19th ICCRTS

The Command Operations Dashboard: A Common Operating Picture of the Operators
Paper #111

Primary Topic: Experimentation, Metrics, and Analysis
Alternate Topics: Organizational Approaches, Collaboration, Social Media / Networking; Data, Information and Knowledge

POC Name: Andrew Duchon, Ph.D.
POC Telephone: 781-496-2490
POC E-mail Address: aduchon@aptima.com
Classification: Distribution Statement "A" (Approved for Public Release, Distribution Unlimited).

Andrew Duchon
Aptima, Inc.
12 Gill Street, Ste 1400
Woburn, MA 01801

Kara L. Orvis
Pacific Science & Engineering
9180 Brown Deer Rd
San Diego, CA 92121

Arwen H. DeCostanza
U.S. Army Research Institute for the Behavioral and Social Sciences
DAPE-ARI-FS
6010 Frankford Street, Office 221A
Aberdeen Proving Ground, MD 21005

Tara Rench
Aptima, Inc.
3100 Presidential Drive, Ste 220
Fairborn, OH 45324

Heather Wade
Aptima, Inc.
3100 Presidential Drive, Ste 220
Fairborn, OH 45324

Seamus Sullivan
Aptima, Inc.
3100 Presidential Drive, Ste 220
Fairborn, OH 45324

Caroline Ziemkiewicz
Aptima, Inc.
12 Gill Street, Ste 1400
Woburn, MA 01801

The Command Operations Dashboard: A Common Operating Picture of the Operators

Abstract

The Army trains battalion and higher units in Mission Command during realistic week-long exercises. Observer/coach/trainers (OCTs) shape the exercise scenario to the unit commander's training objectives, provide training and advice, and conduct after action reviews (AARs). With reduced resources, OCTs have limited access to the unit members' physical and digital interactions, preventing a complete understanding of their activities and performance. The purpose of this research was to determine what information about unit interactions was important and difficult to monitor, yet could be provided via recording, analysis, and display technology. OCTs were observed, interviewed and surveyed to create a list of requirements for a Command Operations Dashboard (COD) providing real-time access to communications data. To achieve these requirements, the COD uses a plug-in architecture to collect communications data and metadata into a unified database, a composable analytic framework to unobtrusively and objectively assess team states, and a user interface showing raw and analyzed data. The COD will be evaluated for its utility and usability in helping OCTs 1) to determine parts of the unit needing more support, 2) to identify healthy and harmful interaction patterns, and 3) to improve training by moving from AAR to current action assessment.

Introduction

The Army trains battalion and higher units in Army Doctrine, Mission Command, and the Operations Process during realistic week-long exercises. These exercises assist the training unit in preparing for full spectrum operations in both decisive action and stability scenarios. The exercises mirror the continuous nature of operations, and consist of a planning phase where units gather information about enemy forces, civilians and terrain and determine a course of action (using the Military Decision Making Process, MDMP), and an execution phase. Observer/coach/trainers, or OCTs, conduct these exercises, supporting the commander's training goals, and running mid and final AARs for the training unit. These exercises are highly interactive and require the commander and staff to coordinate efforts, applying their different areas of expertise in what is essentially an intellectual group problem solving exercise.

During Mission Command and "throughout the operations process, commanders encourage continuous collaboration and dialogue among commanders, staffs, and unified action partners to create shared understanding and facilitate unity of effort" (US Army, 2012a, p. 5). Mission Command therefore requires effective teamwork across space and cyberspace, over time, and in every echelon. Aspects of good teamwork include: high levels of unit cohesion to help units withstand the demands of combat (TRADOC, 2010a, p. 21), mutual trust that flows through the chain of command (US Army, 2012b, pp. 2–2), clear awareness of commander's intent so subordinates can exercise proper initiative in unexpected situations (US Army, 2012b, pp. 2–4), and accurate and timely situational awareness which enables mission command (TRADOC, 2010b, p. 40). In the end, good teamwork relies on good communication since information "needs to flow up and down the chain of command as well as laterally to adjacent units and organizations" (US Army, 2012b, pp. 2–86). When issues occur during training or operations, they are often blamed, after the fact, on poor communications.

Thus, to properly train Mission Command, OCTs need to be aware of the teamwork and communications taking place within the training unit. But how can the OCTs know if a part of the organization is experiencing poor teamwork? While they can observe some of the face-to-face interactions occurring, most of the communications are hidden from view via radio channels, in digital streams like email, chat, videoconferencing and VoIP, or in face-to-face interactions in other locations. Even with access to all this information, how can they know if the pattern of communications indicates poor cohesion or trust that could lead to poor information flow or more severe incidents? To support this understanding, the Command Operations Dashboard (COD) is being developed to analyze and display communications content and patterns, as well as real-time measures of teamwork. With this information, OCTs can have the opportunity to address concerns about the organization as quickly as possible, making the training process more agile and adaptive. The sections below describe the process for gathering requirements for the COD, the technology to make such a dashboard possible, prototypes of the dashboard, and future plans for testing its utility and usability in a live exercise.

COD Requirements Development

A multi-step approach was used to identify OCT requirements for the COD interface. These requirements address what types of information would be valuable to the OCTs, what data would be required to obtain that information, and how that information could be best be provided. The requirements development process began with a review of training materials from past exercises to better understand the organizational structure of the OCTs, their general background and experience, and their job tasks. Semi-structured interviews and on-the-job observations with the OCTs were then conducted to better understand the OCT job and collect an initial list of requirements. These interviews and observations took place during a training exercise with Division and Brigade level staff members. A survey was conducted after the training exercise, in order to identify and rank the most critical and attainable requirements. The interviews, observations, and survey activities are detailed in the following sections.

Interviews and Observations

Participants

Participants included 28 OCTs who were assigned to one of two brigades. Two of the OCTs were acting as lead OCT for one of the two brigades, which meant they managed the other OCTs and coordinated the development of the unit AAR and other feedback. The remaining twenty-six OCTs were tasked to observe specific warfighting functions (US Army, 2011) within their assigned brigade, and in some cases, specific individuals within those warfighting functions. Table 1 indicates the number of OCTs within each warfighting function that participated in the interviews. Most of the OCTs were at the rank of either Major or Lieutenant Colonel, and there was a range of novice and expert OCTs. The most novice OCT was experiencing their first exercise and had been with the OCT group for one month, while the most experienced OCT had supported 30 exercises and been with the OCT group for 32 months. The average experience level for OCTs was approximately one year and six training exercises. All OCTs noted that they were selected for the OCT position because of their expertise within the warfighting function they were assigned to. Although many had not received formal OCT training, they were continuously learning doctrine relevant to the warfighting functions and types of units they were training.

Table 1: Number of OCTs interviewed, observed and surveyed by warfighting function

Warfighting Function	Interviewed/Observed	Surveyed
Mission Command (MC)	5	1
Movement & Maneuver (M2)	4	2
Protection	5	0
Intelligence	6	1
Sustainment	2	2
Fires	2	1
Inform and Influence Activities	2	2
Other (Lead Brigade OCT)	2	1

Procedure

OCT Interviews. Two researchers interviewed OCTs over the course of four days during the set-up of a multi-echelon training exercise with division and brigade level staffs. The main purpose of these semi-structured interviews was to better understand the OCTs' responsibilities and identify specific requirements OCTs had for the COD, including the type of information and user interface preferences based on prototype designs. The interviews also helped the researchers develop relationships with the OCTs prior to observing them during the exercise.

OCT Observations. Following the interviews, five researchers conducted observations of the OCTs during the warfighter exercise. The purpose of the observations was both to validate COD requirements gathered during the interviews and to identify new ones. Due to the nature of the mission and the exercise schedule, the researchers observed the OCTs interacting with two different support brigades during the exercise. A single researcher would shadow an available OCT for a period of time as the exercise allowed. The OCTs were helpful in answering researcher questions and took a fair amount of time to explain what they were looking for and thinking about during the exercise. Many OCTs came to the researchers with suggestions for COD requirements.

