallel—requiring that they maintain and update multiple frames of reference. The challenge for system designers who want to provide automated capabilities is that automation typically extracts humans from core processes that help them to better understand the context of a situation (i.e., the frame of reference). In the case of multiple distributed operators, if automation contributes to some small piece of a single contributor’s (or a reachback cell’s) workflow and artifacts, the resulting impact on shared frame of reference can rapidly propagate to other collaborators as well. If this significantly impacts collaborator productivity, the automated capabilities are likely to fail. This is why many systems already developed to aid PED analysts sit untouched, while analysts use antiquated, but proven...
technologies. While there is no doubt that automation has a significant role in both current and future PED workflows, it must be integrated into workflows so that it is adopted and enhances workflow efficiency and effectiveness. This integration is a challenging task that requires a strategic approach to the planning and development of novel system capabilities.

3. Automation to the Rescue

3.1. Automation Benefits to PED

The PED process has become one of the most important aspects of informing Commander intent and planning battlespace operations. However, as we’ve described, this emphasis on PED has created issues for the limited number of PED analysts involved in the daily activities of supporting the Intelligence Analysis (IA) process. Ever increasing amounts of data are being generated by new multi-INT sensor platforms and demands for analysis of this data are increasing, while the PED personnel footprint is the same or even reduced. For PED to continue to successfully generate valuable and timely intelligence with analytical rigor, automation is necessary to assist overburdened and overtasked analysts.

Many of the analytical tasks throughout the PED cycle lend themselves to automation, if it is employed and used correctly. Benefits of automation for PED mirror those in other domains, and include improved efficiency, reduced cognitive workload, attention direction, task prioritization, and enhanced precision when performing menial or repeated tasks. Examples of several typical PED tasks successfully automated in the past, which can easily be assisted by automation can be found in Table 3.

Table 3: PED tasks with potential for automation assistance

<table>
<thead>
<tr>
<th>Process</th>
<th>Task</th>
<th>Automation Assistance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Processing</td>
<td>Full motion video (FMV) tracking of Be On the Lookout (BOLO) targets</td>
<td>Software suites, such as AVAA, support single selection and automated tracking (reducing several hours of raw video to several minutes)</td>
</tr>
<tr>
<td></td>
<td>Still imagery image correlation</td>
<td>Software can find correlations and linkages between multiple still images</td>
</tr>
<tr>
<td></td>
<td>Remain current on IRs and Commander intent</td>
<td>Currently very cumbersome, analysts often work on stale or outdated IRs. Automation/alerting could provide cues for when IRs are updated</td>
</tr>
<tr>
<td></td>
<td>Fuse FMV, still imagery, change detection, spectral analysis, etc. from different sources and locations into one coherent picture</td>
<td>Automated tools could help gather linked intelligence from disparate sources</td>
</tr>
<tr>
<td>Exploitation</td>
<td>Tag and annotate FMV and still imagery for quick operational reference</td>
<td>Currently done manually. Automation could automatically tag and extend annotations throughout different artifacts</td>
</tr>
<tr>
<td></td>
<td>Access the information and mental models of other PED personnel</td>
<td>Currently done via chat/radio and through querying other analysts. Automation could find references to past data analysis for enhanced framing/contextualization</td>
</tr>
<tr>
<td></td>
<td>Communicate with customer and mission planners to collect the most relevant information</td>
<td>Currently done via monitoring public chat or radio frequencies. Automation/alerting could direct PED personnel to relevant communications to reduce cognitive load by directing attention to relevant information</td>
</tr>
<tr>
<td></td>
<td>Remain vigilant and aware to opportunistically exploit data discovered through non-traditional means</td>
<td>Automation could continuously track relevant information, and monitor “call-outs” within nearby named areas of interest (NAIs) that analysts may be too overtasked to notice</td>
</tr>
<tr>
<td></td>
<td>Maintain consistent artifacts and shared document versions between customers, platform operators, and collectors</td>
<td>Currently combine multiple different chat programs, email applications, and online references to stale or outdated artifacts. Automation could ensure version control</td>
</tr>
<tr>
<td>Dissemination</td>
<td>Generate communication logs</td>
<td>Currently done individually for over 20 chat logs per mission. Automation could bundle these into a single step, generating a common standard output</td>
</tr>
<tr>
<td></td>
<td>Maintain a list of interested customers and operational groups</td>
<td>Currently kept within the mental model of each individual. Automation could suggest relevant groups for dissemination, and maintain references to the individuals involved with each mission</td>
</tr>
</tbody>
</table>
While these may be prime examples of the need for automation in the PED process, they are currently not employed. Soldiers do not trust the automated algorithms and processes enough to fully integrate them into their workflow, especially seasoned analysts who have a set or pre-defined workflow. These automated techniques have not yet been smoothly integrated into the daily work environment of the PED analyst. When observing modern reachback PED cells, we found that analysts may use one or two of the algorithms or techniques shown in Table 3, but they have no central platform that grants access to all these techniques, nor one that combines them in a meaningful way. Some of the most simple and automation-friendly tasks (i.e., logging the over twenty chat logs after a mission is completed) have been overlooked by developers supporting PED, so solutions have yet to be developed. Instead, analysts find themselves juggling disparate links between these techniques, often preferring to accomplish the tasks by hand because the automated techniques are hard to access. PED analysts suffer from the same automation problems as other domains, including their lack of understanding of the automated techniques that have already been developed, the availability and usability of automated software to help with simple or repeated tasks (not developed), their lack of faith in these algorithms (creating mistrust), and lack of familiarity with the tools that can be used.

