Linköping studies in science and technology
No. 1234

# Support for critical analysis of actions in a task-oriented collaborative command and control environment

**Ola Leifler**

Submitted to Institute of Technology at Linköping University in partial fulfilment of the requirements for the degree of teknologie doktorsexamen

Linköping University
Department of Computer and Information Science
Human-Centered Systems
SE-581 85 Linköping, Sweden

Linköping 2010

ISBN 91-1234
ISSN 0345-7524


Typeset using LaTeX

Printed by LiU-Tryck, Linköping 2010

Till min älskade Tilda.

# Abstract

abstract of the thesis, form Abstract.tex

# Contents

# List of Figures

# 1 Introduction

In this chapter, we describe the research question which has framed this work, the application domain in which our work has been conducted and the contributed papers which form the basis of this dissertation.

## Research question

This dissertation provides an analysis of three cases of technical decision support systems to be used in military command and control situations. The application domain studied is the Swedish Armed Forces, which arehave been engaged in peace-keeping operations and operations other than war for quite some time. The main tasks performed by military commanders are to command units and control actions performed by them. In this respect, the tasks of command and control have remained the same for long. However, the current context of military command and control, where goals and priorities may change quickly, collaboration with other organizations may be paramount to success and much more focus is put on a holistic perspective on command, has changed our appreciation of exactly what it is that constitutes commanders' work. Our current appreciations of what decision-making is, in general and in the context of command and control in particular, presents us with great challenges when building systems to support decision-makers. Decision-making is currently not seen so much as a process of treating well-defined problems as optimization challenges by selecting bounded, optimal solutions to resource allocation or planning problems. Instead, decision-making is now seen as a fundamentally social activity of constructing a shared meaning and understanding, which makes support tool construction, and the evaluation of support tools, a larger challenge than a mere computational one. However, that is not to say computer-based support tools for analyzing and automating parts of the deliberation in command and control is not desirable, only that we must understand whether the basis for building decision-support applications are sound with respect to the environment in which they will be deployed and the function we expect them to have. This thesis consists of three case studies in which we have studied both tasks performed and tools support designed to support aspects of command and control.

During the course of the project, we have attempted to reconcile descriptions of *command and control* and *decision support* with one another in a manner that would help us characterize what it means to build a successful operational analysis system for command and control. Chapter 2 gives us a background to these concepts and how they are treated in this dissertation.

Concretely, the questions that we address in this dissertation are:

- *How can decision support systems that can conduct analyses of command and control scenarios support support commanders with their cognitive tasks?*
- *What are the conditions for successfully for supporting commanders through the use of support systems that analyze aspects of $C^2$ scenarios?*

These two questions have been explored in the context of three case studies based on technical platforms for intelligent analysis in command and control. With relatively new methods of analyzing and understanding command and control, the first research question, concerning how

The first case study was based on a support tool for military planning in a specific planning scenario, ComPlan. The tool was intended to provide a framework for different planning scenarios and critiquing mechanisms. The insights from the study along with a better appreciation of current challenges in command and control directed us not towards implementing other scenarios or critiquing mechanisms but towards general-purpose information management tools and task-specific analyses in such a framework. Military staffs typically make use of general-purpose office tools for managing information and communicating with others. Using task-specific information management systems may require both training and support, without which standard office tools are the primary tools used for managing information. Staff members also need to interoperate with others who lack specialized tools and communicate information and orders.

The second case study was therefore based on the analysis of standard desktop documents through the use of a Semantic Desktop framework. In our work, we extended an existing framework for Semantic Desktops with mechanisms for harvesting events from users that related to their particular context of work and provided visual representations of concepts they reason about through a domain-specific ontology in a Planning Desktop. The first and second case studies had

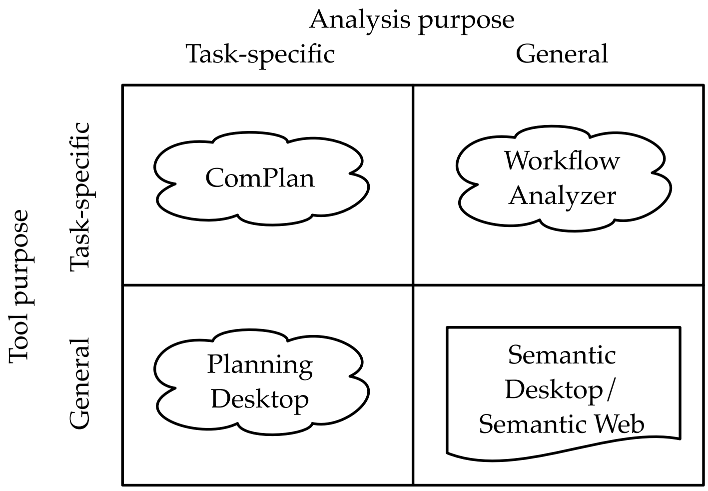


Figure 1: Three cases of intelligent analysis systems in command and control.

mostly concerned the construction of task-specific support in command and control, with a focus on visualization and manipulations of context-specific information. Task-specific support requires knowledge of the specific constraints that should hold in planning an operation or the specific types of information that are important to extract from documents.

The third case study concerned the use of general techniques for information extraction to improve the communication and performance of a staff. In this study, we used general models for analyzing text-based data sources and employed them for the specific purpose of supporting communication within a command staff. The tool used as a framework for the analysis of text was specifically designed for reasoning about communication among members of staff and was evaluated together with researchers studying command and control through the communication of members of staff.

All three case studies used technical platforms for studying methods for analysis of critical information. Figure 1 describes the relationship between these three case in terms of two parameters: The purpose of the technical system that was used and the purpose of the analysis techniques employed.

The purpose of the ComPlan system described in the first study was specific to the task of planning for and optimizing resource usage during a military of civilian operation. ComPlan was primarily intended for scenarios in which the resource constraints were known in advance and therefore possible to encode as part of the scenario itself. One of the primary lessons of designing the system was that there was a difficulty in using specialized tools precisely because they require much task-specific knowledge to be useful. In situations where constraints are not known in advance, the effects of actions are ambiguous or there is less structure available

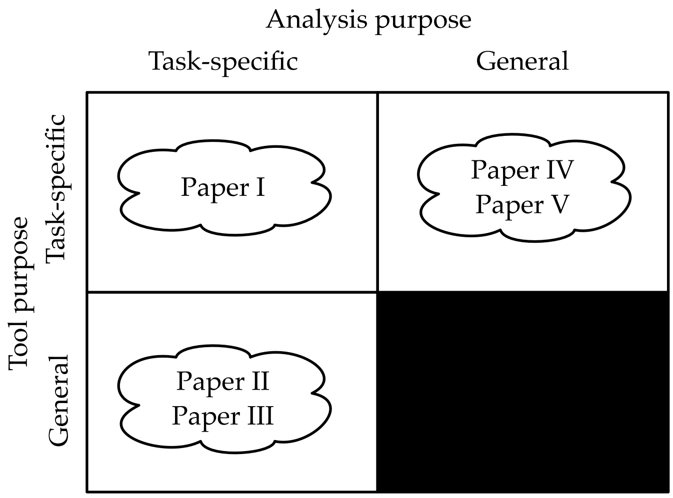


Figure 2: The relationship between the papers included in this thesis

overall, staff members use standard tools for outlining their intentions and synchronizing their work. The second study was therefore conducted in a general information management framework: a Semantic Desktop. The analysis conducted on the products (documents) provided by staff members was specific to the domain at hand but did not require logistical or other constraints to hold but merely assumed that terms and concepts used in planning would be present in documents treated by the staff. The third study was performed with a focus on staff communication, and analysis methods for understanding patterns in written communication among members of staff. The study used a task-specific technical platform for reasoning about patterns in datasets that came from exercises in which several text-based data sources were available for analysis. We used a mathematical representation of text clusters as a basis for analysis of communication patterns in general, not just related to command and control.

## Papers

Figure 2 describes how the papers in this thesis are related to each other in relation to the case studies conducted. Paper I represents the first case study, papers II and III represent the second, and papers IV and V represent the third.

**Paper I Ola Leifler. Combining Technical and Human-Centered Strategies for Decision Support in Command and Control --- The ComPlan Approach. In *Proceedings of the 5th International Conference on Information Systems for Crisis Response and Management*, May 2008**

The first paper presents the ComPlan approach to critiquing as decision support for mission planning, which is the first case study in this thesis. Based on observations of military planning and studies of previous support systems for command and control, the study investigates how critiquing can be implemented as a support concept for military planning. Several techniques for feedback and visualization are integrated in a framework for critiquing, and it presents a framework for the use of visual critiquing based on constraints in tactical, military planning.

**Paper II Ola Leifler and Henrik Eriksson. A model for document processing in semantic desktop systems. In *Proceedings of I-KNOW '08, The International Conference on Knowledge Management*, Graz, Austria, September 3--5 2008**

The first paper in the second case study describes the general requirements for constructing support systems for domain-specific information management in military planning, where plan documents are refined by several instances of command in the process of planning.

**Paper III Ola Leifler and Henrik Eriksson. Domain-specific knowledge management in a semantic desktop. In *Proceedings of I-KNOW '09, The International Conference on Knowledge Management*, September 2009**

The second paper in case study 2 describes an implementation of the model put forth in Paper II and presents semantic document processing as extensions to an existing semantic desktop through content- and structure-based information extraction, domain-specific ontological extensions as well as visualization of semantic entities.

**Paper IV Ola Leifler and Henrik Eriksson. Message classification as a basis for studying command and control communications - an evaluation of machine learning approaches. Submitted for publication, September 2010**

In the first paper of case study 2, we present a feasibility study of using machine learning algorithms to classify messages in a command and control scenario. The aim of the study was to determine the technical options for using machine learning to support command and control researchers understand patterns in workflow data. We also describe how the requirements for constructing support tools in command and control research relate to the requirements for planning tools that we elicited in Paper I.