Results

Following the exercise, two researchers reviewed the OCT interview and observation notes and identified 228 unique requirements for the COD. To facilitate the organization of the 228 requirements, they were grouped into one of 18 categories (see Table 2 for a list of categories and descriptions the initial version of the COD is being built to address). A larger research team then reviewed each of the requirements and 35 were identified as "must-haves" based on information gleaned during the OCT interviews and observations. Forty-eight requirements were deemed as "not critical" or logistically too difficult to include in the current design effort. The remaining 145 requirements were used to create the OCT survey.

Table 2: Subset of requirement categories originating from OCT observations and interviews prioritized for the first version of the COD

Category	Category Description ***Requirements in this category are focused on…***
Filtering Options	Identifying the specific features that OCTs could select from to manipulate and select what subset of the data they would like to view.

Monitor Content of Communications	Monitoring what types of information/ topics were being discussed (key words, specific emails, topics).
Monitor Flow of Communications	Monitoring the flow of communications between individuals, units, WFFs, etc.
Monitor Process	Monitoring or tracking when and how well the unit is engaging in specific processes (e.g., MDMP; battle drills).
Track Key Events	Monitoring and tracking key events during the exercise, including SIGACTs, meetings, etc.
Overarching ("Big Picture")	Monitoring and assessing big picture information during the exercise (more general requirements than other categories).
System Design/Layout	Specifying what design features the COD needs to include.
System Flexibility	Specifying the level of flexibility the COD needs to have to adapt to different exercises, units, etc.
Type of Data	Identifying the different data sources (e.g., email, chat, face-to-face, etc.) that the COD needs to capture and analyze.

Requirements Survey

A survey was developed to rank the importance of the various requirements (beyond the must-haves) found during the observations and interviews. Combining this information with the technical feasibility of the requirement allowed the development team to set priorities for the first version of the COD.

Participants

Sixteen OCTs who had participated in the observations and interviews during the exercise were sent the requirements survey and ten completed it. All of the warfighting functions except Protection were represented by OCT survey responses (see Table 1). Similar to the interviews and observations, the OCTs completing the survey ranged in experience level. OCTs completing the survey averaged 14.5 months experience. The most novice OCT had 4 months of OCT experience and the most experienced OCT had 32 months. The OCTs completing the survey also ranged in the number of past exercises they had been an OCT for (average: 6.1 exercises; range: 2 to 15).

Procedure

Survey Development. Two researchers reviewed the 145 remaining requirements to determine how best to capture them in the survey. Several requirements were similar enough to combine into a single survey item or were reworded to help clarify their intent. Examples were added where appropriate. In survey, the requirements were grouped by broad category (see Table 2 for examples).

Two different types of response scales were used on the survey to ensure that the developers received the information needed to prioritize tasks. For the majority of the survey items, OCTs were asked to rate how critical it would be for the COD to have the requirement in order for the COD to be useful to them. The rating scale ranged from 1 (not at all critical) to 4 (extremely critical; must-have). OCTs could also indicate "N/A" if they did not understand the requirement. For the remainder of the survey items, OCTs were given a list of requirement options (e.g., a list of potential data sources that the COD could track) and asked to rank order them by criticality level (1 = most critical). Finally, a few open-ended questions were included that allowed OCTs to provide comments or suggest additional requirements.

The final survey included 88 rating questions, 10 rank-order questions, and 15 open-ended questions. In addition, five background experience questions (i.e., number of months as an OCT, number of exercises, assigned WFF, and assigned shift [day vs. night]) were included at the beginning of the survey to better understand the sample providing feedback on the requirements.

Survey Administration. The requirements survey was web-administered to allow for rapid dissemination and collection of responses. The OCTs were sent an email with the introductory text and a picture of an initial prototype of the COD (Figure 1) for reference, and a unique and anonymizing survey link. The unique survey links allowed OCTs to enter, exit, and re-enter the survey as often as they needed to complete the survey.

The prototype was created to show the basic elements based on the “must-have” requirements identified previously. The wireframe showed that data could be filtered by source (who sent a message) and type of message as well as by time. Only the messages that passed these filters would be included in a Network view (who was sending messages to whom). The content of the filtered messages could be summarized in terms of Topics over Time (lower left); or in terms of specific keywords shown as a Word Fall (lower right).

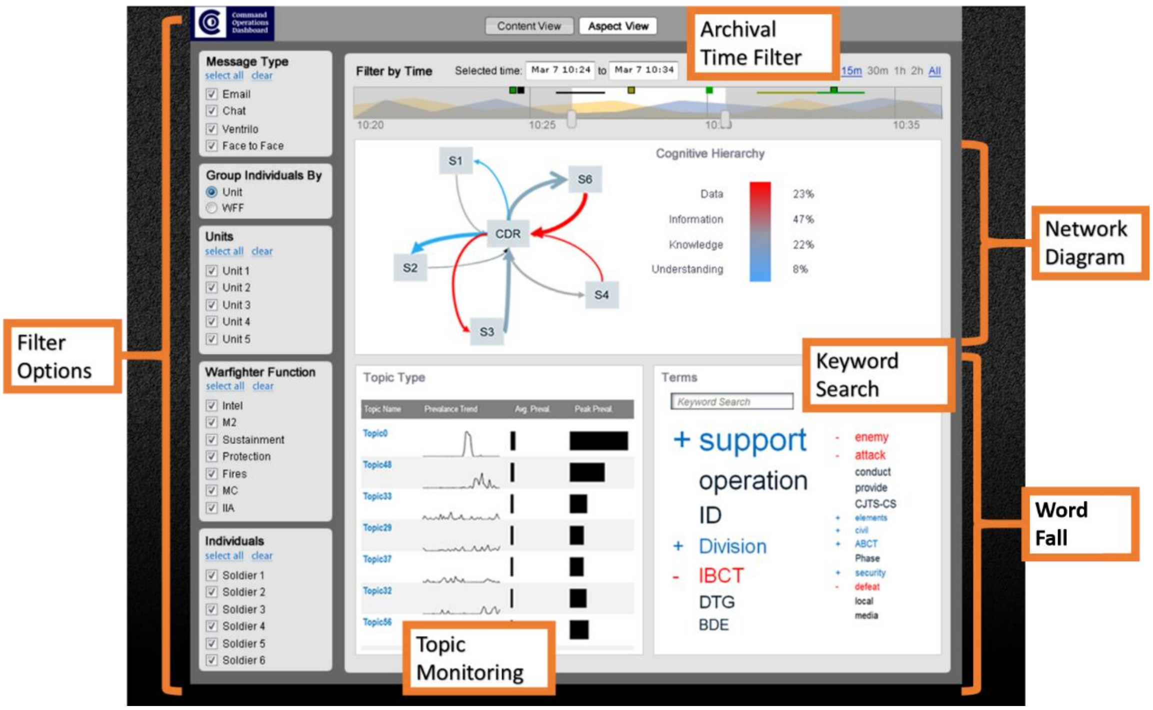

Figure 1: COD prototype sent to OCTs for the survey

Results

Rating Scale. For the survey items that were rated on a 4-point criticality scale, the average, standard deviation, and count by response option (e.g., number of OCTs who selected “not critical,” number who selected “slightly critical,” etc.) was calculated. Results showed variability across the requirements in terms of criticality level, ranging from 2.40 to 3.70 on a 4-point scale. Higher values indicated requirements that were more critical. A cut-off of greater than 3.00 (average) was used to separate the “top OCT requirements” (i.e. the most critical requirements) from the remainder of the requirements on the survey. Fifty-two (out of the 88) exceeded this cut-off, with OCT ratings indicating that these requirements were leaning toward extremely critical. The number of “extremely critical” ratings made for each of these “top requirements” was documented to further prioritize and inform development discussions. It should be noted that all requirements were noted as critical (no item had an average of less than 2.40) which provided validation evidence of the requirements gathered through the interview and observation method.