3.2. Trust in Automation

For any automated capability to succeed in the PED environment, analysts must trust that the capability will both improve their performance and consistently meet their expectations on what it will provide. As Mangio and Wilkinson (2008) point out, military intelligence is conducted on a scale where “the world is literally its province.” PED analysts face the difficult task of making sense of real-world situations that are continuously evolving, occurring in real time, or that have not yet occurred. Given the dynamic nature of the real world, to understand what circumstances can change an analyst’s attitude of trust towards an automated decision support system (DSS), we must identify what variables influence their trust and how the dynamics of the real world affect the respective states of those variables throughout the PED cycle. To identify these variables, a working definition of trust is necessary to frame the conversation. Many researchers have defined what exactly it means for a person to trust another entity. While most of these definitions provide related explanations of what trust means, many differ in their explanations of what trust actually is from a cognitive standpoint. For example, Lee and See (2004b) point out that trust has been characterized as an expectancy (Barber, 1983; Rempel, Holmes, and Zanna, 1985; Rotter, 1967), as a willingness to act (Johns, 1996; Kramer, 1999; Moorman, Deshpande, and Zaltman, 1993), and as a state of vulnerability (Deutsch, 1960; Mayer, Davis, and Schoorman, 1995), yet based on their review of previous models and definitions of trust, concluded that in terms of automated systems, trust is best thought of as “an attitude that an agent will help achieve an individual’s goals in a situation characterized by uncertainty and vulnerability” (Lee and See, 2004a). This concept of trust is important because it helps us understand how system designers can influence analysts’ trust in automated DSS and how that trust will influence whether a system’s automated capabilities can be successfully deployed to support the PED cycle.

Princeton’s WordNet (http://wordnetweb.princeton.edu) defines an attitude as “a complex mental state involving beliefs and feelings and values and dispositions to act in certain ways.” Similarly, Ajzen and Fishbein (1980; 1975) proposed a model for understanding and predicting human behavior comprised of beliefs and perceptions, attitudes, intentions, and behaviors. Their model proposed that our beliefs and perceptions form attitudes, which then influence our intentions to act. These intentions, combined with other attitudes and external factors, guide our behaviors. Applying this model, Lee and See (2004a) suggest that trust in automation is an attitude, based on operators’ beliefs and perceptions regarding different system characteristics. These beliefs and perceptions help the operator to form an attitude of trust towards the system. This attitude guides the formation of intentions, which guide the operator’s behaviors, among which is their choice to rely on a system’s automated capabilities. Of course, this is not to say that trust is the only factor influencing the intention to rely on an automated system, but it is one of several attitudes the operator holds which combine to form their intention to use automated capabilities, a critical requirement for the success of any automated solution (Lee and See, 2004a). Also, the intention to rely on the automation does not necessarily equate operator action. Rather, the intention to rely guides the operator’s behavior along with other factors, such as time constraints, workload, and self-confidence (Lee and See, 2004a; Lee and Moray, 1994). However, as trust in automation is one of the major contributing attitudes leading to reliance on novel automated capabilities (Cohen, Parasuraman, and Freeman, 1998; Danaher, 1980; Endsley, 1996; Parasuraman, Molloy, and Singh, 1993; Wiener, 1985), having an appropriate attitude of trust is critical to the success of both complex and straightforward automation capabilities.

While many system attributes (e.g., reliability, performance) are known to influence trust in automation, it is important to note that it is not the actual state of these attributes that influences trust, but rather the perceived state of
these attributes (which may not align with the true automation capabilities). Atayan et al. (2006) and Lee and See (2004a) provide three different classes of cognitive processes that use these perceptions to form attitudes of trust: (1) analytic processes; (2) analogical processes; and (3) affective processes. Analytic processes are based on deliberate consideration of available system information regarding key attributes that are identified based on an operator’s knowledge, experience, and mental models of a system (Atayan et al., 2006). They demand the most cognitive resources, but are also the most likely to provide an accurate understanding of the system’s true trustworthiness (Atayan et al., 2006). In an analogical process, an analyst has a stored rule or procedure for determining if the system is behaving in a trustworthy manner; however, while the rule or procedure can be based on perceived system information, it can also be based on factors such as reputation or third-party information (Miller, 2005). Affective processes influence trust on the basis of emotional impressions or feelings (Atayan et al., 2006). The attitude of trust is influenced by the analyst’s stereotypes or assumptions, which are based on the affective impressions from prior encounters with the system or even from stereotypes held towards technology or automated systems in general (Atayan et al., 2006; Tseng and Fogg, 1999). So in the case of Arjen and Fishbein’s (1980) framework, analytic and analogical processes based on system information represent analyst perceptions of the DSS, while affective and analogical processes based on stereotypes and assumptions represent analyst beliefs towards the DSS. Interestingly, both Miller (2005) and Slovic et al. (2004) provide evidence showing that affective processes and analogical processes are often more important in human interactions with automation than analytical processes, especially in the case of new or complex systems, a critical finding to consider when designing and developing novel automated solutions to support the PED or other complex workflows.