**Paper V Ola Leifler and Henrik Eriksson. Communication analysis tools in command and control - open-endedness and transparency as design criteria. Submitted for publication, 2010**

In the second paper of case study 3, we build on the results in Paper IV and present a study and a design for a workflow analysis tool (Workflow Visualizer) that can assist researchers understand command and control workflows. The study comprises three parts in which we have interviewed command and control researchers, constructed a support tool to facilitate the study of command and control scenarios and evaluated the tool together with researchers to understand how the affordances of a support tool based on machine learning techniques in command and control research.

How do people navigate and form an understanding of patterns in large datasets?

## Outline

The remaining chapters of this thesis will describe the background of the technical systems and the tasks carried out with them in each case study. Chapter 3 describes the research method employed when studying these three cases, how cases were selected, and how each system was designed and evaluated. In Chapter 4, we describe the first case study, in which the ComPlan planning system was devised as a means of providing analysis support in military, logistically oriented planning. The second case study, The Planning Desktop approach to analyzing semantically rich text documents in planning, is described in Chapter 5. Chapter 6 describes the third case study which we conducted with command and control communications data, where we extracted information from several command and control scenarios and provided a design for an analysis tool that could provide command and control researchers with new methods for understanding how commanders cooperate and support them in their tasks. Finally, in Chapter 7 we relate the results from each of the studies to each other and provide conclusions regarding the conditions for successfully implementing support tools in dynamic command and control environments.

# 2 Background

This thesis addresses the issue of how intelligent decision support systems can be used in military command and control and how the function performed by such systems relate to the tasks performed in the domain of $C^2$. To this end, this chapter provides an overview of which classes of support tasks are performed with each technology and how these tasks are related to human tasks in command and control. To understand the relationship between the functions of command and control and technical systems that provide help to commanders, we will need to define first what commanders actually do, in terms of the functions they perform and how they perform them and then, how technical systems interact with commanders and influence their behavior. Last, we will also elicit how the functions in $C^2$ are evaluated and the relationship between three classes of support systems and the performance of staff members. These three classes of support systems are automated planning systems, information management systems and $C^2$ research analysis systems.

## Command and Control

The traditional view on *what* a staff of commanders do and *how* they do it centered round the notion of planning, preparations and deliberate actions [158, 92]. Military commanders were considered as the central points of decision-making who were tasked with gathering information, organizing an assembly of military units into a force appropriate for accomplishing a given, political goal[1] by targeting a

[1] often to subdue an enemy

single *Schwerpunkt* (center of gravity) of the enemy [38]. Jomini, who interpreted the strategies employed by Napoleon and wrote of how combinations of massing and manoever led to success on the battlefield, used similar concepts for describing how and where to employ force for maximal effect [43] as have other military thinkers. A center of gravity in modern warfare has been interpreted as an abstract concept such as ``the will of the people of a country to continue fighting'', or something very concrete such as ``the capital of the Soviet Union''. The notion is supposed to convey the most essential condition for an opponents continued war effort that, if significantly affected, would result in victory. Clausewitz was not comfortable enough with the concept of Schwerpunkt to specify it in other than circular terms. That is, a Schwerpunkt is that which will bring victory about, and that which will bring victory about is a Schwerpunkt. He stressed that there are no universal laws of warfare and no universal means by which to achieve victory [158].

The conceptualization of command, along with the conceptualization of warfare and the role of military forces, has changed radically over the last 20 years. The end of the cold war, and with it the end of a single, well-defined, overarching threat in western countries, the arrival of new means of communicating, visualizing information and dividing labor through technical systems, and new tasks and threats for military commanders, has resulted in the task of command and control being investigated by researchers from many fields.

One of the assumptions held in traditional views on military command was conception of the generation of a plan as a central act in command and one which must precede and determine the actions of units at the disposal of a commander. Although the circumstances of war always present commanders with uncertain, ambiguous intelligence and outcomes, the idea of producing plans and performing them were considered central aspects and the uncertainties and dynamics of war were considered ``frictions'' and suitable targets for technical development. Command and control is often a function performed by large groups of commanders who collaborate, with different roles and responsibilities, to provide a course of action at the echelon they are assigned to. The process they follow has been described as following that of a single fighter pilot [22], in which commanders react to external events that call for action, deliberate on options, execute an action and monitor the outcome. These four activities have been labeled Observe-Orient-Decide-Act and constitute a loop in which commanders, alone or in group, are supposed to operate.

The four principal activities of the OODA loop have long been identified as the main tasks performed in $C^2$. In older guidelines for planning at the operational level in NATO [119], after initiating directives have been established, a certain, well-defined procedure is assumed to follow among members of staff for gathering intelligence and in particular evaluating options for action. Intelligence information along with requests of assistance typically moves up the chain of command one step at the time, towards single nodes in the directed tree of command where

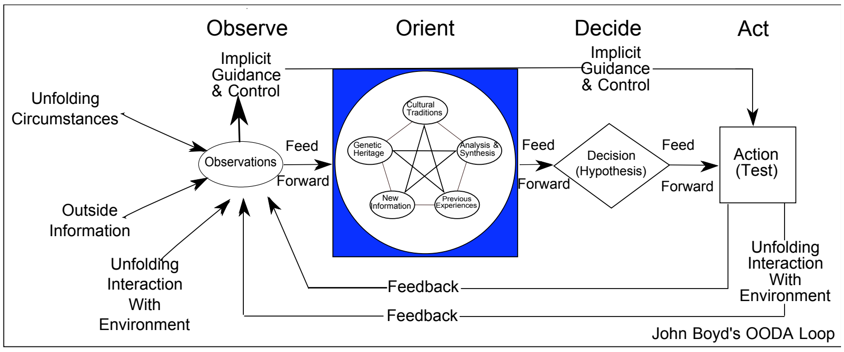


Figure 1: Boyd's Observe-Orient-Decide-Act-loop

commanders and analysts decide on how to process requests and interpret information. Orders and directives, on the other hand, are submitted to lower levels of command to be further specified and dissected, until tactical orders are issued to single units on the field. Several changes to this classical model of command have been suggested or already implemented over the last 20 years, however. First, the shift in nature of the objectives military commanders are given has opened for new options for interpreting what options are available for action. With the end of the cold war, operations of the military forces of many western countries became much more diversified, with peace-enforcing operations, anti-terrorist and humanitarian aid as equally important as the ability to repel a traditional military adversary. Along with new tasks to perform, researchers and policymakers began to identify how new information technologies could be used to accomodate these new tasks. In particular, command and control researchers described how future commanders would use Network-Centric Warfare (NCW) as a means of leveraging information superiority against military opponents [2, 32].

The NCW concept aims at reducing the time it takes in a large organization to process information, both intelligence information bound for higher command and orders directed to subordinate units. As a means of reducing the time to make decisions and put decisions into effect, Alberts et al suggested that commanders should be given access to information systems that would reduce or completely eliminate the need for human intervention in the communication between units [2]. By using electronic communication networks inspired by the World Wide Web, units on the field would be able to directly contribute battlefield information to central command and receive updates on the high commanders intent instantaneously. Also, NCW stated that units should be able to synchronize their actions without the need for interventions by higher command altogether, further reducing the need for detailed instructions to subordinate units. With a network of military units connected not as a tree but as a web, the authors of the NCW vision saw a possibility of transforming military organizations much the same way as the World Wide Web has transformed the ways people around the world communicate. With technology as an enabling factor for new methods of conducting warfare and the end of the cold war marking the end of centuries-old entrenched views about the purpose of military action and military tactics, Alberts et al saw information technology as a means to break old bonds. Arquila and Ronfeldt also describe how new tactics will be enabled through the use of communications technology, and how the concept of swarming an enemy with all available resources at the same time will become possible and desirable once the necessary coordination can be achieved through communication technology [11]. Together with the end of the cold war, however, military thinkers, and particularly those concerned with the essence of command and control, began to consider what it is that commanders actually *do* when they deliberate and communicate, *how* they do it and how it is *relevant* compared to all other factors for the outcome of conflicts.

Commanders are supposed to command others by rationally plan for a course of action that take into account all the known facts about a situation, a given operational goal, a set of military resources available to achieve the goal. This formulation of the deliberation that precedes the production of a battle plan suited the designers of computer-based support systems in the 1980's who have interpreted this description as amenable to at least partial automation by Artificial Intelligence planners [93]. However, with several AI systems developed and tested since then [21], several problems have emerged. Although AI systems are capable of producing plans that conform to a well-defined set of criteria, using a well-defined set op possible actions, planning systems are still problematic to develop and deploy in several command and control settings [41]. One way to understand the issues involved in deploying decision support systems for command and control is to describe in more detail the joint cognitive and social functions of a command and control staff and the describe the relationship these functions and technical systems for aiding plan development, execution monitoring and other aspects of command and control.

In particular, we will discuss whether command and control can be appropriately described as decision-making, or whether there are better characterizations.

### Command and control as Decision-making

Decision-making is classically described by Game Theory as the process by which an intelligent being selects optimal solutions among a set of alternatives to maximize some expected utility value [109]. According to this theory, a known goal should be described as something which depends on performing certain actions, and that performing an action deterministically produces an effect that brings the decision-maker closer to his goal. Given several options for how to achieve his goal, the decision-maker should select the option which has the highest expected utility according to some metric of utility.

This description is technically appealing because it describes the activities involved in decision-making as inherently rational, well-defined and known in advance. For board games such as chess, the description fits well with the restricted nature of the game and can guide the design of AI agents that play the game. The description is, however, very difficult to reconcile with work in command and control.