Ranking Scale. The ranking items were analyzed separately, as the ranking of criticality for each requirement option was relative to the other options within that item. For each ranking item, the average rank was computed for each requirement option, with lower averages representing greater criticality (1 = most critical). The number of options that were ranked varied by item, with as few as four and as many as twelve. A two-part strategy was used to determine which options ended up in the "top ranking requirements" list. First, the average ranks for each option were examined within each group to see if there was a natural break in the averages across options. If there were two to three critically-ranked options and then a big jump, then those top two to three options were added to the requirements list. However, if there was not a natural break in the averages between options, then the top three options were selected as "top ranking requirements." In at least one case, there were only four ranking options, and no natural break, so all four options were included as "top ranking requirements." Thirty-two requirement options were selected as "top ranking requirements".

Open-Ended Items. A few open-ended comments were provided by OCTs. The open-ended responses were generally limited and reviewed by researchers. No additional requirements came out of the open-ended responses.

Final Requirements

To focus COD development, the final requirements were ranked by the development team based on the order in which they should be logically developed, the certainty about how to implement the requirement, and budget constraints. Below we discuss those planned for the initial release of the COD.

Must-have Requirements

Table 3 lists the categories of must-have requirements in the initial COD release, i.e., those that were deemed necessary to have in the COD, so were not surveyed. The requirements associated with the Backend component (not directly seen by an OCT) were used to develop the CommsDB and Communications Data Collector described below. The Admin component would be another tool by which to enter information into the system. The other requirements are UI components and were used to sketch out the basic aspects of the prototype shown to the OCTs in the survey (Figure 1).

Table 3: Must-have requirements

Category	Requirement	Component
Monitor Flow of Communications	System should track when communications are happening (time-stamp).	Backend
Monitor Flow of Communications	System should provide raw network diagram data. The expertise of OCTs allows them to make meaning of that data and detect anomalies.	Network
Monitor Process	System needs to monitor collaboration.	Network
Overarching	System needs to allow for OCTs to observe interactions near real time.	Backend
Overarching	System needs to allow for OCTs to access archived data/ information that can reveal what happened within a past period of time (e.g. several hours to 10 minutes).	Backend
Overarching	System needs to consider and present data from multiple echelons (e.g. Battalion and Brigade; Brigades and Division).	Filter
Overarching	System should help OCTs monitor the unit unobtrusively, and when OCTs are not present.	Backend
Overarching	System should assist OCTs in monitoring what is happening in the unit (OCTs noted they can't write down information quick enough before the next thing happens; said he can't hear/ see everything).	Admin
Overarching	System should display raw data/ information in a way that can be easily digestible within five seconds.	General UI
System Design/ Layout	System should have a main screen that provides: 1) filter options for OCTs to choose from; 2) general information with regard to who is/was talking to	General UI

	who (e.g., network diagram); 3) some high level insight into what they are talking about (e.g., key words), and 4) high level assessment of state of team. The OCTs will then want to dig into specific communications.	
System Flexibility	System needs to be flexible enough to accommodate the different roles and role structures associated with different types of units.	Backend
System Flexibility	System needs to consider that some individuals will fill multiple roles (this happens especially during night shifts) and multiple individuals may cover a single role. There will not always be a 1 (person) to 1 (role) mapping.	Backend
Track Key Events	System needs to continuously display a timeline (which includes date/time and key events) with comms information. Tracking of events provides context into what is happening with the communications.	Timeline

OCT Highly Rated Requirements

In the survey, the OCTs highly rated a total of 52 requirements, but only five of these were deemed feasible in the initial release. A number of the others would require a) access to information sources that were unlikely to be available initially (e.g., Requests for Information); b) too much domain knowledge (e.g., standard operating procedures); c) other Backend development (e.g., OCT user accounts); or d) further validation of measures (e.g., team cohesion) which is still being conducted in related work (e.g., Orvis, Duchon, & DeCostanza, 2013).

The five requirements addressed in the initial release of the COD are presented in Table 4 and are related to the monitoring of basic flow and content of communications.

Table 4: OCT highly rated requirements

Category	Requirement	Component
Monitor Content of Communications	Capture who is talking to who and when, with some insight into what they are talking about	Network, Word Fall
Monitor Flow of Communications	Track how communications flow within and between units and echelons (e.g., BDE1 to BDE2; BDE1 to DIV)	Network
Monitor Flow of Communications	Allow OCTs to click on the lines in a network diagram to see more details about the specific communications (e.g., specific emails, key words, quantity or volume of communications, communication mode)	Network
Monitor Flow of Communications	Allow OCTs to see who is in control of communication flow (e.g., who is central? Who is interacting with the most people?)	Network
Track Key Events	Allow OCTs to look at communications before, during, and after key events (e.g., injects, SIGACTs, meetings, network failure, unit member leaving or changing roles)	Timeline
Track Key Events	Monitor how key role players share information immediately following key meetings	Timeline, Word Fall

OCT Ranked Requirements

A total of 32 requirements made the cutoff in the rankings for the various questions. All of these requirements are listed as they provide insight into what the OCTs value most.

Data Sources

For the data sources question ("For the COD to be useful to you, please rank the importance of being able to monitor/track communications that occur via:"), the OCTs ranked face-to-face highest, followed by Ventrilo (a system used for text and voice chat), Command Post of the Future (CPOF), VoIP and email. While email (e.g., Microsoft Exchange) can be easily obtained directly with an auto-forward rule, unfortunately a number of other critical sources are very difficult to access (e.g., Ventrilo, CPoF). The physical, face-to-face interactions of the Soldiers during the exercise are what the OCTs themselves are monitoring most, but they of course cannot be everywhere at all times. Fortunately, face-to-face interactions can be captured via badges described below.

Filters

In terms of filters ("For the COD to be useful to you, please rank the importance of being able to filter and look at communications data by:"), knowing the basic information about the communications (mode, direction and system) were most important, with specific documents and Priority Information Requests (PIRs) also making the cut. Similarly, when focused on specific types of categories that could be applied to communications ("For the COD to be useful to you, please rank the importance of being able to categorize communications by:"), PIRs were also at the top of the list, followed by commander's critical information requirements (CCIRs), specific information requests (SIRs), and targeted areas of interest (TAIs).

Message Content and Flow

The OCTs also indicated that they were interested in seeing the keywords present in the communications. Beyond just monitoring for keywords, OCTs were asked if they would find it helpful to track particular pieces of information as they flowed through the communications, e.g., the S2 receives and email from an intel analyst and then forwards it to the S3 who tells the commander. The OCTs were interested in tracking the flow of all four types of information presented in the survey: CCIRs, SIGACTs (significant action, i.e., one that may change decision making), PIRs and MSEL (master scenario event list—the event injected into the simulation).

Table 5: Top OCT ranked requirements (lower Average means ranked more critical).