There is a large body of literature that establishes relevant system attributes that have been empirically shown to influence analyst trust in automated systems. Prior efforts have also identified factors that can influence an analyst’s initial trust (i.e., faith) in the system, before any first-hand experience. Initial trust levels are especially important when training new analysts on these systems. Table 1 summarizes the factors that contribute to an analyst’s attitude of trust towards an automated system. The table presents both perceived system characteristics and characteristics that influence analyst faith empirically shown to influence attitudes of trust. The one exception is system availability, which we included because we assumed that if an analyst believes that a system (or given capability) is not available, it affects their trust that the system can perform as expected (and be accessed or relied upon in the future). System designers who want to augment the PED cycle with automated capabilities should design to include these attributes; however, creating an automated solution based only on these attributes does not guarantee adoption by PED analysts.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Evidence</th>
</tr>
</thead>
<tbody>
<tr>
<td>Understandability</td>
<td>SA – Extent to which a system supports the formation of veridical mental models regarding its processes, capabilities, utility, etc.</td>
<td>(Llinas, Bisantz, Drury et al., 1998)</td>
</tr>
<tr>
<td>Predictability</td>
<td>SA – How regularly system actions are performed in the expected manner</td>
<td>(Lee and Moray, 1992; Merritt and Ilgen, 2008; Lerch, Prietula, and Kulik, 1997; Master, Jiang, Khasawneh et al., 2005; Muir and Moray, 1996)</td>
</tr>
<tr>
<td>Familiarity</td>
<td>SA – Degree to which a system employs procedures, terms, and cultural norms which are familiar, friendly, and natural to the operator</td>
<td>(Llinas, Bisantz, Drury et al., 1998; Jian, Bisantz, and Drury, 2000; Safar and Turner, 2005; Sheridan, 1989; Master, Gramopadhye, Bingham et al., 2000; Parasuraman and Miller, 2004; Spain and Madhavan, 2009)</td>
</tr>
<tr>
<td>Expiication of Intention</td>
<td>SA – Degree to which the system explicitly communicates that it will act in a particular way</td>
<td>(Llinas, Bisantz, Drury et al., 1998; Sheridan, 1989)</td>
</tr>
<tr>
<td>Usability</td>
<td>SA – Extent to which the system is effective to use, efficient to use, safe to use, easy to learn, and easy to remember how to use</td>
<td>(Muir and Moray, 1996; Parasuraman and Miller, 2004; Spain and Madhavan, 2009; Rogers, Sharp, and Preece, 2011; Kim and Moon, 1998; Briggs, Burford, and Dracup, 1998)</td>
</tr>
<tr>
<td>Competence</td>
<td>SA – Extent to which the system possesses the requisite or adequate abilities or qualities to perform a given task/assignment effectively</td>
<td>(Merritt and Ilgen, 2008; Master, Jiang, Khasawneh et al., 2005; Muir and Moray, 1996; Madsen and Gregor, 2000)</td>
</tr>
<tr>
<td>Reliability</td>
<td>SA – Extent to which the system is free of errors</td>
<td>(Llinas, Bisantz, Drury et al., 1998; Lee and Moray, 1992; Master, Jiang, Khasawneh et al., 2005; Muir and Moray, 1996; Madsen and Gregor, 2000; Jian, Bisantz, and Drury, 2000; Safar and Turner, 2005)</td>
</tr>
</tbody>
</table>
For any change in analyst trust to occur, the analyst must first decide to rely on the automated system, with the only exceptions being information received by the analyst regarding the system’s capabilities or reputation from third-party channels or general assumptions and stereotypes that affect faith in the system changing due to new knowledge or experiences unrelated to the system. The decision to rely on an automated capability arises from situations where the analyst’s reliance threshold (i.e., the point at which they decide to rely on an automated capability based on their system perception, workload, perceived risks, etc.) for the automated system is exceeded by the combination of changes in their trust in the system, self-confidence, and trust in alternative options (e.g., manual task completion, other automated capabilities). Consequently, any circumstances that decrease the analyst’s self-confidence or their trust in alternative options creates the potential for this prerequisite to be met and in turn for trust in the automated capability of interest to evolve through decisions to rely on those capabilities. As previously discussed, real-world dynamics can create these circumstances due to changes to overall task uncertainty, which results from changes to data sets, problem structure and/or organizational uncertainty, or availability of alternative options.

When the decision to rely on automation is made, the circumstance that creates the potential for analyst trust to change is one where there is a difference in the system’s perceived current capabilities versus the perceived capabilities of the system during previous recent states of reliance. These differences can have multiple causes. For example, the transparency of the system can influence the ability of the analyst to accurately perceive true system capabilities. Real-world dynamics can influence this transparency by changing factors such as the availability of pedigree information that can be displayed, the availability of system resources needed to display the information that produces system transparency, the conditions of the operating environment (e.g., a blackout), etc. Another circumstance that can cause these differences is real-world events/changes that have occurred since the previous state of reliance that change either the degree of uncertainty characterizing the task or the availability of the system or its features. These changes in availability and/or the degree of uncertainty of the task can result from any number of real-world circumstances that arise. For example, the discovery of a new source of information can: (1) add to the data set uncertainty by increasing complexity, volume, heterogeneity, and/or uncertainty; (2) expand the breadth of focus and required domains of expertise, adding to the uncertainty of many PED problem structures; and (3) shift the risk of providing incorrect recommendations or reduce the time to provide operationally relevant intelligence.

| Dependability | SA – Degree to which the system is available and working to complete commitments | (Linas, Bisantz, Drury et al., 1998; Lee and Moray, 1992; Merritt and Ilgen, 2008; Master, Jiang, Khasawneh et al., 2005; Muir and Moray, 1996; Jian, Bisantz, and Drury, 2000; Safar and Turner, 2005) |
| Robustness | SA – Degree to which the system can perform well under varying conditions | (Linas, Bisantz, Drury et al., 1998) |
| Usefulness | SA – Degree of utility of the system capabilities | (Linas, Bisantz, Drury et al., 1998) |
| Confidence | SA – Extent to which the system can provide a confidence assessment to analysts | (Jian, Bisantz, and Drury, 2000; Safar and Turner, 2005) |
| System Availability | SA – Degree to which they system is available and working at a given moment in time |
| Faith | AA – Strong belief and trust in and loyalty to a particular conviction for which there may be no proof | (Lee and Moray, 1992; Master, Jiang, Khasawneh et al., 2005; Muir and Moray, 1996; Madsen and Gregor, 2000) |
| Background Knowledge | AA – Previous knowledge used to form stereotypes or assumptions about the trustworthiness of the system | (Atoyan et al., 2006; Tseng and Fogg, 1999) |
| Personal Attachment | AA – Characterized by the strength of a bond, as of affection or loyalty or fond regard for the system | (Madsen and Gregor, 2000) |
| Personality | AA – Combination of personal attributes—behavioral, temperamental, emotional and mental—that motivate an analyst’s beliefs towards an automated system; | (Merritt and Ilgen, 2008; Kelly, Boardman, Goilau et al., 2003; Lee and See, 2004a) |
| Extraversion | AA – Characterized by tendency to be sociable, assertive, and dominant | (Merritt and Ilgen, 2008) |
| Propensity to trust machines | AA – Characterized by a stable, trait-like tendency to trust or not trust technology/automated systems | (Merritt and Ilgen, 2008) |