*Goals* in command and control are very difficult to define in the manner required for rational problem-solving according to the deductive process in classical Game Theory. Although overarching, political goals guide the operations military commanders are tasked to perform, such goals can be defined as "providing security for people living in an area" which is difficult to evaluate. Trends in military policy have also come to emphasize that goals should be less concerned with what can be achieved with military destruction power alone but rather look at what can be achieved with military and other resources together [16]. In complex, open systems such as those in which military commanders find themselves, several goals may even have to be attained at the same time [23].

*Options* for action may be straight-forward to define if goals are defined in terms of the direct effects of military power, such as the destruction of a strategic enemy center of gravity. The goal of destroying the communications network during operation Desert Storm in Iraq 1991 clearly required missiles and airborne missions, but was not sufficient for establishing a working democratic regime in Iraq 2003. In current operations, where commanders are supposed to collaborate with the surrounding community, non-governmental organizations and local authorities, the options for actions become much more difficult to enumerate, and even relate to the achievement of goals such as providing security.

### Decision-making as selecting optimal solutions

In many professional situations studied by researchers, the act of making decisions is not even characterized by the selection of optimal solutions by combinations of a limited set of possible actions [83]. Instead, it can best be described as process through which intentions are formed by the actions taken, and actions are taken based on cultural assumptions about what is possible and appropriate to do [121]. The entire command process can in this respect be regarded an ongoing social activity of negotiating between interpretations of goals, means and outcomes of actions [148]. With this context of reasoning, there is no meaningful definition of *optimal solution* and *selecting action*. Optimality cannot be evaluated because there is no single well-defined utility function to apply and selection cannot be performed because it presupposes a pre-determined set of options, whereas options for how to use resources are usually constructed as part of the problem-constructing process in military command and control [72]. Schön provides examples of what it means to construct a problem in this manner [135]:

> A nutritionist, for example, may convert a vague worry about malnourishment among children in developing countries to selecting an optimal diet. But agronomists may frame the problem in terms of food production; epidemiologists may frame it in terms of diseases that increase the need for nutrients or prevent their absorption; demographers tend to see it in terms of rate of population growth that has outstripped agricultural activity; engineers, in terms of adequate food storage and distribution; economists, in terms of insufficient purchasing power or the inequitable distribution of land or wealth. [135, pp. 4-5]

Although it would be highly unethical not to think of malnourishment as a problem per se, all the above framings of the problem hinge on the notion that no child should be malnourished. Had the problem concerned a broken car on the other hand, one could imagine that some people would not even consider it a

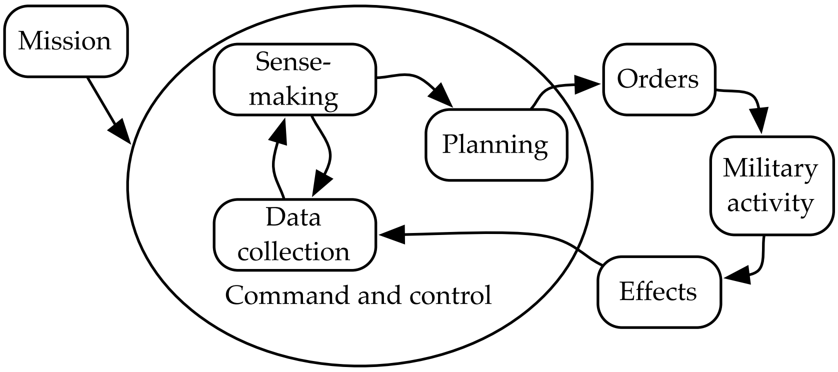


Figure 2: Brehmer's Dynamic OODA loop, where the central act is sense-making rather than decision-making, adapted from Brehmer (2005).

problem to be solved but merely a cause of the situated action that you take the bus instead [148].

### Functions of command and control

Decision-making in command and control as well as other contexts may not be best characterized as a process of selecting optimal actions, which has inspired researchers to consider models for command work that focus not so much on how commanders produce plan documents but instead how the build a common understanding of a situation (situation awareness) [131, 155]. Descriptions of command and control began to outgrow the OODA [22] costume when researchers began to add new possible activities and interplays between activities in the battle staff and new ways to frame the problem of command and control.

In Brehmer's description of the Dynamic OODA loop [24] (Figure 2), there is an inner circle of activities related to making sense of a situation that is common to all actors in a battle staff. Jointly, they improve their understanding of a situation by gathering information and integrating new information in their existing frameworks of understanding. In the course of building an understanding, the participants reach a state in which they feel confident to act, and issue orders to their subordinate units. Those orders can later be modified as new information becomes available, or old information is re-interpreted. In this description of military command and control, there is no linear process similar to observe-orient-decide-act, where there is a clear end-point when issuing orders (*act*). The process of command and control, as described in the DOODA model, emphasizes the importance of *sense-making* instead of decision-making [30].

Sense-making has been described as a goal-oriented, more specific concept compared to *situation awareness* [49]. The two concepts have been used to describe understanding, both individually and in teams, with a vocabulary that is similar to that used when studying the process of learning in general (e.g. [51]). In Endsley's description of situation awareness, he focuses on the process of recognizing a situation, that is, recognizing what is important and how he can act [49]. Situation awareness (SA) is supposed to be comprised of three levels of understanding:

1. Perception of elements in current situation (SA level 1)
2. Comprehension of current situation (SA level 2)
3. Projection of future status (SA level 3)

Although Endsley exemplifies SA specifically in professional contexts, the hierarchical view of knowledge as internalized in larger structures and used for making inferences and evaluations is similar to the learning taxonomy of Bloom et al [51]:

1. Knowledge
2. Comprehension
3. Application
4. Analysis
5. Synthesis
6. Evaluation

The theory of SA was developed within the context of aviation safety [49] and has since been adopted metaphorically for command and control [144, 9]. In command and control, the conditions for perceiving a situation and acting are radically different though, with the distributed nature of work, the complicated distribution of tasks between both human machines and technical systems, different time scales and means of affecting the environment call for different interpretations of the concept. Also, SA has been defined both as the end result of a process of comprehension and the process itself [50]. Equating SA to a process makes the task of evaluating there level of SA difficult. Also, it makes the application of SA as an indicator of team performance difficult. Several technical researchers have attempted various definitions of SA that can guide the construction of command and control support systems [9, 104]. In their attempts, they assume that patterns that are of relevance in situation awareness,

## Planning and Cognition

One of the central issues in understanding command and control, which is of particular importance to building support systems for decision-makers, is the role of plans and planning. In studies of other professional settings, Schön has characterized the relationship between action and deliberation using three main concepts: practice, patterns and theory[135]. The *practice* of a professional, whether military commander, physician or teacher, shapes the patterns they are likely to observe, which in turn guide the theories and explanations they devise to organize those patterns. Conversely, the theories constructed make certain patterns possible to observe and have a direct influence on how they perform their duties, their practice. Similar views on the tight interplay between situated actions and deliberation have been described by Suchman [148, 149] and Winograd and Flores [162]. Both Suchman and Winograd and Flores describe cognition and action with the intent of supporting designers of support systems for professionals. In their descriptions of what constitutes cognition, they use Heideggers theories of thinking and being as a situated practice that, to be properly described, always needs to relate to a context in which cognition is performed. Also, cognition is inherently social, much as Feyerabends description of science as a social activity of constructing meaning [34]. Suchman, Winograd and Flores all stress the importance of the situatedness of actions and interplay between deliberation and action. An important consequence of their view is that deliberate plans do not direct human action, plans merely frame a problem and (possibly) improve our understanding of goals, conditions and priorities. They describe how we use plans in civil engineering and other settings as tools for thinking about problems, but that they are not treated as scripts for action by people who are supposed to execute them [149].

Hutchin's describes how a team commanding a naval vessel maintains control over the ship even after an important instrument has broken down [78]: the crew manages to find navigate the ship by organizing themselves as individual problem-solvers, communicating both constraints and partial solutions to the problem to one another, thereby contributing both to a solution of the problem at hand (navigating the ship) and an understanding of the problem (how the instruments had failed).

The social nature of making joint decisions, contributing both partial solutions and partial problems to the group, presents certain requirements to support system builders [1]. However, Ackerman argues that the more advanced technical support systems for decision-making we devise, the more brittle they seem to become due to their rigidity in defining how to interpret and present the world, how to define roles and authority and how to communicate with others [1].

It is not the case that planning as a way to organize actions is inappropriate for all human endeavors according to Suchman, Winograd, Flores and others. For some problems, the act of specifying a partially ordered sequence of well-defined actions is very productive. Managing the logistics of air transports that go on regular schedules with tight regulations is a problem which lends itself well to the type of ordering that Game Theory stipulates [85]. Problems which can be solved by more or less automated methods are called *tame* problems, in contrast to *wicked* problems [130]. Understanding when a problem is to be considered tame and when it can be considered wicked is thus of great importance in support tool design.

### Wicked problems

Wicked problems were first described in the social sciences [130] when it became apparent that professionally crafted solutions to complex societal problems seemed bound to fail. The concept of wickedness has to do with the problems inherent in formulating the problem: a wicked problem is, by definition, one that is resolved by being formulated in a certain way. There is no single solution to a wicked problem, because people are bound to approach the problem with different sets of values, where some may not even see the problem as a problem at all. Thus, a resolution, or an acceptable compromise, seems to be the best a decision-maker can offer. Framing and re-framing a problem using different sets of concepts is central to managing wicked problems, not formulating plans for optimal solutions. For problem instances in command and control that we consider to be less tame, other tool support than automated planning systems is relevant.

Researchers on cognition have advocated communication tools based on structured arguments as one approach to support successful management of such wicked problems [105]. Facilitating team communication and understanding through shared visual representations has also been proposed in general [36, 60] and specifically in the context of command and control for space flight applications [80] and in military settings [110]. An early attempt to facilitate the communication of team understanding was concept maps, that use boxes and arrows between boxes to denote concepts and relationships between concepts [120]. These shared representations are believed to be useful in particular if the shared constraints that the participants reason about regarding problems are made visible to all [146].