Category	Top Requirements	Average	*SD*
Data Sources	Face-to-face	2.90	*2.18*
	Ventrilo	3.80	*2.53*
	CPOF	4.40	*2.01*
	VoIP	4.60	*1.90*
	Email	4.70	*2.54*
Filters	Specific mode of communication	3.80	*2.25*
	Directional flow (sent vs. received)	4.20	*1.48*
	Specific system	4.50	*2.42*
	Specific document	4.70	*2.11*
	PIR	4.80	*2.97*
Categorize	PIR	2.10	*0.74*
	CCIR	2.40	*1.35*
	SIR	4.90	*1.79*
	TAI	4.90	*1.85*
Content	Monitor PIRs	1.10	*0.32*
	Monitor SIRs	2.40	*1.17*
Flow--Details	Key words in comms	2.30	*1.06*

	Breakdown by comms mode	2.80	*1.81*
	Quantity (#) of comms sent or received	3.00	*1.25*
	List of specific emails	3.20	*1.55*
Flow--Tracking	CCIR	2.00	*0.82*
	SIGACT	2.40	*1.17*
	PIR	2.60	*1.35*
	MSEL inject	3.00	*1.05*
Key Events--Tracking	Briefs	3.10	*2.02*
	Working group meetings	3.20	*1.32*
	SIGACT	4.50	*2.88*
Process	Track running estimates	2.00	*1.05*
	Speed of a decision	2.20	*1.14*
Overarching--Comparison	When CDR is present vs. absent	1.50	*1.08*
	Across event types	2.50	*1.18*
	Day vs. night	2.80	*0.79*

Events and Processes

OCTs were asked, "For the COD to be useful to you, please rank the importance of tracking communications around each of the following types of events." Of the options, only Briefs, Working group meetings, and SIGACTs were significant. The OCTs also wanted to understand the unit's processes but only in terms of tracking running estimates for key positions and the speed of a decision (i.e., the time between information coming and a decision being made).

Comparisons

Finally, OCTs were asked to "rank the importance of being able to compare the content, flow or effectiveness of communications" between different situational conditions. The OCTs indicated that it would be most helpful if they could see the differences in communications when the commander is present versus absent. During the observations, the OCTs noted a number of instances of how the communications in the TOC (tactical operations center) changed when the commander left. Day and night shifts can differ significantly, especially in these exercises where the night shift is composed of more junior, though often no less effective, personnel who can behave within a less strict hierarchy. Feedback from OCTs also suggests that comparing communications across different types of events (e.g., after an IED attack vs. a personnel recovery) would be helpful.

Requirements Summary

A multi-step approach was used to identify OCT user requirements for the COD. Two hundred and twenty eight overall requirements were identified through interviews and observations. One-hundred and nineteen of these were identified as logistically possible and/or extremely critical requirements through both researcher identification and a survey. Moving forward, this list of requirements will inform the design of the COD, which is described next. It should be noted that fulfillment of all requirements would require access to nearly all of the information in the Army Battle Command Systems which is currently impossible due to the variety of systems and lack of APIs as noted, as well as the informal recording often used (e.g., via Word or PowerPoint).

Software Components

The requirements discussed above for a Command Operations Dashboard entail development of a number of different software components. On the backend, access to a variety communication streams is provided by a data-flow system called the Communications Data Collector (CDC) with plug-ins for each source. Because the OCTs need to understand who, as an individual person, is talking to whom regardless of the type of communication channel and who is playing which role when, a source-neutral database (the CommsDB) was created which can represent these relationships, as well as the raw data, metadata (org charts, events, etc.), and analyses. To conduct analyses, from the simple determination of which PIR a message may be concerned with, to the complex assessment of a team's cohesion, a previously developed content analysis package was used, as well as a composable system for creating new communications-based measures. Finally, the raw data and analyses must be made available to the OCTs in close to real-time so a number of web services have been created to feed a web-based display with components to allow filtering by organization and time in order to view specific networks and contents of communications.

Communications Data Collector (CDC)

The first step to the process is obtaining the communications data in a near real-time streaming manner. The Communication Data Collector (CDC) collects communications data from a wide variety of sources and stores the data in the CommsDB. The CDC is a data-flow system that uses a variety of collectors, each tailored to a specific communication source, to collect the actual data. Once a message is received, it is turned into a source-neutral format that is then stored, along with any associated metadata, in the CommsDB. Once storage is completed, analysis components outside of the CDC are alerted to the newly arrived message.

The CDC is a Java-based application that utilizes a number of applications in order to collect data: it utilizes Apache Camel as a light-weight enterprise system bus for routing communications through arbitrary workflows for preparing comms data for analysis, ActiveMQ for providing a JMS server for internal and external communications with Comms Analysis components, and Postgres (or any SQL database with a JDBC driver) for storing the source-neutral communications data. The standard flow of communication collection begins with the receipt of a new message from some communications source (XMPP chat message, email, etc.). This message is parsed by the appropriate collector, and represented in the CommsDB format with the appropriate Java objects. Finally, a new message receipt announcement is created and broadcast to a JMS topic, allowing Comms Analysis algorithms to begin their work. The CDC currently supports the collection of: e-mail (Outlook), XMPP- and IRC-based chat, Sociometric badges (Olguín et al., 2009), and Cisco-based VoIP phones.

The Communications Database (CommsDB)

The raw communications data collected by the CDC are stored in the Communications Database, or CommsDB. The goal of the CommsDB is to be a universal database that can accommodate any kind of communication or interaction, that is, data where there is an entity (person, group, or thing) producing a message that others may (or may not) receive. In our work to date, we have used this conceptual schema to organize data from a wide variety of sources, including email, chat, forum messages, Twitter, blogs, scientific journal articles, news articles, push-to-talk, face-to-face interactions, and a variety of data sets from academic studies of transcribed communication. Not only does the CommsDB help one conceptually to see the similarity between different types of data, but this schema also allows separate development efforts to more easily combine and share the results of their analyses. The current implementation of the CommsDB uses the Java Persistence API (JPA) (Keith & Schincariol, 2009) framework's code-first capabilities to generate its schema from a set of Java classes with JPA annotations. Figure 2 shows the main CommsDB tables.

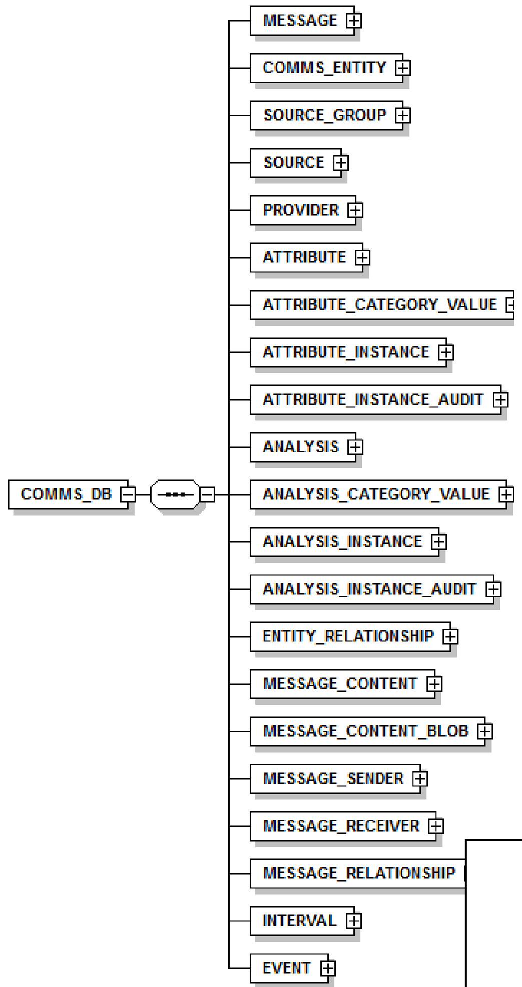

Figure 2: Main CommsDB Tables

At the core of the CommsDB is the **MESSAGE** table (see Figure 3). The **magnitude** field can hold any numerical data related to the message, e.g., the strength of a Bluetooth signal between two Sociometric badges. The messages themselves are linked to **MESSAGE_CONTENT** which can hold the plain text from any number of sections of the message, while binary or other mime-types are held separately in a **MESSAGE_CONTENT_BLOB** table. The zero or more senders and receivers of messages are held in the **MESSAGE_SENDER** and **MESSAGE_RECEIVER** tables, respectively. The entities that do the sending and receiving are contained in the **COMMS_ENTITY** table (Figure 4).