**Abbreviations:** SA = System Attribute, AA = Analyst Attribute
adding to the organizational uncertainty. All of these changes affect overall task uncertainty, and can therefore affect the capabilities of the automated system to perform its job in a way that meets analyst expectations.

Other circumstances that can shift analyst trust are situations where the analyst relies on the automated system and it meets/exceeds expected capabilities (or fails to meet its expected capabilities), causing the analyst to update their expectations of various system capabilities as system feedback is made available to allow the analyst to self-assess the appropriateness of his/her decision to rely on the system. In these circumstances, Lee and Moray (1992, 1994) found that only the most recent experiences relying on the system affect trust. Therefore, first-hand experience with the potential to change trust has an unspecified rate of decay; without feedback on the appropriateness of relying on a system, perceptions have a diminishing potential to influence trust. If feedback is received, the probability of trust changing has an inverse relationship to the duration of time that has passed since relying on the system and receiving the feedback. Real-world dynamics influence when feedback can be provided (e.g., the system’s result can only be assessed after a milestone event occurs in the world).

Another circumstance that creates differences in perceived system capabilities at different times of reliance is changes in the analyst’s ability to follow system cues to diagnose its true capability. For example, over the course of an analysis (or a career), an analyst may receive training on the system or the system may receive a software update that makes the task of selecting the appropriate cues easier for the analyst. Experience over time helps the analyst develop his ability to accurately perceive the true system capabilities; however, if system capabilities do not match analyst expectations, then, with each interaction, trust can decrease until the analyst abandons the system.

These examples describe different real-world circumstances that can change the attitude of trust towards an automated system. Returning to Mangio and Wilinson’s (2008) original characterization of the scale of analysis—“the world is literally its province”—the number of real-world variables that can change over the course of an analysis is infinite. The degree to which these variables affect the ability of an automated capability to perform as expected is the critical relationship that will influence analysts’ attitude of trust in that capability. Therefore, system designers must design automated systems that not only facilitate the initial establishment of trust, but also provide system feedback elements that can help maintain trust over time through an appropriate man-machine dialogue.

3.3. Trust and the Man-Machine Dialogue

Trust is a dynamic relationship. The state of trust at any given time is determined by the positive and negative real-world experiences and observations that occur over the course of a relationship between two actors, such as an operator and an automated system (Falcone and Castelfranchi, 2004; Gao and Lee, 2006a,b; Jonker, Schalken, Theeuwes et al., 2004; Lee and See, 2004a). The experiences that comprise the operator-system dialogue occur when the operator relies on the automated system to perform a task. When the operator relies on the automated system, trust either increases or stays the same (if the experience is positive), or it decreases or stays the same (if the experience is negative) (Jonker, Schalken, Theeuwes et al., 2004). Negative experiences have a stronger negative effect on trust than the positive effect of positive experiences. To judge an experience as positive or negative, there must be a preexisting reference point that the experience is judged against. Therefore, the trust that an operator has in a system generates an expectation of the system’s capability to perform a task. This expectation is the reference point for comparison. The results of the comparison generate the perception that the system performed as expected or better than expected (a positive experience), or performed worse than expected (a negative experience). Over the course of the operator-system relationship, trust evolves as the outcome of this series of confirming observations (Jonker and Treur, 1999).

Within the PED environment, analyst perceptions of system performance are based on two different sources of information. The first source is feedback from third-party sources, including: (1) other analysts who use the system; (2) other systems (including manual execution of a task) performing the same or related tasks; and (3) feedback from consumers or supervisors on the product created with the system’s output. In the case of system feedback, the perceptions of the primary system’s capability to perform a task can change before, during, or after the task. For example, let’s say you enter a destination into a GPS navigation system and rely on it to provide the shortest route a your friend’s house. Before starting your trip, you could verify the route with a map. During your trip, you could check another GPS system to validate that the route is the shortest. After you arrive, you could check with your friend to see if they thought your route was the shortest. Regardless of when you receive this feedback, if the results match those of your GPS system, then your perception that the system will perform as expected will stay the same or improve; however, if they do not match (and you believe the alternatives are more trustworthy), then your perception of the system’s capabilities do not improve and may degrade (Jonker, Schalken, Theeuwes et al., 2004).

Third-party sources can also provide information without analyst initiation. For example, in the GPS example, you could arrive at your friend’s house, and they may ask what took you so long and describe a shorter route. For PED analysts, this situation can arise if a consumer of the disseminated product tells the analyst that their information was incorrect (e.g., “the target was not where you said it would be when we checked”), or that they missed a relevant time window (e.g., the target they correctly identified escaped in a truck because the analysis took
too long). If this error was based all or in part on the analyst’s decision to rely on the automated system, then similar to the GPS scenario, the mismatch of analyst expectations and actual performance creates a negative experience. Therefore, system designers should create system outputs that meet the expectations of automation consumers. This will facilitate positive attitudes of trust, increased reliance, and more appropriately calibrated system expectations.