### Command and Control Performance

When characterizing command and control as a process through which commanders attempt to make sense of a situation, the performance of a command team can be understood as the level of understanding they have of a situation and how to act. Such understanding may be difficult to qualify and in particular to quantify. Other approaches to characterize command and control are no less challenging to quantify. Command and control can be understood as a continuous process for decision making [121, 129], a process for sensing the environment [3], as a joint cognitive system integrating people and machines [77], a system for distributing functions among actors [76], for communicating intent [138], as a structured workflow among a set of actors [159] or in terms of the specific psycho-social aspects of a command team [27, 8]. Depending on the perspective, different methods and tools are required to understand staff work and evaluate tool support.

Although all of the perspectives offered by all these approaches to studying and explaining command team behavior may have their distinct advantages for understanding command and control, the abundance of theoretical concepts for understanding $C^2$ may in itself present a problem for the evaluation of the effectiveness of intelligent support systems. In the following sections, we shall describe

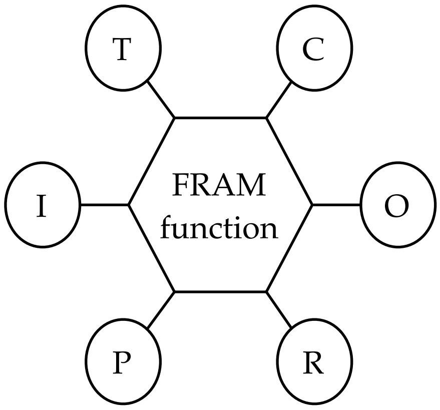


Figure 3: The FRAM hexagon, representing the six aspects of functions that are central to the FRAM representation of cognitive functions in joint cognitive systems. The hexagon represents a single cognitive function that can be coupled with others in a larger system.

a set of such intelligent support systems for military command and control, as well as concepts of support systems that are used in other settings but not in military command and control.

### Functional modeling

The process of command and control can be described as a set of abstract functions performed by either humans, machines of combinations thereof. Command and control can thus be seen as a socio-technical system which performs joint functions and is best described not in terms of what either machines of humans do as part of that system but what components based on combinations of humans and machines do [77]. Modeling a battle command staff as function allocation [76] could yield models for how functions of man-machine configurations affect one another in a network of constrained actions [75]. Hollnagel introduced a method for visually representing such constraints in the FRAM model [75] which has been applied to both aviation safety and command and control [165].

Using the FRAM analysis method, researchers model six principal aspects of each function as seen in Figure : input, output, preconditions, time, resources and control. Resources are typically energy of material that is consumed or transformed when performing a function, and control represents that which controls the performance of the function, such as a plan or a guideline. These aspects can be linked to other aspects of other functions, so that the output of one function may be the input, resource, precondition or control of another. Once modelled, this manner of visualizing functions can help visualizing constraints, and thus the conditions for system performance. Cognitive systems researchers argue that the identification of how aspects of functions affect other functions, commanders are better equipped to reason about the constraints they operate under. In command

and control, such constraints can include the coverages of radar sensors, movement radii of military units or communication bottlenecks. Several projects have developed decision aids that aim directly at bringing those constraints into the light for commanders [122, 59, 4].

In general, however, the concept of decision aid, or decision support, is very broadly defined as any system which helps users in their work process towards making decisions on the planning and execution of actions. They can be classified according to the theories of analysis that they adhere to, and the issues that are managed through the system.

### General decision support systems

In an overview of the state of the art and practice of decision support systems of 1989, Andriole provided an ordering of decision support systems according to their purpose and the structure they add to the decision-making process [7]. Their stated purpose could be to:

1. define and structure problems
2. collect, fuse and filter data sources
3. generate courses of action
4. select from a set of possible options

This list presents different possible approaches to decision support systems depending on structure inherent in the problem domain at hand. Another characterization of decision support systems comes from Management Science, which is concerned with corporate strategies and theories of how large corporations work. Clark provides an overview of decision support systems for Management Science where the systems are categorized according to the activities they are intended to support [37]. Interestingly, the four stages of activities that comprise planning in the corporate domain (analyzing the environment, planning direction, planning strategy and implementing strategy) coincide well with the description of the Observe-Orient-Decide-Act model (Figure 1) developed for military command and control. Clark describes how most decision support tools for corporate planning until 1992 had been concerned with forecasting and were used to support either the generation of plans, the implementation of them (through the formulation of plan documents for instance, or as part of control systems) or the analysis of the environment. Only few concerned the evaluation of what missions, objectives, values and expectations to focus on. Pinson et al describe, in a paper on a system to support strategic planning in corporate settings for high-level corporate managers [123], how corporate strategic planning is considered to be a linear process model consisting of four stages as also proposed by Clark, but Pinson also stresses that the decision-making process is ill-structured. Pinson describe the construction of a decision support system to support corporate strategic planning through the use of an automated planning system. Their example problems described is *Enterprise Expansion*, which their system proposes a automatic solution to through a sequence of abstract actions beginning with *Research Market Segmentation and Target Segment Viability* and *Report Consumer Attitudes to the New Product*. Goals can be hierarchically ordered and dependent on one another, but based on a fixed repository of known actions. The ill-structured nature of decision-making is not discussed in relation to their system.

In Andriole's general description of decision support, systems for helping decision makers have performed best when targeted at well-structured problems:

> Generally speaking, decision support system are more successful when targeted at [structured problems] and progressively less successful when targeted at less structured, ill-defined problems such as strategic planning and tactical operations [7, pp 8].

However, $C^2$ researchers have not characterized command and control in general as consisting of well-structured problems, even as most military decision support systems have concerned somewhat restricted domain of a course of action generation through automated planning systems and selection from a set of possible options in mixed-initiative planning systems that help commanders by requesting them to supply unknown values for the calculation of automatically generated courses of action.

## Automated planning

Planning systems are here defined as systems designed to produce a formally sound, partially ordered sequence of actions that start at an initial state and end in a final, desired goal state has been reached. During the course of planning, that is, producing a set of actions, an approach based on automatic planning may cede control to a user who fills in information required to produce the final plan, in which case the system is called a mixed-initiative planning system. Other systems have capabilities that include support for execution monitoring or communication apart from planning, which we shall call *combined systems*. These systems representare both research prototypes and working systems employed by armed forces for planning and monitoring aspects of military operations.

### JADE

JADE is an operational system for force deployment planning, and especially force composition [111]. Force composition can be defined as the problem of determining which military units should be assigned which tasks in a battle plan, depending on the abilities and the constraints that can be assigned to each unit. JADE integrates data on force capabilities from software called the Force Management and

Analysis Tool (ForMAT), which can be used to create custom-tailored groups of military units for solving individual assignments. Such may consist of command units, maintenance units, defensive units and units with different offensive characteristics. Together, these units may form groups suitable for particular tasks. The problem which JADE solves involves how to assign optimal groups for larger operations where forces may support one another and have different assignments over time. To accomplish this, JADE provides an interface to several legacy data sources for information on airports, seaports, military locations and military units. The information that JADE receives from these data sources, along with a user's intentions regarding mission goals to an automated planner.

### CTAPS

The Continency Theater Automated Planning System (CTAPS) is the collective name of a group of components for providing an air tasking order, which coordinates air missions that are to be performed over a relatively short time period [67]. The CTAPS system has been fielded within the US Air Force and, similarly to JADE, has been used to integrate several steps of a process involving manual labor and rigorous procedures of calculating flight schedules for airborne missions. The CTAPS system has evolved from a tool to create a prioritized target list into a comprehensive software suite for producing complete plans for air missions called Theater Battle Management Core System (TBMCS) [81]. CTAPS at its heart provides a software for decising which targets to attack with air units. The target lists that become the end result of the process of using CTAPS are usually subject to change during the execution of targets, either because of changes in evaluations of prioritiex, or because new information becomes available that makes some targets less interesting than others, or because breakdowns or losses force planners to re-evaluate their plans. The authors of CTAPS outline some suggestions for how replanning in such circumstances could be achieved, including the option of letting units coordinate their own efforts to a larger degree, and have more freedom and control over how to manage conflicts in case of replanning needs. Interestingly, the option of providing such freedom, which would be in line with the vision of Network-Centric Warfare, was, at the time when CTAPS was constructed, considered to expensive in terms of the required bandwidth for communication between units on the same level in the chain of command. Instead, the authors describe how CTAPS could be made even faster as a planning system so that incremental changes could be performed during missions, or that such changes would have to be postponed until the next planning cycle due to the costs of cancelling operations that have already begun.

### ARPA-Rome Planning Initiative

The ARPA-Rome Planning Initiative (ARPI) was a research effort to radically enhance the capabilities of automatic planning and scheduling so that these research

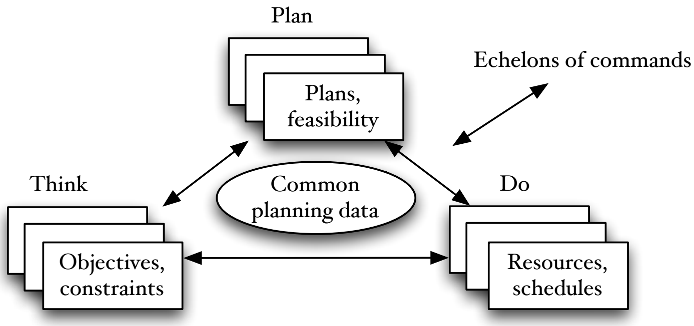


Figure 4: A distributed, collaborative planning process.

areas would address real-world problems faced by military commanders [19]. The initiative was divided in three tracks: a research track, a technology transfer track and a demo track [79]. The research track was to focus basic AI research efforts on large-scale, realistic problems faced by military commanders. The program also recognized that the model of crisis management as a sequential process akin to the OODA loop would not be appropriate to describe the complexity of a distributed, collaborative planning process. Instead, due to the nature of planning as a collaborative venture, in which shared data sources are used, manipulated and populated during the planning process by several command functions, the initiative stressed that AI systems deployed for helping commanders in their work should support collaboration. The research track aimed at bringing metrics-based evaluation to applied AI research, focusing on large-scale, shared problems. Also, the project recognized that finalized plan products were not the central aim of the command process, but that communication of intentions and rationale was more important. Specifically, the products that emerged from the initiative were technological showcases of new planning and scheduling systems that generated plans with new abilities to visualize the sequence of actions that form a plan and use previous cases as the basis for new plans. Several specific applications were developed as Integrated Feasibility Demonstrators (IDFs) that provided automated planning capabilities for force deployment, evacuation operations, and transportation planning.