Each type of message (e.g. email, chat, phone, etc.) has an associated entity type that goes with it that represents the actual systematically identifiable entity that sent or received the message (e.g. email address, chat handle, phone number, etc.). In order to organize multiple communication media from these exercises, unique "role" and "person" entities are created for each individual involved, and then the "role" entities are related to the medium-specific entity with a "user_of" relationship in the **ENTITY_RELATIONSHIP** table (Figure 5). This allows different data sources to be imported at different times without necessarily knowing beforehand which person or role is using which account.

For filtering, every major type of data stored in the CommsDB (messages, entities, relationships, sources, etc.) can have attribute and analysis data associated with it (see Figure 6). *Attributes* are generally used to represent collected metadata inherent in the original source, while *analyses* are used as a repository for the results of analysis algorithms run on CommsDB data.

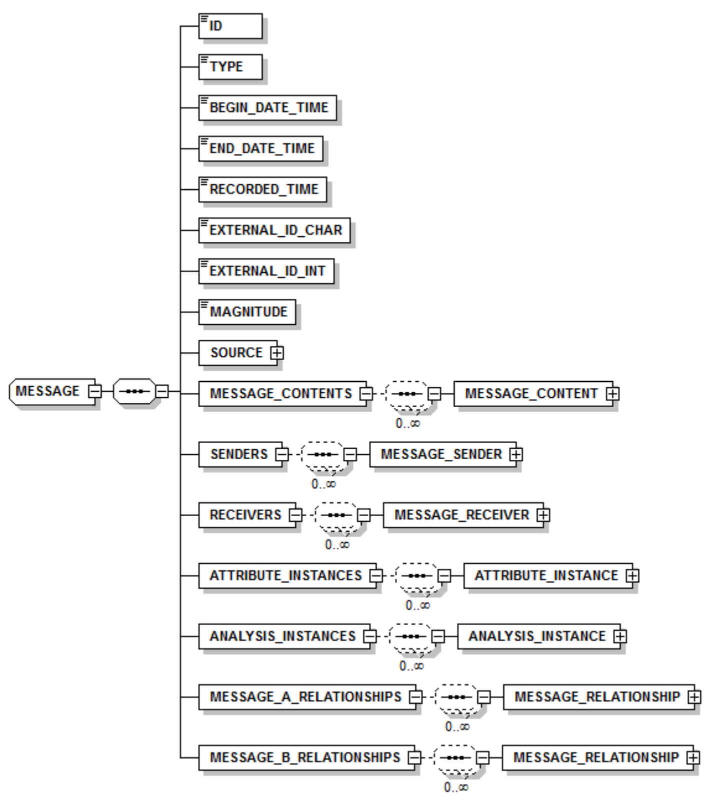

Figure 3: MESSAGE Table Fields and Relationships

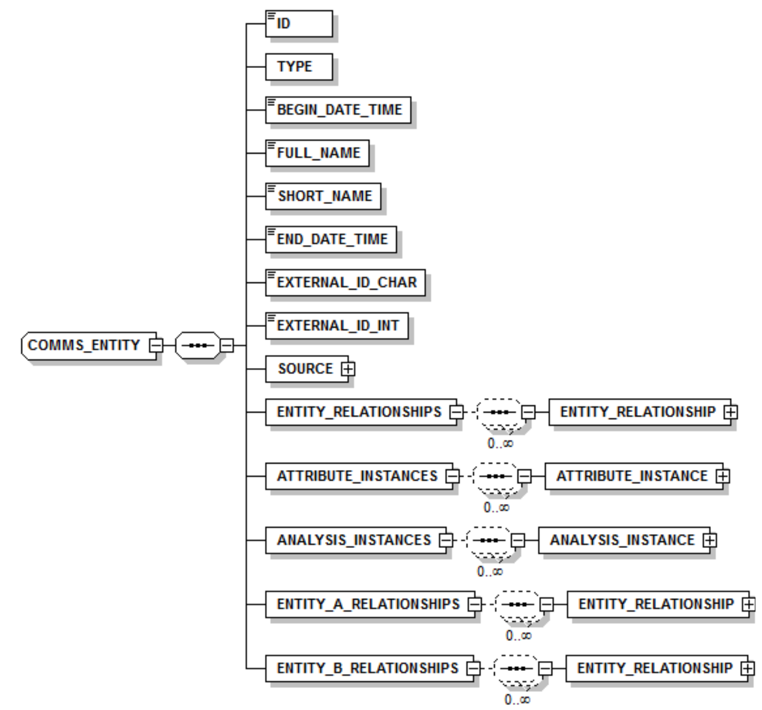

Figure 4: COMMS_ENTITY Table Fields and Relationships

The **ATTRIBUTE** and **ANALYSIS** tables store the definitions of the different types of attributes and analyses that can exist. The **ATTRIBUTE_INSTANCE** and **ANALYSIS_INSTANCE** tables store the instances that are actually associated with a resource (message, entity, etc.). While attributes and analyses have a similar structure in most respects, they do have one key difference: Attribute instances have a database uniqueness constraint that only allows one attribute instance of a particular attribute type per associated message or entity, while multiple analysis instances of the same type are allowed per message or entity as long as the attribute instances are created by different providers.

For example, the "leader" of a team may be given by the manning roster, e.g., the role of Brigade S2 is the leader of the Intel warfighting function. This information would be stored as both an attribute of that role entity, and as an entity relationship ("leader_of") between the Brigade S2 role entity and the Brigade Intel WFF organization entity. During the course of an exercise, there may be both a day-shift and night-shift person entities playing that role. An algorithm analyzing the content of email and chat to assess leadership (e.g., Duchon & Patterson, 2014) within the Intel WFF might show, with some probability or confidence, that the person playing the night-shift Assistant S2 is really showing the most leadership. The results of other "probable leader" analysis techniques could be associated with the other person entities, but have a different provider. By storing these all in the CommsDB, another algorithm could then combine the results to provide yet another estimate. In addition, the analyses are stored in an "audit" table with a timestamp, so, for example, as more email comes in, one can store an updated estimate of the leader and a user interface could then display how the estimate has changed over time. To power the Timeline (e.g., to quickly find the communications around time points of interest, e.g., after an IED attack) and to be able to do the types of comparisons that the OCTs requested (e.g., between, say, when the commander is in the TOC or not), time intervals and events must stored in the CommsDB as well. An **INTERVAL** represents any period of time and has a beginning and end, and can accommodate automatically created ones (e.g., 15 minute time blocks) and exercise-related ones (e.g., the first 24 hours of the exercise). An **EVENT** is a

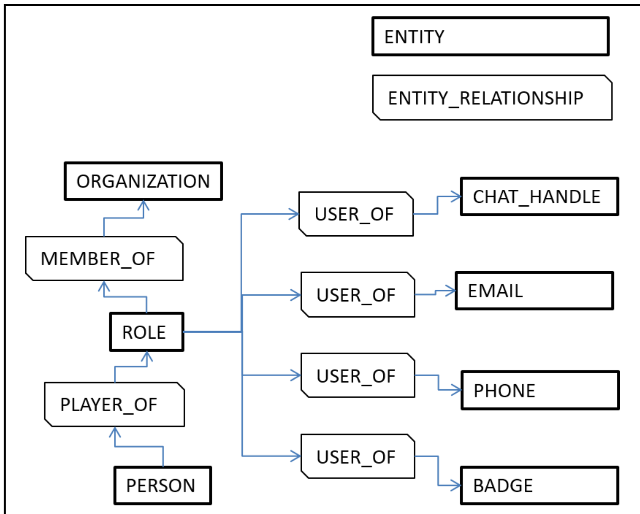

Figure 5: Common Entities and Entity Relationships

particular type of interval, having additional properties of a **category** field (such as "Scenario" for events from the MSEL) and a list of associated entities (to store who was involved in an event, such as the participants in a particular meeting).