The second source of information that can influence operator expectations of system performance is the dialogue between the analyst and the automated system. The effectiveness of the operator-system dialogue is determined by how effectively and efficiently the system communicates its intentions, capabilities, and decisions during processing, and how targeted that communication is toward the goal the operator is trying to accomplish with the system. An effective operator-system dialogue dynamically updates the operator’s attitude of trust towards the system, so that proper expectations of the system’s capabilities can be applied when deciding to rely on the system’s output. To facilitate this goal, Weir (1989a) suggests that automated systems must support three conceptual layers of dialogue between the operator and system: a control flow layer, a display manipulation layer, and a meta-dialogue layer.

The control flow layer of the dialogue lets the operator affect the course of operation, which affects their expectations of the automated system (Weir, 1989a). It is the primary dialogue that takes place in most man-machine systems because it is the dialogue that supports the interaction cycle, shown in Figure 2, that allows the operator to guide the system’s processing to meet their goals. To support this function, the system interface must provide opportunities for the operator to input control actions (Weir, 1989b). Therefore, this layer of the operator-system dialogue includes operator control actions, system status/state information that helps the operator understand which actions are available, and feedback on the results of the operator’s actions (Weir, 1992). This dialogue is likely to provide the operator with basic information cues that will influence the operator’s perceptions of the system’s capabilities, influencing their level of trust in the system.

![Figure 2: Control flow dialogue cycle](image)

How this specific layer of the dialogue affects trust or operator perceptions is not described in the literature, but we believe that this dialogue supports operator perceptions of system robustness and usefulness, which have the potential to influence trust. The state of these two system attributes can be communicated by the system options available to the operator. For example, a GPS navigation system could present the user with options to search for a destination, enter a destination by known address, select a destination from a map. The availability of these features enhances the user’s perception of the system’s overall usefulness when entering a destination, and increases their knowledge of the underlying system. Therefore, system designers should make available capabilities transparent to the user. The robustness of the system can be communicated by a feature, such as user-activated options that allow the system to adapt to unexpected situations. In the GPS example, dynamic routing suggests a U-turn or alternative routing when the user goes off course, which communicates that the system can perform under varying conditions if the planned process does not go as the system expects. As the user experiences these features, their perceptions of system capabilities and attitude of trust towards the system evolves.

The display manipulation layer of operator-system dialogue is desirable, but not always present in automated systems (Weir, 1989a). This layer of the dialogue, shown in Figure 3, allows the operator to control how information is displayed in the control flow layer and the meta-dialogue layer.
This layer allows the operator to learn which display components can be customized to the operator’s preferences. For example, most GPS navigation systems have settings to change the visual display of the route for daytime and nighttime driving. This setting allows the user to alter the control flow dialogue so that it is more effective, familiar, and usable. By providing the ability to customize the display, designers increase the operator’s perception of system capabilities, thereby increasing trust in the system. The display manipulation dialogue can also influence attitudes of trust another way: The visual design of system interfaces has been shown to influence affective impressions (Fogg and Tseng, 1999) and affective impressions have been shown to mediate attitudes of trust early in the operator-system relationship (Parasuraman and Miller, 2004; Spain and Madhavan, 2009). Therefore, the display manipulation layer has the potential to greatly influence trust in the important initial stage of trust evolution (Merritt and Ilgen, 2008).

The third layer of operator-system dialogue recommended by Weir (1989a) is the meta-dialogue layer. This layer does not directly affect the control flow of the system; but it is related to the control flow dialogue because it provides support information for control or for relevant aspects of the system’s capabilities to achieve different actions (i.e., self-health) (Weir, 1989b). The meta-dialogue layer can provide information such as help features, event logs, or confidence that different processes will operate as expected, framing the dialog and operator expectations user. Figure 4 shows the framework of the meta-dialogue.

Unlike the control flow and display manipulation layers, the framework for the meta-dialogue is not a straightforward cycle. The meta-dialogue is more of a one-way system-to-operator report on self-health. Although this information is critical to the analytical and analogical processes that drive trust during the history-based stage of trust evolution (Merritt and Ilgen, 2008), it is often missing due to the difficulty in deriving this information, or inaccessible because system designers provide it in an expert interface to support advanced troubleshooting. For example, modern automobiles have electronic diagnostic systems that monitor the vehicle to help mechanics diagnose and solve common problems. Despite the existence of this meta-knowledge of the system’s state and capabilities, when an issue arises that threatens to alter the vehicle’s capabilities, the only line of communication to the vehicle’s owner is the check engine light, which communicates no information on the type or severity of the problem—it communicates only that something is wrong and they should seek out expert assistance. This interaction was likely designed this way based on the design team’s assumption that the driver-automobile dialogue falls within the novice-to-expert class of assistance meta-dialogues: the diagnostic system outputs numerical fault codes useful to experts, but unsuitable for communicating the type and severity of the problem to the driver.
Meta-dialogues can provide precise or best-estimate cues for the user to perceive the state of different system attributes, such as reliability, explanation of intentions, dependability, and confidence—all of which influence operator attitudes of trust. Within the PED domain, there is no literature available on operational systems that employ this level of dialogue; however there is evidence that connects displaying this information with facilitating appropriate levels of operator trust in automated systems (Antifakos, Kern, Schiele et al., 2005; Ceruti, Das, Ashenfelter et al., 2006; Liu and Williams, 2002).