### ACPT and O-P$^3$

As one part of ARPI, researchers created a planning tool for air campaigns by using a Decision-Centered Design Approach to incorporate human judgments in the planning process. The resulting tool, ACPT [107], was developed as an attempt to make assumptions and human heuristics used in planning explicitly known to an AI planner. One such assumption that could be made of the planning process was that planning is a collaborative activity with shared representations that are manipulated, created and dissected continuously by several parties. In the Open Planning Process Panels system (O-$P^3$) [99], Levine, Tate, and Dalton developed a system for collaborating during the planning process and sharing different parts of the products created with one another in a manner defined by the Planning Process Panels. Using the O-$P^3$ model, the ACPT planning system was adapted for collaborative planning, where the augmentation had three purposes:

1. To make explicit which part of the planning process every user was in,
2. allow users to compare and evaluate planning products, and
3. control the next step in the planning process, given the information from the current step.

The ACPT planning system was used to automatically create a comparison matrix of different military Course of Action (CoA) plans, together with a flowchart tracking the military planning process used to create plans. Following the test case with air campaign planning, the O-$P^3$ model has been extended and used for case-based, collaborative, mixed-initiative planning for emergency management [152, 125] under the name I-X Process Panels.

### HICAP

HICAP [112] is an integrated environment with several planning aids that let users create a hierarchical task network through a task network editor prior to planning, so that an HTN planner may plan accordingly. The system also included a conversational case-based planning tool called *NaCoDAE* [26]. Through the task network editor, the user is guided through a set of questions of how a task may be planned and which options there are.

The authors of HICAP were guided by a set of requirements that they had found necessary to implement to ensure success for a planning tool. They found it should be:

- *doctrine-driven* so that doctrine task analysis would guide the plan formulation process.
- *interactive* by letting the human planner edit a plan interactively.
- *provide case access* to previously created plan segments.
- *perform book-keeping* of task responsibilities for the force elements available for planning.

### CADET

CADET [87, 69, 86] is a planning tool by Kott et al. which has been used experimentally in an integrated environment featuring several different views, different input mechanisms as well as the use of a mixed-initiative planning system.

It is a planning application that can act on its own as well as guided by a user. At its core, it is an automated AI planner which uses task decomposition as its main strategy for planning, much like Hierarchical Task Networks (HTN) planners do. CADET also offers the possibility of either creating a plan in a mixed-initiative fashion or to generate a prototype plan directly from a high-level CoA description. With the latter option, a human planner is expected to verify the results afterwards and correct the results of wrong assumptions. The CADET developers found that, in an evaluation with real commanders and real exercises as a basis, their tool compared well when put against the performance of human planners with no help [128, 86]. The quality of the produced plans was determined by human judges to be on par with, or even slightly better than those produced without the help of CADET. Also, in their testing scenarios, Kott et al. found that it took significantly less time for those who had used CADET compared to those who had no tool support at all.

However, there were some problems with the CADET system. Most notably, the developers had decided not to develop the interface for presenting the results of CADET more than to have a table with all plan information (a *synchronization matrix*). This representation turned out to be troublesome for users to interpret, although synchronization matrices were something they were accustomed to from their work. Kott et al. speculated that this may be due to that someone else's (CADET's) ideas may require different visual representations to make sense than the representation you may produce for your own thoughts. Another finding which is of interest to us is a reflection made by users in an integrated experiment where the CADET tool received information from a sketching application (*NuSketch*) [58] and a natural language interpreter (*CoA Statement Creator*). When the users had received a plan from CADET and modified the resulting synchronization matrix, they wanted CADET to means to reflect these changes in the other views of the plan. Phrased differently, one could say that the users wanted to have all views (the sketching application, the statement creator and the synchronization matrix) *interconnected* so that NuSketch and CoA Statement Creator would not only be used to *create* planning information, but also to *present* the resulting plans.

### Mixed-initiative planning systems

Apart from the systems above, several others have been developed as planning systems for military command and control and similar applications. One class of planning systems that have aimed at overcoming the limitations of AI planners that must encode all domain knowledge completely and consistently is *mixed-initiative planners* [29, 142]. Initiative in general, and in particular what *mixed* initiative means, can be defined by using different reference points. Cohen presents an overview of four different definitions of initiative used by designers of mixed-initiative planning systems [39].

- The first definition regards the flow of conversation as the object to control when taking the initiative: either the user controls the dialogue by asking questions and making demands to the planning system, or the planning system may ask the user for clarifications and resolutions to conflicts.
- The second definition relates initiative to controlling the formulation of goals and tasks during the problem-solving process, and where the interaction between user and system is not regarded.
- The third definition combines the first two, by defining initiative as presenting a goal to the other party in a conversation, making initiative situated in a conversation but contextually limited to the formulation of goals.
- The fourth definition presents a slightly more advanced view of control in which a user takes initiative if she takes the first turn in a goal-oriented process, where a dialogue may consist of several such processes.

Cohen presents TRAINS as an example of a system that uses definition 1 (see Section 2), Traums work on agent systems as an example of definition 2 (e.g. [156]), Fergusons and Allens work on conversational agents that act independently of the problem solver would be examples of definition 3 (e.g. [54]) and provides definition 4 as a possible superset of the other three which none of the reviewed systems had implemented.

Burstein presents a survey of issues important to consider when constructing mixed-initiative planning systems, which do not relate directly to theoretical definitions of initiative but rather consider features inherent in planning systems that are important to develop or manage [29]:

1. *Plan-space search control management*, which means to coordinate how different agents in the planning process (humans or machines) search for solutions to planning problems.
2. *Representations and sharing of plans to communicate intents and ideas*. Humans often represent plan information using only texts and graphics and to make it easy to process the information provided by an automatic planner, system designers need to consider a representation which is most convenient for humans
3. *Plan revision management*. Several revisions of plans may have been developed in parallel, which presents the mixed-initiative planner with a problem of how to manage these together, by merging or otherwise.

4. *Planning and reasoning under uncertainty*. A principal reason for an AI planner to include a human partner in the planning process is to manage uncertainty, but exactly how can the AI planner make use of a human's appreciation of, and management of uncertainty?

5. *Learning*, to ensure that an AI planner learns what is expected as output.

6. *Inter-agent communication and coordination*, to make sure that plan revisions and modifications can be communicated at all, in a unified manner between all agents.

**TRAINS**

One of the earliest mixed-initiative planning systems, developed for a logistics domain, was TRAINS by Ferguson et al [55]. In TRAINS, users interact through multiple modes with a route planner for trains, both through a map and a domain-specific natural language dialogue. When designing the system, Ferguson et al realized that command and control scenarios, even as well-defined as train transportation, were difficult to models since there are huge sources of potentially useful information that can enter a planning problem and it is not until the act of planning that human planners decide on which aspects are useful to consider and which are not [55]. Also, they noted that rather few pure automated planning systems were being used for logistics planning due to the fact that the problem to plan for was often defined as part of the problem-solving process, and that option exploration was more critical to success in the planning process than creating complete and consistent plans. In later work, Ferguson and Allen propose the use of dialogue systems for collaborative problem solving, where the computer system which the dialogue systems provides an interface against is not necessarily an automated planner but possibly other problem-solving agents [54].

**Case-based plan re-use**

Another early mixed-initiative planning project was the OZONE project [142]. In the OZONE project, the researchers strived to provide support for solving practical planning and scheduling problems, which traditional AI planner had neglected. In their critique against AI planning systems, Smith et al stated that AI planners forced users to adhere to the conditions and the formalism of the planning system rather than the other way round. In particular, they claimed that the iterative nature of refining constraints and goals during planning, the integration of planning and resource allocation, and the reuse of earlier solutions to typical problems was not supported by earlier systems. For the OZONE system, they developed an ontology of concepts to be used both by several parts of the planning system but also for reusing components of earlier planning and scheduling sessions. In their ontology, they defined the concepts of demands, activities, resources, products, and constraints. The concept of plan reuse by case-based seeding of a planning system was tested by Gervasio et al for crisis response planning [63] and by Breslow et al for a general, conversational plan development system (NaCoDAE) [26].

**Plan sketches, templates and meta-theories**

Myers provided an early architecture for a mixed-initiative planner which would flesh out a user-supplied plan sketch, consisting of partial information about goals and constraints, into a complete plan [114]. For users to understand better how to influence plans, Myers suggested the use of a domain meta-theory in planning, where general concepts of domain descriptions could be made more easily available to human planners [117]. In a 2000 paper on domain meta-theories, she suggested that user involvement in the planning process was critical for the success of using automated planning systems, and that an ontology based on the concepts of roles, features and measures could be used to describe the planning from a human perspective. For example, with a planning operator move with two parameters (location1 and location2), the role of location1 could be *Origin*, whereas the role of location2 might be *Destination*. A feature could be used to differentiate operators from one another, so that a movement operator requiring location1 and location2 to have ports would be described as having the feature *Water transport*. A measure would be a partial ordering which could be applied to features, with *affordability* as a measure that could value *Water transport* higher than *Air transport* and *time-efficiency* providing the reverse order. These concepts, and their intended use in a planning scenario, convey a clear conceptualization of users' role in planning: to direct and influence an automatic process of using a given set of planning operators to lay out an ordered sequence of actions in the form of a plan. In later work on the CODA system by Myers, to be used in the application domain of military Special Operations Forces mission planning, the conversational and collaborative aspects of planning were emphasized more than domain-modeling [113]. Myers also stated that the strategic nature of planning for Special Operation Forces would effectively prevent complete formalization of the domain. However, plans were seen by CODA users as useful tools to communicate ideas and that collaboration in planning with shared, limited resources benefitted from a unified system for creating and sharing plan fragments. In later work on the PASSAT mixed-initiative system [115], Myers et al put the user in focus by using templates of old plans, formulated as hierarchical task networks [52], and a tool for sketching plans by combining parts of old plans into new ones[116]. All constraints encoded as part of the domain in PASSAT had to be resolved before invoking the planner, so no violations of constraints or unknown information would be allowed.