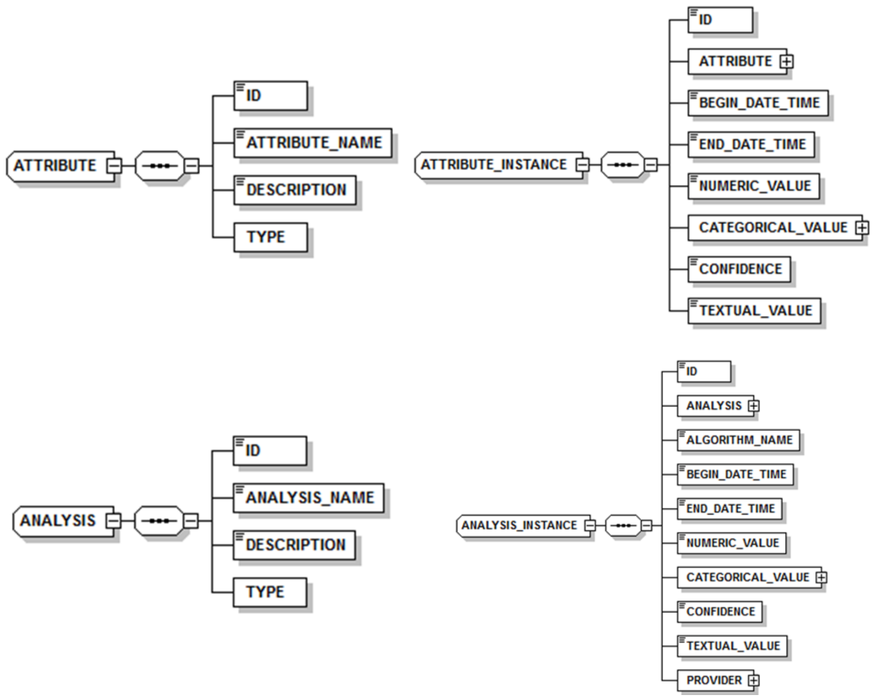

Figure 6: ATTRIBUTE, ATTRIBUTE_INSTANCE, ANALYSIS, and ANALYSIS_INSTANCE Table Fields

Metadata

As hinted at in the discussion of analyses and intervals, beyond the real-time communications that the CDC can collect, a great deal of metadata about the exercises is also required to be integrated into the CommsDB to provide context for the communications. This metadata includes who is playing what role, who (or what role) is at each VoIP phone number, background information about the scenario, and lists of specific events during the scenario. The latter documents help provide information about what should be talked about in the communications and using these documents we have created a statistical topic model (Blei, Ng, & Jordan, 2003) with which we can then analyze each communication to obtain a measure of its "work relevance." In addition, the survey data collected on team states and demographic information about the individuals are added as entity attributes to ensure they are synchronized with the rest of communications data.

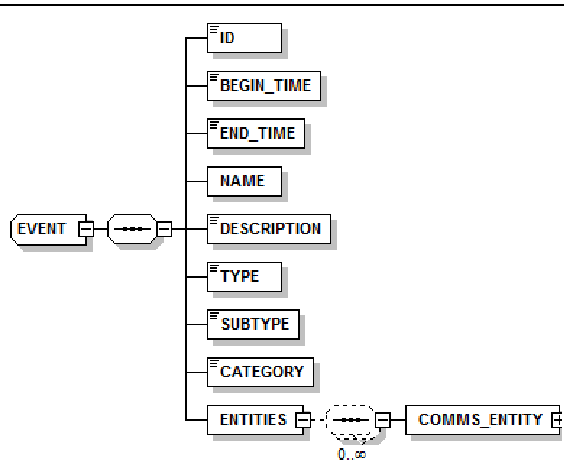

Figure 7: The EVENT table

Streaming Communication Collection and Analysis

While the OCTs are primarily concerned with access to the raw communications, finding those communications and guiding the OCTs to them requires analysis. Determining the PIR that a message is related to, determining which teams to spend more time with because they are less cohesive, comparing a unit's SOPs to its actual behavior, and even just determining "how much" communication is happening at any time, all require analyzing the communications.   For this reason, when a new message is collected by the CDC, a new message announcement is sent over JMS triggering a potential new analysis cycle.

Comms Analysis

Figure 8 gives an overview of the Comms Analysis workflow. The first step in the analysis process is to create a dataset, which includes the set of messages and entities to run analyses over. To create a dataset, two key pieces are needed: an interval, as defined above and a data filter which defines the values of the specific CommsDB objects to be loaded, e.g., **MESSAGE type**, **ENTITY type**, etc. Using this data filter, for example, an analysis can be run over all organizations in which members sent EMAIL messages.

Once a dataset has been created, *algorithms* are run over it. Each algorithm defines a set of *analyses* to be performed. An *algorithm* is a block of code performing the steps to analyze the dataset. An algorithm can analyze the data using a single analysis, or multiple. It returns a collection of results which are persisted to the CommsDB. An *analysis* is a measurement with one of three result types: *numerical, textual,* or *categorical*. In addition to the result value, all analyses have a *confidence* value referring to an amount of belief in the actual analysis value or that value's probability. These analyses are not unique to an algorithm, so it is possible to have multiple algorithms run overlapping analyses.

Example Analysis

As an example, we briefly discuss what is required to help the OCTs see how much communication is happening when and between whom. The problem is to equilibrate somehow the magnitude of these different channels—e.g., length of an email, IR pings from a badge, minutes of a VoIP call, etc. It was decided that the COD would simply convey to OCTs whether communication occurred or not (a binary value) during a fixed time interval on a given channel.

EntityMessageSentBinary Algorithm – This algorithm is run over a single entity. For each mode of communication (EMAIL, FACE-TO-FACE, etc.) and for each 15-minute interval, this algorithm calculates a binary value representing whether a message was sent by that entity, using that mode of communication, during that interval.

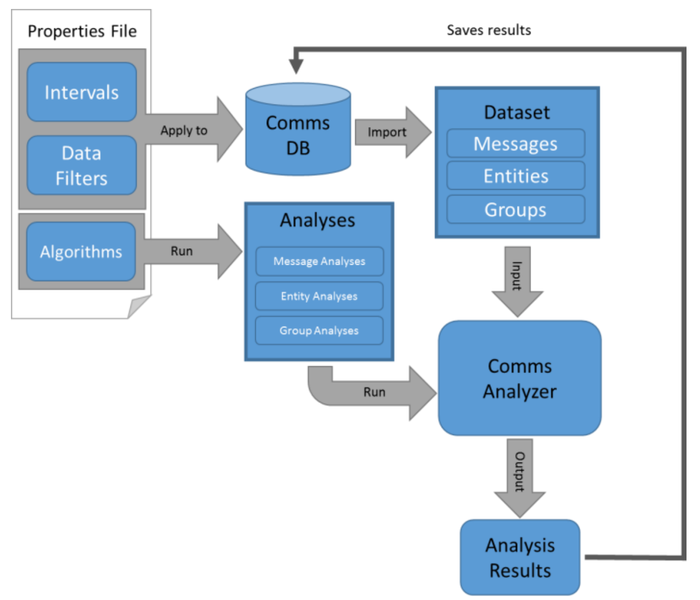

Figure 8: Comms Analysis Workflow

The goal of the Timeline view (below) is to represent whole-organization-level changes and response to the events. Each point on the timeline represents a 15-minute interval, the value for which is the sum of the binary values over each of the possible modes of communication and each of the entities (taking into account possible filtering using the filter widget).

EntityPairsMessageSentBinary Algorithm – This algorithm is run over a pair of entities (assume they are called A and B). For each mode of communication and for each 15-minute interval, the algorithm calculates a binary value representing whether a message was sent by entity A to entity B using that mode of communication during that interval.