As Weir (1989a, 1992) states and others have validated (Alty and Johannsen, 1989), designing system dialogues that support all three layers yields a more flexible and supportive user interface. When systems implement these three layers of dialogue, they create appropriate perceptions of a greater number of key system attributes, as well as calibrate and generate temporal specificity of the operator’s attitude of trust [24]. For example, Table 2 presents the three dialogue layers and the system attributes known to influence attitudes of trust that can be supported through effective dialogue design. We can hypothesize, based on the breakdown of attributes, that the meta-dialogue layer will provide the most benefit to operators in terms of trust tuning because it can affect the largest group of key attributes and because it is the only dialogue layer that can directly affect the perception of reliability, which is one of the most influential factors in mediating attitudes of trust (Brown and Galster, 2004; Dzindolet, Peterson, Pomranky et al., 2003; Madhavan and Wiegmann, 2007; Parasuraman, 2000; Parasuraman and Riley, 1997; Wickens and Xu, 2002).

<table>
<thead>
<tr>
<th>Dialogue Layer</th>
<th>System Attribute Supported</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flow Control</td>
<td>Competence</td>
</tr>
<tr>
<td></td>
<td>Availabilty</td>
</tr>
<tr>
<td></td>
<td>Usefulness</td>
</tr>
<tr>
<td></td>
<td>Robustness</td>
</tr>
<tr>
<td>Display Manipulation</td>
<td>Understandability</td>
</tr>
<tr>
<td></td>
<td>Familiarity</td>
</tr>
<tr>
<td></td>
<td>Usability</td>
</tr>
<tr>
<td>Meta-Discourse</td>
<td>Understandability</td>
</tr>
<tr>
<td></td>
<td>Competence</td>
</tr>
<tr>
<td></td>
<td>Reliability</td>
</tr>
<tr>
<td></td>
<td>Dependability</td>
</tr>
<tr>
<td></td>
<td>Explicitness of Intention</td>
</tr>
<tr>
<td></td>
<td>Confidence</td>
</tr>
</tbody>
</table>

While supporting all three of the dialogue layers to foster appropriate trust dynamics is an ideal, actually designing and implementing an effective dialogue which accomplishes this goal is not a straightforward endeavor. How effectively the system dialogue facilitates appropriate attitudes of analyst or operator trust can be evaluated by its ability to accurately communicate the system’s true capabilities. Using Brunswik’s Lens Model (Cooksey and Freebody, 1985; Bisantz, Kirlik, Gay et al., 2000; Bisantz and Pritchett, 2003) as a framework, Bisantz and Seong (2001) and Llinas et al. (1998) illustrated how the perceived capabilities do not always match the true capabilities and how this relationship is a factor of the availability, selection, and diagnosticity of the cues the system provides. Applying the three-layer framework for an operator-system dialogue increases the availability of cues to the operator by providing a more robust set of information that is representative of system health. When displaying highly diagnostic cues, the ability of the system to report its self-health (i.e., its ability to perform as designed and expected) via the meta-dialogue layer provides a direct channel to communicate highly diagnostic system capability cues to operators.

4. Automating PED

4.1. Designing Successful Automation

As the previous sections discussed, for an automated system to facilitate trust, it must effectively communicate its ability to perform as designed and expected. While this may seem like a simple requirement to system designers, effectively building a system that can proactively recognize system shifts and effectively communicate self-health in a timely manner to calibrate operator expectations is a significant challenge. For the PED domain, this issue is compounded by the reluctance of PED analysts to deviate from existing proven workflows and technologies. This reluctance is understandable given that they have minimal resources to learn nuances of new systems and adapt their procedures during mission critical operations. Therefore, we established a series of guidelines for system developers that want to develop successful automated (or other technology-driven) capabilities to assist analysts in the PED or similarly characterized domains.
The underlying assumption with these guidelines is that successfully deployment of novel, technology-driven capabilities relies on the establishment and maintenance of trust in the introduced system’s capabilities. To establish and maintain this trust, analysts must decide to rely on the new capabilities so they can benefit from the various feedback elements afforded by supporting the different man-machine dialogue layers discussed in Section 3.3. Consequently, the guidelines provided in Table 4 are intended to be applied iteratively, initially supporting decisions to rely on the automated capabilities, creating an initial attitude of trust, and then incorporating the feedback dialogue layers to appropriately calibrate and maintain this trust to motivate continued reliance on system capabilities. This approach benefits system designers because it encourages rapid deployment of simple automated capabilities before the creation of more complex capabilities that may have a greater degree of performance variance (based on factors such as the quality of inputs made available to the system).