**Other mixed-initiative systems**

The Jet Propulsion Laboratory of California Institute of Technology has designed a planning system (ASPEN) [126] specifically for the purpose of iterative repair, when a certain subset of tasks are common to several operations but others need to change with changing conditions and requirements. ASPEN has been used for managing space missions where timely and accurate planning of similar tasks is critical. One specific property of the ASPEN system is to identify conflicts in resource usage and how they affect a plan [141].

Another mixed-initiative planning system for command and control is ComiRem, which could be defined as a constraint manager as well. The system contained a set of known time and resource constraints [143] regarding the conduct of special forces missions and allowed a user to enter information on when certain tasks should be completed. The system is used to maintain temporal restrictions among the tasks laid out by the user. The system performs continuous checks on the feasibility of the plan as the user performs structured modifications. The range of possible modifications is determined by the internal model that ComiRem maintains over possible degrees of freedom in the scheduling of special operations.

For the civil emergency The SIADEX system was constructed with three [53]

Mixed-initiative planning: [40, 73, 82, 25, 57, 5, 17, 12, 126, 141, 154, 147]

Using logistical problems of goods deliveries to customers, Anderson et al describe how directing how search heuristics are employed, and what parts to keep of a solution when generating another one keeps humans tightly in the loop of handling the planning problem [6].

CoA Planning: [99, 100, 71, 18]

## Critiquing

Critiquing systems is another class of support systems devised to complement human decision makers instead of replacing them, as earlier expert systems had attempted to do [118]. One of the principal arguments against expert systems in the 1980s was that these systems attempted to incorporate all procedural and relevant domain knowledge required to automatically diagnose and solve problems in expert domains such as medical diagnoses [68]. The main underlying assumptions that guided the development of early knowledge-based systems, expert systems, came from the cybernetic tradition of viewing knowledge as something which can be encoded as packages and transmitted verbatim between sender and recipient [161]. Knowledge is, in the cybernetic tradition,

- composed of universally true pieces of information that are interpreted by agents to form larger structures, and
- separated from the subject knowing something.

Such a view of knowledge stands in contrast to the view presented by philosophers such as Heidegger, according to whom knowledge is intimately coupled to the subject knowing something and intrinsically non-transferrable [74]. Human beings are, according to Heidegger, intimately coupled to the world in which they live, and think through the artifacts the use and the context the live in. Thus, concepts in the mind of one person, made concrete through the action of speaking and writing, sets in motion a process through which may the concepts attained by another individual are changed, but not in the way the cybernetic would dictate. According to a general critique against AI approaches to knowledge reasoning and problem solving, Dreyfus argues that to frame the world in concepts that are relevant to reason about in contrast to those that are not relevant to reason about is fundamentally flawed [47]. Human intelligence, according to Dreyfus, cannot be de-coupled from the reasoning apparatus that performs it, and artificial intelligence would therefore require complete, low-level emulations of the physical functions performed the human brain. Apart from such radical critiques, even the more moderate critiques of expert systems noted that human knowledge is difficult to encode, although there could be reason to believe that systems for identifying clearly defined errors could be helpful in notifying people of what they might have overlooked [70].

A critiquing system makes the user aware of faults and possible improvements or disregarded stimuli. Originally, *critiquing* as a concept originates from research on Expert Critiquing Systems [139], where researchers developed means for physicians to evaluate their courses of action before proceeding with medical treatment [106]. These critiquing applications were developed as an improvement over expert systems (i.e. automated problem solvers which suggest treatment) to overcome limitations with automated systems. It was recognized that *trust* was an even more important issue than the issue of performing correct inferences from large sets of medical knowledge, which relates to the arguments by Rittel and Webber regarding the conceptual difficulty of producing plans as a means of dealing with wicked problem [130].

The need for trust led to the development of expert critiquing systems, which are in essence relaxed versions of expert systems that only analyze user solutions to problems and notify users if their solutions are potentially dangerous or wrong. By allowing users more control over the problem-solving process, it was noted that expert critiquing system were much easier to use and trust.

Silverman and Wenig outlined a method for constructing expert critics, which mostly concerns the knowledge acquisition process [140]. In their presentation of a method for acquiring knowledge of human problem-solving knowledge, they outline five hierarchically ordered types of questions:

1. Domain Task-Structural Questions (e.g., what are the types of tasks that occur?),
2. Normative Cue Utilization Questions (e.g., what is the relevant universe of cues?),
3. Missing Concept or Misconception Questions (e.g., what cues are experts focusing on, and which of those are relevant?),

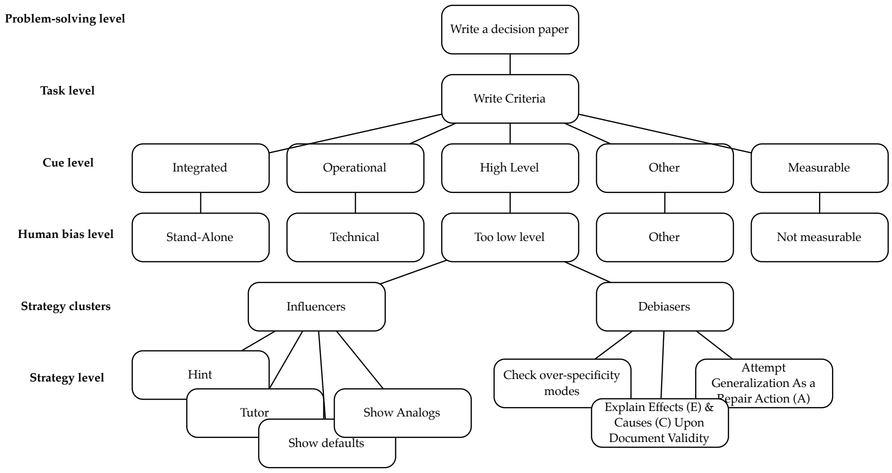


Figure 5: An example of an expert critic ontology according to Silverman and Wenig.

4. Influencer, Debiaser Tutor and/or Director Questions (e.g., how should the system better cure the user to dampen his bias), and
5. Fine Tuning Questions (e.g., how should bias and strategy information be modulated and presented to enhance the collaborative relationship?).

These questions, when using during the knowledge acquisition process, would provide the foundations for building sound critiquing systems according to Silverman and Wenig. As an example of how these questions could result in a specification for an expert critic, Figure 5 shows a tree where answers to each type of questions are ordered according to the sequence outlined. The scenario described relate to writing decision papers explaining the rationale for procuring military equipment, and justifying the evaluation made as part of the decision. The relevant cues for determining that a decision paper is correctly written, as determined by an expert on writing procurement documents, are categorized with the adjectives Integrated, Operational, High Level and so on. Paired with each of these normative cues are possible human errors that are likely to be committed, and these in turn are coupled to strategies for repairing each type of error.

Given this outline of questions to guide the knowledge acquisition process in constructing expert critiquing systems, the domains in which expert critiquing systems are relevant are the ones in which these questions make sense and provide clear answers if asked to domain experts. In the context of command and control systems, critiquing systems have been deployed mainly to support the production of plan documents. We provide examples of such systems in the following sections.

### INSPECT/EXPECT

INSPECT was an air campaign planning tool with critiquing support [157]. As a basis for its critiquing, it used knowledge entered through the EXPECT knowledge capture tool and was primarily used to test plans for structural correctness with respect to decomposition of tasks and resource assignments. It used a textual and form-based representation of plans and presented the user with a dialogue-like textual interface. The cues concerning air campaign plans, which INSPECT was responsible for bringing to the user's attention, concerned the structural correctness of the products involved. INSPECT was developed as an add-on to ACPT (see Section 2) and provided inspection of aspects that related to the completeness of an air campaign plan. As examples of how an air campaign plan could be inspected and criticized, the authors of INSPECT describe a set of issues that INSPECT tries to resolve:

- *Objective with no child*, meaning that a goal (node) has not been decomposed into tasks to perform to achieve the goal,
- *Objective with no parent*, meaning that a task (node) does not serve a higher objective,
- *Objective has no measure of merit*, meaning that it has not been stated how to evaluate when the objective is reached,
- *Objective with no structured specification*, meaning that the objective is not defined using the language known to INSPECT, and it can therefore not be further evaluated,
- *No objective fulfilling the basic tenants of air power*, meaning that there is no description of how to achieve one of the basic prerequisites for conducting air operations,
- *Objective with too many parents*, meaning that a task relates to too many higher objectives, possibly indicating a vaguely formulated task in need of specification
- *Incompatible sequence restrictions*, meaning that the sequencing of tasks may be incompatible, so that tasks A and B have to precede one another B simultaneously
- *No primary aircraft available for an objective*, meaning that resources which should primarily be used for the stated objective are not available,
- *Incoherent decomposition*, meaning that the a task is described using more general terms than the parent objective it is related to.