The goal of the network view is to show the presence or absence of communications between individuals. For each edge (from entity A to entity B) in the network view, the edge thickness represents the number of communication channels over which entity A sent messages to entity B. These are summed over the number of 15-minute intervals that occur within the filtered time window. Edges are only displayed in the network view for entities that are applicable after taking into account possible filtering using the filter widget.

Command Operations Dashboard (COD) User Interface

Based on the requirements and feasibility, the initial release of the COD will have four components: a Filter component, a Timeline view, a Network view, and a Terms view. Together, they are designed to show what and when individuals communicated with each other, how events affected communications, and how communications vary across groups. The Overview screen uses HTML5 and JavaScript to connect to the Comms Webservices which feed these display components.

The *Filter* component allows users to select and deselect messages based on the groups of individuals involved and the type of message. In the initial release, individuals can be selected based on unit or warfighting function. The People panel updates as the filters change to show a list of currently selected individuals by role, rank, and name. Activity can also be filtered based on the type of communication, such as email or face-to-face. As these filters update, the other views will reflect only data based on the current selection parameters. The *Timeline* view shows a line graph of overall communications volume over time (based on the EntityMessageSentBinary Algorithm) as well as a “swim lanes” visualization of exercise events. Intervals are depicted as bars, while events are shown as tick marks below them. The *Network* view shows all the currently selected individuals with edges represented communications between pairs of individuals. The thickness and strength of an edge reflects the number of communications shared between the pair. The graph uses a force-directed layout so that people who communicate more frequently will be drawn closer together, while those that are more isolated will fall on the outside of the graph. Finally, the *Terms* view shows a sorted list of the most used terms in communications that match the current filter settings.

Comms Webservice

The Comms Webservice provides information on what roles and individuals are present, as well as their attributes such as Unit and Warfighting Function. It also exposes communication network analysis results (providing data for the Network view), term usage (providing the data behind the Terms view), events and intervals (providing data for the Timeline view).

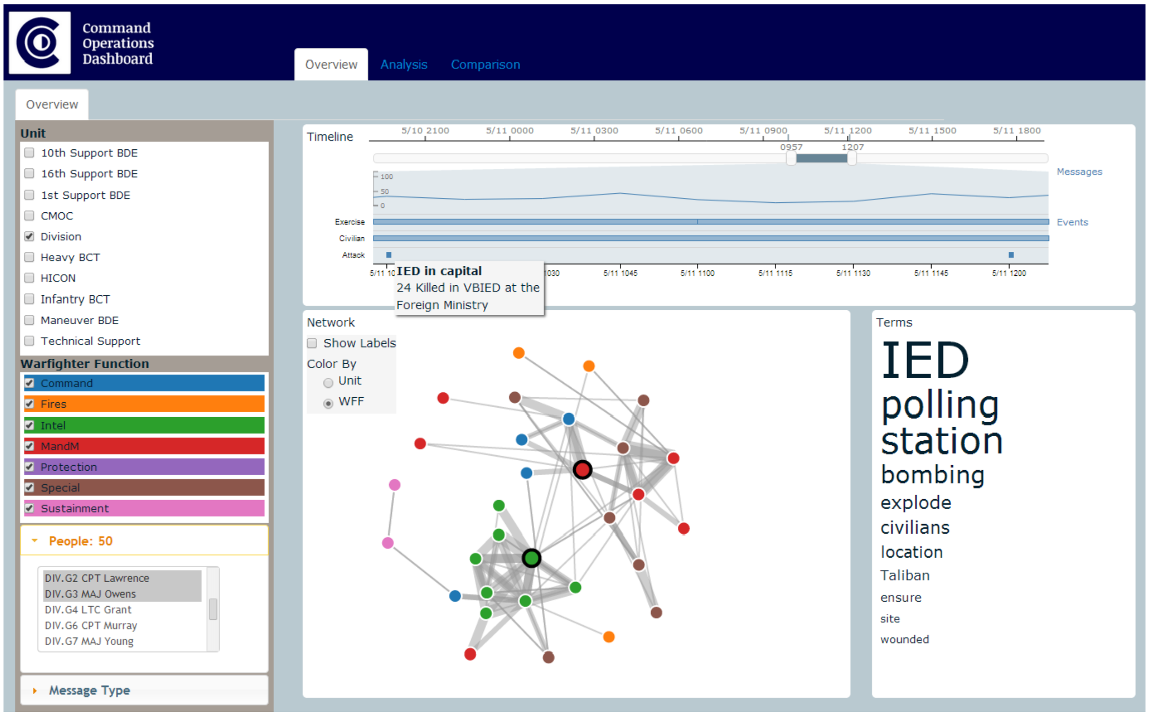

Figure 9. The current COD Overview interface, showing the Filter component, Timeline view, Network view, and Terms view.

In addition to the services that the COD interacts with, a processing pipeline generates data for the services and analyses. This pipeline runs Aptima's LaVA™ suite for feature extraction and topic assignment which are necessary for some analyses as well as for the Terms view. All messages with content (e.g., email and chat) are processed to extract collocations (multi-word terms determined statistically) and ignore stopwords, so the content shown in the COD Terms view is more representative of the meaningful aspects of the communications.

The Comms Webservice can also power third-party access via JSON. Standard CRUD (create, read, update, delete) operations on all the basic resources are possible with a simple credentials system. This allows third-parties to analyze the data collected in real-time and return the results to the CommsDB for display to OCTs. This means that other academic and commercial partners can develop systems of analysis using the webservice which can then be subsequently brought into the often classified exercise environment. This is critical for purposes of extensibility in general and researchers in particular so the data collected is a much more valuable and reusable resource for team science.

COD Assessment Plans

Versions of the CDC, CommsDB, and Comms Analysis components have been deployed at three large-scale (>700 participants) U.S. Army exercises. Data from the first exercise was provided after the fact and enabled the development of the CommsDB and Comms Analysis components, as well as initial testing of some measures (Duchon et al., 2011; Orvis & DeCostanza, 2013). The second exercise employed the first version of the Comms Webservice running live against email and provided further data for research (Orvis et al., 2013). Finally, in the third exercise, an initial version of a data display was deployed which presented real-time timelines of some team measures such as emergent thought leadership (Duchon & Patterson, 2014). Parts of the system have also been applied to analyze communications for a variety of purposes besides Army exercises. For example, content analysis of Air Force chat can be used to assess mission performance (Duchon & Jackson, 2010), Army teams in a virtual environment can be assessed for deployment readiness (Horn, Rench, Wade, & Duchon, 2014), and emergent thought leaders can be identified in ad hoc teams (Duchon & Patterson, 2014). All of these applications and associated analyses can be brought back into the COD to provide even richer content and context for trainers and commanders.

With the results of the requirements analysis reported here and subsequent development, plans are in place to test the usability and usefulness of the COD in an upcoming live exercise. The System Usability Scale (Brooke, 1996) and TAM questionnaire (Davis, 1989) will be used to obtain OCT assessments of each component. In addition, best use cases will be documented in order to help OCTs use the COD to be more effective, efficient and thorough in their understanding of the training unit.