<table>
<thead>
<tr>
<th>ID</th>
<th>Recommendation</th>
<th>Justification</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Integrate with existing workflows and workflow support systems</td>
<td>Analysts’ reliance on existing tools and systems is unlikely to change given the high operational tempo and their constrained resources. Without buffers that enable analysts to experiment with new tools and augmented workflows, it is unlikely new technologies will succeed over existing, proven systems and practices. By integrating with existing systems and workflows, analysts can rely on novel automation capabilities to start calibrating their expectations of how those capabilities can enhance their productivity, leading to the establishment of an initial attitude of trust.</td>
</tr>
<tr>
<td>2</td>
<td>Focus initial system capabilities on low-level automation</td>
<td>The performance variability of an automated capability usually depends on the variability of the inputs it must act upon. If inputs change consistently or have questionable pedigrees, system performance is likely to suffer. For new systems being deployed to the PED domain, if performance is highly variable, then it will be harder for analysts to appropriately calibrate their expectations for a given system interaction. This creates the potential for inappropriate attitudes of trust that can lead to further mismatches in system performance and expectations, or analysts deciding not to rely on the capabilities at all. Instead, novel systems should deploy a set of low-level automated capabilities that do not require inputs with varying degrees of reliability. These capabilities often target highly redundant tasks. This approach facilitates appropriate calibration of performance expectations (given that performance is unlikely to change) and fosters initial establishment and ongoing maintenance of positive attitudes of trust in the system. This initial, calibrated attitude of trust then serves as a foundation for deploying more volatile automation capabilities; system designers can leverage the existing trust to encourage reliance on new system features which foster appropriately calibrated expectations, preventing abandonment of automation when expectations are not met.</td>
</tr>
<tr>
<td>3</td>
<td>Maintain transparency in automation</td>
<td>This is a standard guideline for any automated system. For analysts trying to maintain context over multiple, parallel analyses, assigning data processing or workflow tasks to automated capabilities will create a degree of separation for analysts and degrade their frame of reference unless automated processes are made transparent.</td>
</tr>
<tr>
<td>4</td>
<td>Use an ongoing dialogue to calibrate expectations</td>
<td>Similar to guideline 3, automated capabilities should maintain an ongoing dialogue (based on the foundations discussed in Section 3.3) to foster appropriately calibrated expectations of system performance. This guideline ensures that attitudes of trust do not degrade when analysts make the decision to rely on automated capabilities.</td>
</tr>
<tr>
<td>5</td>
<td>Do not force reliance</td>
<td>Related to the guideline 1, integrating any new capability into the PED workflow and forcing reliance is likely to result in an extreme prejudice towards the system if it fails in any way to meet analyst expectations. Forcing analysts to rely on automation and abandon their already proven and trusted systems will create the potential for analysts to enter into system reliance with a negative view of the system and a significant desire to validate the system as faulty or not meeting requirements to justify reverting to their trusted systems. Instead, novel capabilities should be provided as options, enabling analysts to depend on capabilities when they see the benefit for the capabilities. Providing this choice causes analysts to enter the interaction with positive expectations for capabilities to benefit their own effectiveness, removing the bias from situations of reliance and improving (with properly calibrated expectations) the likelihood that analysts will trust the system.</td>
</tr>
</tbody>
</table>

### 4.2. PED-X Collaborate Work Environment

Our most recent development work for the evolving PED community focused on bolstering the effectiveness of the typical PED workflow, analyzing their process, and ensuring our development coincided with current practices. Guided by the first requirement listed in Table 4, we were careful to integrate within this space, instead of attempting to replace or modify current practices. With this strategy, we increased the usability and accessibility of automation, which through effective human-computer dialogue encouraged analysts to incorporate these practices in their daily activities.

Through our numerous Knowledge Elicitation sessions with both reachback analysts and PED training experts, we identified the central pillar of the PED cycle to be communication, with a particular focus on chat-based collaboration. Chat is the main medium by which information requirements (IRs) are communicated, requirements for collection and exploitation are updated, and stakeholders are informed of analysis results. However, automation
and new technologies are minimally employed, with PED personnel continuing to rely on antiquated chat clients (Figure 5) that can easily become overwhelming even when there are only three or four chats occurring simultaneously.

**Figure 5: Overloaded mIRC chat window**

By integrating existing automated tools with chat platforms similar to the chat clients that analysts are already familiar with, and developing automated alerting and attention direction systems for integration within these collaborative chat environments, we hope to raise awareness and acceptance of these tools, increasing familiarity and daily use, especially since we restricted our innovations to integrated smoothly within their current workflow. Figure 6 shows a brief overview of our PED-X collaborative work environment, which significantly enhances the standard chat employed within the military today, while still maintaining a level of comfort and familiarity for the user by not significantly deviating from existing interfaces. Its graphical user interface (GUI) is modeled after the many existing email/chat clients in use today, including Gmail, Outlook, and Yahoo! Mail. We hope to enhance trust in our system and its extensive library of automated technologies by starting with this level of familiarity.

**Figure 6: Overview of PEDX collaborative chat software**

While an enhanced chat client may be needed, the automated tools that new technologies and recent developments can provide address the greatest needs within the PED domain. Through our PED-X software, we designed automated tools that fit seamlessly within the typical interactions analysts have with their chat environments. For example, small changes, such as time-stamping and dating, provide extensive benefits later in the analysts’ process, or when they are referencing past information. Through our KE sessions with users, we found that
This very small piece of information is often not included in historical logs, or is only partially included (e.g., only times, no dates), rendering the entire historical record useless. Therefore, our PED-X system includes this information at every interaction. It provides temporal information when information is shared, updated, or focused upon. We believe that automated support, provided at this low level and developed within an existing process (following the second guideline of focusing initial automation on simple support) can begin to foster trust in our system, allowing more complex automation to be successfully adopted as it is developed and deployed.

A more advanced collaborative chat system requires more advanced automation, but we tailored our implementation to remain familiar and unobtrusive to users to foster trust. Perhaps the single largest complaint we received during our KE sessions was the sheer amount of conversations they have to maintain during a mission, and the lack of support provided by their communication systems. Analysts are constantly forced to remain vigilant on over twenty different chat windows during a mission, causing them to miss important or critical information as it scrolls by undetected. Automation and attention direction can solve this problem, and we provided this solution within PEDX. Consider the conversations occurring in Figure 5—none of the chat windows indicate that they require the attention of the analyst. The windows all look identical, regardless of the importance of the conversation they contain. In PEDX, as shown in Figure 6, some conversations are automatically shown in bold to indicate their “unread” state. The user can intuit from this cue that the window requires their attention because there are elements of the conversation they have not yet seen. Once viewed, the boldface disappears. While this level of automation may seem useful, we followed the fifth guideline in Table 4 when we designed this feature to make this cue unobtrusive and ignorable, if analysts choose not to use it. We followed the fourth guideline to carry this feature through the PED-X system, applying it to the automation window. We also provided unread indicators within conversations, using color to indicate level of importance. Figure 7 shows a conversation with a single subthread, with bold text showing it is unread, and a green icon displaying the number of unread messages within the subthread. The color green indicates this subthread is not particularly important. We plan to develop this feature so it can be customized by analysts.