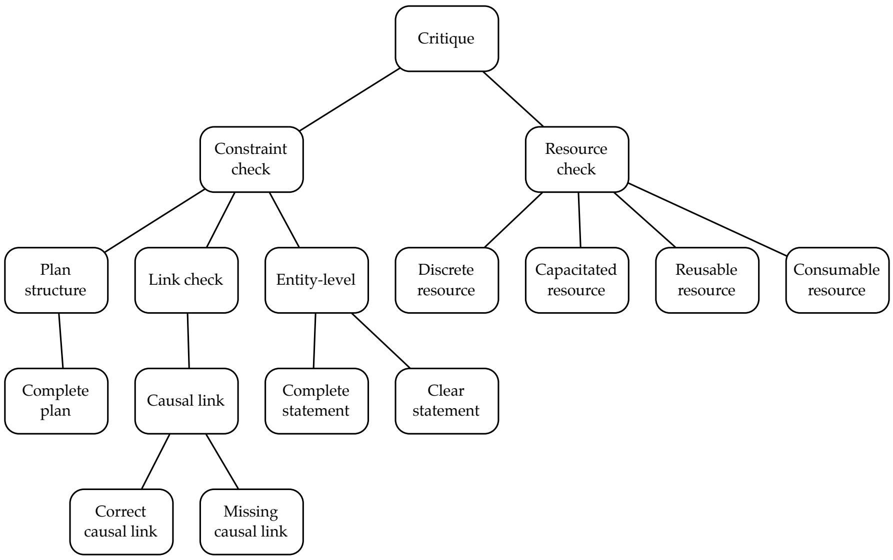


Figure 6: A critic evaluation ontology as described by Blythe and Gil

## Knowledge Acquisition for Critiquing

INSPECT provides a concrete example of a system that attempts to help users correct structural errors in air tasking orders. Other approaches to provide critiquing as a means of decision support have studied what knowledge that is actually useful to model as part of the critiquing knowledge base and how to extract it. Specifically related to Course of Action critiquing, Gil et al present the EXPECT method of eliciting expert knowledge [20]. They describe a method for adding new critiquing knowledge by starting from a generic set of evaluations and add new evaluations for each domain. They describe how the structure of evaluation follows a generic pattern, much as Silverman states that the structure of a constructed critic follows a particular, hierarchical pattern [140].

Figure 6 describe a critic evaluation ontology as proposed by the problem solving method in EXPECT. A notable difference compared to Figure 5 is that the relationship between correctness and incorrectness is less straight-forward: *missing causal link* and *correct causal link* are both related to the concept of a causal link. Also, the ontology for *how* to provide feedback and evaluate plans is treated as a completely separate ontology and not defined in relation to *what* to evaluate. The approach demonstrates an alternative strategy for knowledge acquisition, although still using the concept of an ontology for evaluating plans and products.

In the Asgaard project [136, 137], ontology researchers built a support tool for critiquing in medical settings based on an ontology of medical guidelines and patient records. They also describe a set of questions that need to be answered as part of the knowledge acquisition process, although these are questions are ordered according to a set of corrective tasks that the system is supposed to perform, such as verification of a guideline, the applicability of guidelines given a patient, the recognition of intentions and so on. They also define a strict ontology for when certain parts of a medical treatment plan should be put into action and when not.

### The HPKB and RKF Programmes

As a response to the difficulty of early knowledge-based systems to provide enough common-sense guidance to commanders in military command and control, DARPA launched first the High Performance Knowledge Bases (HPKB) and later the Rapid Knowledge Formation (RKF) research programmes, aimed at radically improving the knowledge bases available to both existing and future decision support systems. As a result of the programmes, ontologies [65], ontology construction systems and rule construction systems tailored to command and control were developed.

Both research programmes were highly technical in their focus, driving research on artificial intelligence systems through the construction and evaluation of ontology- and rule-related systems. One of the first systems created as a result of the HPKB initiative was Disciple, a knowledge-acquisition system developed for military applications and especially for strategic analysis of enemy center of gravity [153]. Disciple was essentially a system for instantiating a general ontology of Center of Gravity concepts with scenario-specific information that students in an army course were supposed to enter by themselves in an editor, provided a description of the scenario in free text. The Disciple system would use scripts for asking questions to the course participants and eliciting assumptions they made. The Disciple system mixes natural language dialogue with pre-defined terms that are specific to the scenario to ensure consistency as the human expert adds steps in his or her reasoning process regarding what should or should not be considered a center of gravity. Disciple can guide an expert in producing rules regarding how to determine what should be considered the center of gravity in pre-defined cases, but the rules produced were not always consistent or optimally formulated. However, the formal approach to learning about center of gravity was appreciated by the participants in the studies undertaken with Disciple.

In an RKF effort to enhance rule-acquisition from human subject experts, Pool, Gil and others evaluated the SHAKEN and KRAKEN rule elicitation systems for Course of Action analysis [124, 66]. Military experts on Course of Action analysis were asked to enter their knowledge regarding how they would conduct and evaluate suggested courses of action in the SHAKEN/KRAKEN knowledge acquisition tools. Similarly to the Disciple approach, the authors noted that core concepts in ontologies represented in description logics-related languages [145] such as sufficient and necessary conditions when describing classes of objects were difficult to grasp for military experts. When provided with concrete examples of how their own rules for evaluating courses of action could be represented as rules in propositional logic, however, they could more easily relate to the formalism required by the knowledge acquisition tools.

The rules and ontologies developed with the tool sets from these two research programmes were evaluated in the same context they were developed in one case, and not evaluated at all in the other. The Disciple ontology regarding what constitutes a center of gravity represented one, hard-coded version of a human expert's evaluation and was intended to be used when training others to use the exact same evaluation criteria, regardless of whether this would make the course participants more apt at making strategic decisions in the face of new situations. The SHAKEN/KRAKEN systems were evaluated on the basis of the ease with which subject-matter expert could produce formally sound rules, not based on whether these rules were useful in any larger context.

The PLANET ontology, which was developed as a plan ontology for the purpose of inspecting or more efficiently generating military and other plans, was evaluated based on the percentage of concepts in the ontology that were considered similar to concepts in existing plans developed without PLANET [65]. In their paper on the development and evaluation of PLANET, Gil and Blythe note that

> Ontologies are generally accepted to be useful just on the basis of their existence.

### The roles of ontologies

Computer-represented ontologies are generally used to model a restricted aspect of the world using a formal language derived from Description Logics [13]. Ontologies have been considered a necessary prerequisite for building systems that are capable of providing support for humans doing tasks that require much background knowledge, such as planning military operations [64, 127].

Gil [64] describes three distinct taxonomies for in planning scenarios: *action taxonomies*, *plan taxonomies* and *goal taxonomies*. Action taxonomies can be used to relate individual plan actions to each other, so that *Speech Acts* can be declared to be special forms of *Communication Acts*, and that the actors involved must be capable of speaking and hearing. Using such a declaration of action properties, an automatic planner or plan authoring tool may deduce how plans can be constructed. Plan taxonomies can be used to classify solutions to a planning problem (plans) by relating them to each other with respect to their solution space. That way, a planner can search more efficiently for solutions and make sure to not search within a given solution space more than once. Goal taxonomies are used to relate goal statements and method capabilities in the action taxonomy. For example, there may be no individual action which achieves the goal as expressed by the user, in which case the reasoner needs to establish if there is possibly a set of actions which together may achieve the desired goal state, or if several objects need to be involved when performing actions to achieve the goal. Such *goal reformulations* are possible using the *EXPECT* planning system [150], and allows for efficient translation mechanism between goal statements in human-readable form and the domain representation accessible to the planner.

The approach of creating formal ontologies so as to statically structure taxonomies of actions, plans and goals, and enumerate all possible types of entities and their relationships in planning, has been challenged by other, however. Those who criticize the very utility of constructing formal ontologies argue that the requirements for formal correctness make the systems created too narrow to be of real use [15]. Formal ontology construction can in this respect be put in contrast to the *hermeneutic* approach to knowledge, where the act of constructing knowledge structures defines knowledge [31] and where knowledge is not so much a noun on its own but an attribute of people knowing something, and that knowledge therefore cannot be considered without reference to explicit users and their context [102].

In an alternative approach to building system ontologies, Fonseca and Jasist argue that the very foundations for building ontologies is flawed, as different perspectives on concepts are inconsistencies for a formal ontology, but can enhance appreciation of a problem for humans[2] [56]. Ambiguities in formal ontologies must be resolved, whereas ambiguities for humans can signify that a concept simply is ambiguous. The primary criticism of Fonseca and Jasist against formal ontologies is that the domains in which they can be formulated must necessarily be very narrow for them to be as consistent and complete as required by formal reasoning engines, and therefore, they risk to be brittle and of very restricted use. In their outline of how to construct systems for helping people in knowledge-intensive tasks, Fonseca and Jasist argue that ontologies should primarily be considered tools for helping people communicate, just as Winograd and Flores propose that communication tools instead of automated reasoning tools should be developed for helping people manage problems [162].

## Machine Learning

Automated planning systems can automate parts of the plan construction process involved in command and control, and critiquing systems may be used to correct for known errors when producing plan documents. A third class of systems, which have only to limited extents in command and control [28], is machine learning [108].