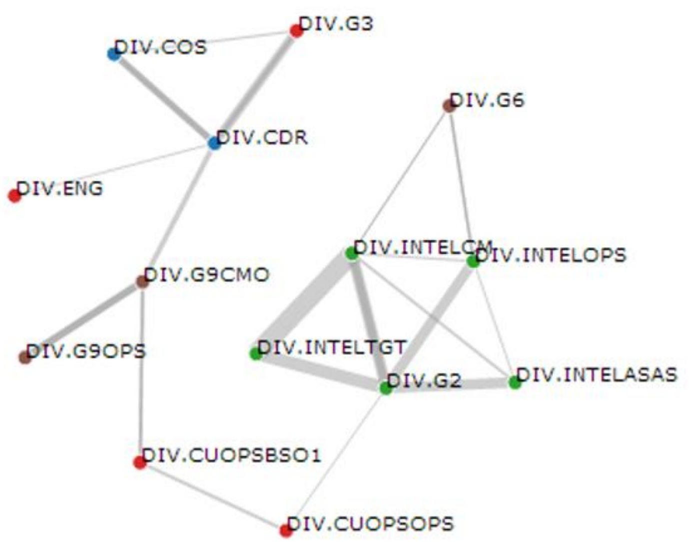

Figure 10: Sample face-to-face network

Take for instance the network of face-to-face data from a one-hour period of a division-level exercises (Figure 10). The Commander (CDR) interacts a lot with the Chief of Staff (COS) and Chief of Operations (G3). The Intel cell (green) is tightly knit as well. However, most OCTs will immediately comment that they are surprised by the lack of interaction between the G2 (Intelligence) and the G3 (Operations). Since there is always plenty to observe, the lack of interaction may be less salient when physically present, and if this type of network occurred during a critical phase of the operation, it could help OCTs understand why a coordination failure occurred and be used as illustration for immediate, simple and direct feedback without having to wait until the AAR.

Conclusion

The Command Operations Dashboard (COD) is being developed to provide OCTs with real-time information about communications in the training unit. Based on observations, interviews and surveys, requirements were developed for what would make the COD be most useful to OCTs. Based on this information, an end-to-end system was created which collects, organizes, analyzes and displays information for use by the OCTs. Ongoing validation studies use a composable analytic framework that applies advanced text analytics, network analyses, and dynamical systems analysis to these data to unobtrusively and objectively assess team states and processes. The resulting raw and analyzed data is expected to help OCTs by 1) guiding them to parts of the unit requiring more support, 2) providing solid evidence of healthy and harmful interaction patterns, and 3) improving training by moving from AAR to current action assessment. These assertions will be tested in an upcoming exercise.

This effort is the first step towards building tools for operational use by commanders to better understand their organization and provide them information to improve Mission Command. Similar requirements gathering is currently being conducted with brigade commanders to understand how they might use these data and analyses in practice. However, the systems being developed here are not only relevant to the Army or military—communications are critical for the proper functioning of any organization. In the current networked world, a wide variety of channels are available for individuals to communicate with each other. Social media and the sensor revolution, including sensors on smart phones, have opened up even more means by which information can be shared directly or indirectly with others. Task-specific tools and programs used for specific work endeavors (e.g., common operating picture systems, or "big boards" used for traffic control systems, power grids, and military operations) also provide methods by which one person can communicate with another by changing the status of information in the display.

These different forms of communication could be used as a rich source of data to aid organizations in a variety of domains. As discussed here, leaders and trainers could automatically assess the performance of teams (Mesmer-Magnus & DeChurch, 2009) to focus their limited resources on those teams needing more support. In terms of productivity, new information could be channeled to those users who can best make use if it (Xu, Ong, Duan, & Mathews, 2011), and away from those who cannot, in order to reduce information overload (Maes & others, 1994). For AAR (Ockerman et al., 2010) or e-discovery (Oard, Baron, Hedin, Lewis, & Tomlinson, 2010), these communications could enable a full understanding of the course of events in an organization and how decisions were made. To fulfill these functions, all of the many different channels of communication must be unified into a single accurate picture. This common operating picture of the operators can reveal the true functioning of an organization through its communications, so that leaders and trainers can assess and improve teamwork for improved operations as a mission unfolds.

Acknowledgment

This work was supported by W91WAW-09-C-0058 from ARI and W91CRB-11-C-0068 from DARPA. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of DARPA, the Army Research Institute, or the U.S. Government. We thank Gabriel Ganberg, Matt Jacobsen, and David Bauer for additional software and design support.

References

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent Dirichlet allocation. *The Journal of Machine Learning Research*, *3*, 993–1022.

Brooke, J. (1996). SUS-A quick and dirty usability scale. In P. W. Jordan, B. Thomas, B. A. Weerdmeester, & A. L. McClelland (Eds.), *Usability Evaluation in Industry*. London: Taylor Francis.

Davis, F. D. (1989). Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS Quarterly*, 319–340.

Duchon, A., & Jackson, C. (2010). Chat analysis for after-action review. Presented at the Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC).

Duchon, A., McCormack, R., Riordan, B., Shabarekh, C., Weil, S., & Yohai, I. (2011). Analysis of C2 and "C2-Lite" Micro-message Communications. Presented at the AAAI-11 Workshop on Analyzing Microtext, San Francisco, CA.

Duchon, A., & Patterson, E. S. (2014). Identifying Emergent Thought Leaders. Presented at the 2014 International Social Computing, Behavioral Modeling and Prediction Conference (SBP14), Washington, DC.

Horn, Z., Rench, T., Wade, H., & Duchon, A. (2014). Accelerating team readiness: Using technology to assess ground truth in team dynamics. Presented at the NDIA Human Systems Technical Conference, Washington, DC.

Keith, M., & Schincariol, M. (2009). *Pro JPA 2: Mastering the Java Persistence API*. Springer.

Maes, P., & others. (1994). Agents that reduce work and information overload. *Communications of the ACM*, *37*(7), 30–40.

Mesmer-Magnus, J. R., & DeChurch, L. A. (2009). Information sharing and team performance: A meta-analysis. *Journal of Applied Psychology*, *94*(2), 535.

Oard, D. W., Baron, J. R., Hedin, B., Lewis, D. D., & Tomlinson, S. (2010). Evaluation of information retrieval for E-discovery. *Artificial Intelligence and Law, 18*(4), 347–386.

Ockerman, J., Case, F., Koterba, N., Williams, G., Conrad, G., McKee, S., … Welshans, J. (2010). Process Instrumentation Systems for Training and Operational Test Needs With Case Study of Use at JEFX 09. *ITEA Journal*, *31*, 264–274.

Olguín, D. O., Waber, B. N., Kim, T., Mohan, A., Ara, K., & Pentland, A. (2009). Sensible Organizations: Technology and Methodology for Automatically Measuring Organizational Behavior. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, *39*(1), 43–55. doi:10.1109/TSMCB.2008.2006638

Orvis, K., & DeCostanza, A. H. (2013). Shared Mental Models of Command Intent. Presented at the 28thAnnual Conference of the Society for Industrial and Organizational Psychology (SIOP), Houston, TX.

Orvis, K., Duchon, A., & DeCostanza, A. (2013). Developing Big Data Based Performance Measures: A Rational Approach. In *The Interservice/Industry Training, Simulation & Education Conference (I/ITSEC)* (Vol. 2013). Orlando, FL.

TRADOC. (2010a). *TRADOC Pam 525-3-1: The Army Operating Concept*. Fort Monroe, VA: Training and Doctrine Command.

TRADOC. (2010b). *TRADOC Pam 525-3-3: The United States Army Functional Concept for Mission Command 2016-2028*. Fort Monroe, VA: Training and Doctrine Command.

US Army. (2011). *Army Doctrine Publication (ADP) 3-0: Unified Land Operations*. Fort Leavenworth, KS: United States Army Combined Arms Center.

US Army. (2012a). *Army Doctrine Publication (ADP) 5-0: The Operations Process*. Fort Leavenworth, KS: United States Army Combined Arms Center.

US Army. (2012b). *Army Doctrine Reference Publication (ADRP) 6-0: Mission Command*. Fort Leavenworth, KS: United States Army Combined Arms Center.

Xu, M., Ong, V., Duan, Y., & Mathews, B. (2011). Intelligent agent systems for executive information scanning, filtering and interpretation: Perceptions and challenges. *Information Processing & Management*, *47*(2), 186–201. doi:10.1016/j.ipm.2010.07.001