![Figure 7: Unread messages within threads](image)

More complex examples of simple automation techniques are implemented within a robust attachment system built directly into our PED-X toolkit, which while changing their workflow slightly, still considers the guidelines described in Table 4. While certain chat software currently in use may support basic sharing of attachments, most analysts communicate the mission artifacts necessary for their tasks through external email clients. This requires that they maintain relationships between conversations and email attachments, increasing their cognitive burden and creating more versioning issues. Within PED-X, we support the direct attachment of any file type directly within the chat window. As shown in Figure 8, this attachment system is intuitive and has a standard drag-and-drop interface familiar to most analysts. By providing this simple automation, we remove the need for analysts to track distal relationships between attachments and conversations, saving them time and increasing the accuracy of their analysis.
Figure 8: Drag-and-drop PED-X attachment system

Once trust is established through the initial simpler layer of automation described previously (guideline three from Table 4), more complex automated techniques can be deployed to reduce PED analyst workload. We have several proof-of-concept designs that highlight such techniques, including tools that can listen, ingest, and participate in conversations as if they were a human participant. These robots, as we have named them, can be optionally included or left out of conversations, in accordance with the fifth guideline and to ensure analysts are not having automation forced on their workflow, and these robots can have varying functionality. For example, two robots we designed and demonstrated within our first development cycle are a location/weather robot, as well as an acronym defining robot. The location/weather robot quietly listens to entire conversations, identifying text that may contain geolocations. Once identified, as shown in Figure 9, the robot can automatically include either a map of the mentioned region, or the current and projected weather for the region, so all users are aware of exactly the precise location being referenced. The input that triggers the automated robot, in this case a geolocation, is highlighted to ensure transparency in the automation process, showing the analyst exactly what triggered the robot’s response.

Figure 9: Automated inclusion of mapping information within conversations

This level of autonomous interaction is also provided by the acronym defining robot. During a discussion with an analyst, the various and multiple different uses of acronyms, particularly across different hierarchies of the military and non-military customers, was identified and referenced specifically as a point of confusion. We solved this problem by creating a robot that can identify acronyms and provide a list of definitions for users to select from and share with other users to ensure a common frame of reference.

Figure 10: Acronym definition robot

4.3. Early Evaluations

Through multiple opportunities for engagement with actual PED analysts we effectively tested some of our proof-of-concept automation techniques and workflow integration ideas. These engagement opportunities involved
multiple trips to observe smaller special operations forces operating out of Ft. Huachuca, as well as extensive visits
to the Cyber Center for Excellence at Ft. Gordon. These trips, particularly the latter, provided us with direct
communication and access to over forty current PED analysts, instructors, supervisors, and even the commanders
that regulate the entire PED cycle through their IRs. Carefully starting from small automated assistance including
alerting and attention direction techniques, we introduced a newly designed chat client with similar features to chat
clients used daily by analysts. We augmented this familiar interface with alerts and other changes, ranging from
simple “unread message” boling techniques to pop-up alerts. Overall, these technologies were easily accepted by
the analysts, instructors, and Commanders we interacted with, since they seamlessly integrated into their workflow.
We then received input for additional areas that human-computer interaction and automation could help.

We next presented our advanced proof-of-concept automation tools designed to smoothly integrate with the PED
process. These tools included a robust attachment system embedded directly within their chat client to maintain
version control, several collaborative mapping and weather tools that were integrated alongside chat, and enhanced
tagging/annotation abilities for still imagery. These were praised, particularly by the supervisors and Commanders,
who daily demand the results afforded by these technologies from their analysts. We found analysts were relaxed by
the familiar interfaces, and therefore more trusting in the information provided by the automation. Additional
feedback and feature requests for future development were generated from these demonstrations. For example, while
it may seem like an obvious feature, the ability to share an image with another user with an automatically overlaid
grid was requested by multiple analysts independently during our interactions, and will be developed during
subsequent development periods.

5. Conclusions

This paper provided an overview of our attempts to integrate novel automated capabilities into existing and
evolving PED workflows. We presented guidelines for successfully deploying new technology to PED (and
similarly characterized domains) which were centered on facilitating the initial establishment and ongoing
maintenance of trust to motivate appropriate decisions to rely on new capabilities. Our strategy focused on building
initial trust through low-level automation that integrates with existing workflows and systems and maintains
consistent performance by operating without the need for highly variable system inputs. This strategy reduced the
need for analysts to frequently calibrate system performance expectations, helping the system to meet expectations.
Once initial trust is established, it can be leveraged to support deployment of more complex automation with a
higher degree of performance variability. The initial attitude of trust serves as a buffer that will encourage decisions
to rely on the automated capabilities, so expectations can be properly calibrated and appropriate decisions to rely on
automation can be made. We based this strategy on our frequent engagements and observations with members of the
PED community and leveraged these opportunities to begin validating our initial prototype PED Collaborative Work
Environment. We presented an overview of many features of the Collaborative Work Environment and described
the initial feedback we received from representative users.

Our next steps include formally documenting the our domain understanding by applying a modeling formalism,
such as a work domain analysis (Naikar, Hopcroft, and Moylan, 2005), cognitive task analysis (Patterson, Woods,
Tinapple et al., 2001), or functional resonance analysis model (Hollnagel, 2012). We will continue to refine and
harden our prototype Collaborative Work Environment through frequent interactions with the PED community and
have plans to conduct formal evaluations to validate our design strategy recommendations and design concepts.

6. References

the Association Francophone d'Interaction Homme-Machine, 115-122.
behavior. Human Factors, 45, 260-280.
Meeting, 48, 147-151.