By using machine learning techniques, an automatic classifier builds a model of the relationships between attribute values of instances and decision classes that they belong to. Every instance may only belong to one decision class. Alternatively, a clusterer may infer clusters, or groups, of instances provided metrics for comparison of attribute values but with no prior division of messages in classes. An *instance* is a multi-valued observation that the classifier or clusterer uses to build

[2] see Section 2 for an example of different perspectives on a problem

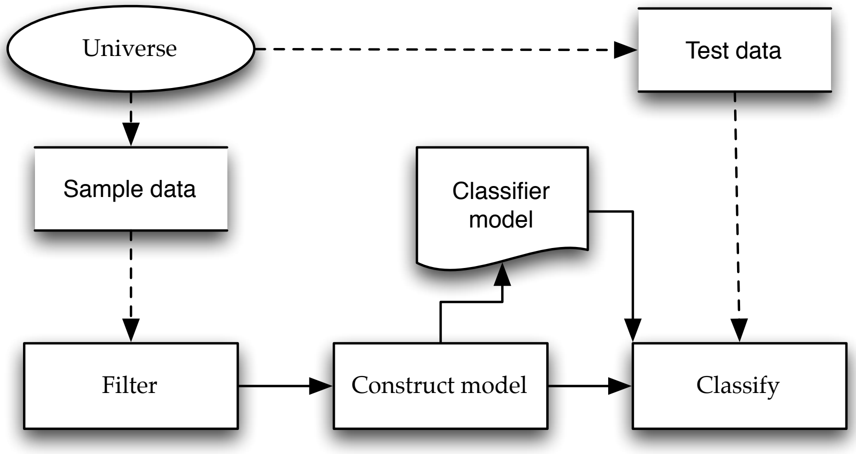


Figure 7: Machine learning as a sequence of three steps

their internal representation of instance groups. Instance attributes may be nominal (discrete and finite), textual, or numerical and, as assumed by almost all machine learning approaches on single instances to make inferences about the class of new, unknown instances. The degree to which a classifier can build a model that accurately classifies known instances is called the precision of the classifier. Generally, clusterers and classifiers are consider only single observations as the basis for patterns and not sets of observations. That is, models built using traditional Machine Learning techniques will not be able to find inter-relationships between individual measurements but only classify single measurements as belonging to one out of a discrete number of classes. Each individual Machine Learning technique may have further limitations as to what specific patterns it is likely to find.

The most prominent use of classification and clustering techniques for command and control have been in the area of information fusion, where low-level sensor data are to be converted into high-level concepts according to some standards [9]. As in the case with ontologies to support plan construction, machine learning researchers tend to define problems such as sense-making in terms of instrumental, game-theoretic terms. Arnborg et al describe how situation awareness in command and control can be reduced to game-theory by assuming that there are crisply defined expected utilities and probabilities for success for all actions [9]. With such assumptions, they go on to state that quality in decision making can be crisply defined in terms of the extent to which *objectively relevant* information is used. The justifications for making such assumptions in the light of research on command and control are not discussed. Apart from information fusion, machine learning approaches have been suggested for war-gaming, that is, the process of simulating a battle between own forces and those of an opponent [28]. Such games

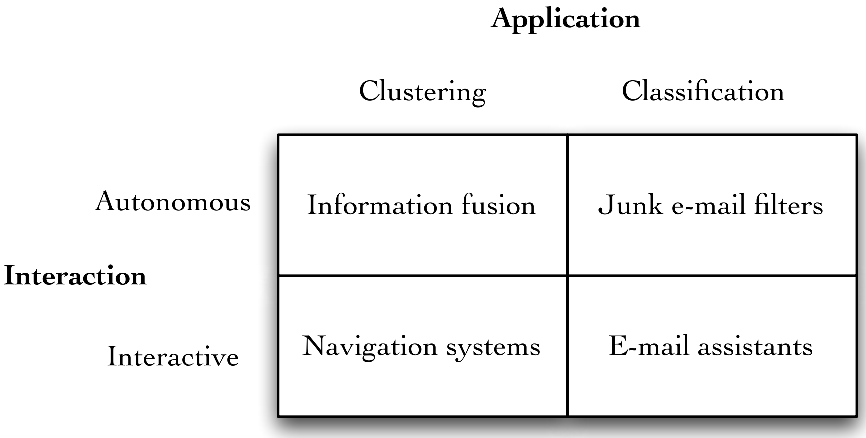


Figure 8: A categorization of machine learning system according to the interaction mode and application mode of the system

rely on game-theoretic, rationalistic assumptions of decision-making and that unknown quantities can be modeled reliably.

Machine learning techniques have also been devised for the problem of information management, both as an automated approach to process e-mail [101, 132] and as a support for navigating document collections (e.g. [42, 103]. Machine learning techniques have been devised either as separate parts of e-mail clients for classifying messages either in two classes (spam/no spam) or in a fix number of classes related to the workflow of the person using the system (e.g. [89]), but also as parts of agent-based e-mail assistants that organize messages in contextually relevant categories for users [90, 14, 151].

Depending on the domain, contextually relevant information may be found in either the structure, contents and usage of messages and documents. Mining for relevant information from large heterogenous corpora became popular and increasingly relevant with the advent of the World Wide Web [84], with applications such as personalization [48] and terminology mining [133]. Some applications have relied on the ability to extract semantically relevant concepts from the structure of machine-generated documents (e.g. [88, 10]) or workflows (e.g. [61, 160]). Other projects have also studied how to extract contextually relevant terms from messages as support for finding similar conversations based on some metrics of similarity [91, 33]. Tracking and gathering data on user behavior has been applied to personalization, where the user interface of web services can be adapted to the navigation behavior of a user [48].

Machine learning applications which require user interaction, such as e-mail tagging systems that rely on user-supplied data to complement their input and make predictions about the classes to apply to messages, are usually not labeled *mixed-initiative* as planning systems. However, similarly to planning systems, machine learning systems can be categorized in two ways: as autonomous (rule-based) compared to interactive and as systems for supporting navigation (clustering) or providing a classification.

In Figure 8 we display two different ...

## Semantic Desktops

*Semantic desktops* are inspired by the semantic web in the sense that the semantic desktop brings semantic-web technologies to the user's desktop [35, 134, 46]. A principal idea behind the semantic desktop approach is to support working and reasoning with semantic entities that are normally scattered across several different resources. One of the advantages of semantic desktops is that they promote both formal and informal work processes and flexible information flows. Furthermore, these systems avoid the rigid structures sometimes imposed by traditional dedicated information systems. However, without in-depth knowledge of the application domain (i.e., specific information about the content, structure, and purpose of the documents), it is difficult to provide semantic services beyond general document indexing and search.

# 3 Method

In this thesis, we explore the use of three different main technologies for supporting three activities in command and control. The conclusions we have been able to

## Action Research

Each iteration moves from the more controlled and structured to the less, from automation to guidance.

On the value of theories for guidance in support system development for command and control. Theories regarding systems development, functional modeling, AI planning

Exploratory research and prototyping for the purpose of provoking and understanding.

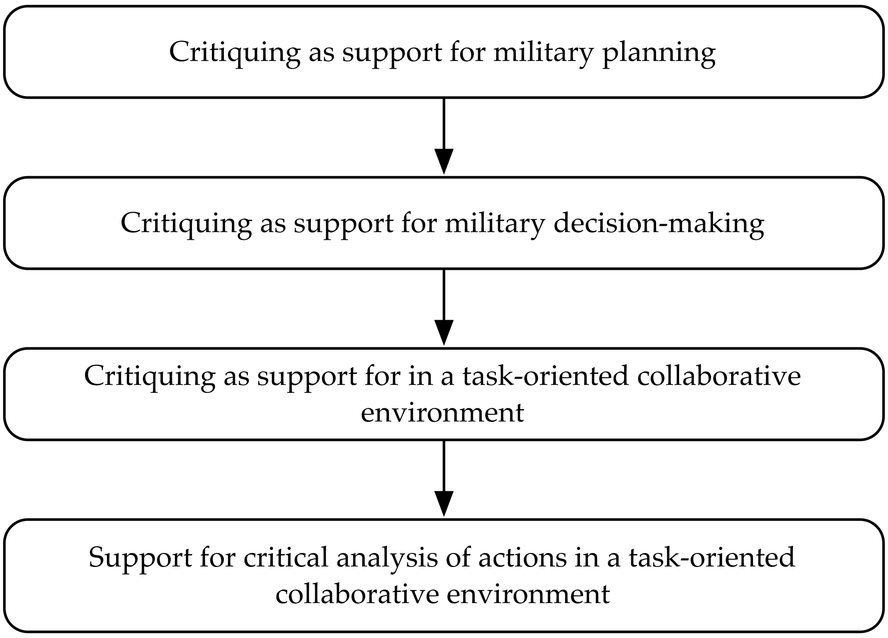


Figure 1: Four stages of defining the subject of this thesis

# 4 ComPlan

This chapter describes the *ComPlan* approach to providing decision support for commanders in the field of mission planning. In our work with the ComPlan approach, we followed the assumptions stated at the beginning of this thesis (see Section **??**).

At the end of chapter **??** we explained why all these properties were desirable and why they should guide the design of critiquing systems. Our argument was based on a survey of critiquing systems and other approaches to decision support. Chapter **??** explained our work on extending an existing experiment platform with a knowledge elicitation interface for creating critiquing rules. From this work, we concluded that planning support tools would need to incorporate well-integrated feedback mechanisms. Also, our interview study (see Section **??**) provided insights on which kinds of constraints and potential problems would be most amenable to critiquing. In this chapter, we will expand on these ideas and explain our design of the ComPlan planning support approach.

## Knowledge representation

Eliciting and encoding expert knowledge is a crucial and often limiting step in the creation of knowledge-based decision support. Our DKExpert system demonstrated a way for end users to create rules of their own so as to reduce problems with traceability and coverage In Section **??** we presented some of the approaches used to extract knowledge on military mission planning. In these projects, knowledge is encoded using formal ontology languages like *Loom*, used by the EXPECT tool [150], or Hierarchical Task Networks (HTN) used by HICAP [112]. In these cases, the ontology language used is well suited for expressing relationships like inheritance, part-whole relations, sequencing of actions and, to some limited extent, numerical constraint properties such as that fuel levels, which cannot be negative. When we first considered how to implement domain knowledge, we saw these types of description-logic-based languages as the obvious choice. They offer formally sound structures, good expressiveness and are widely used in the knowledge-based systems community. However, we also realized that this approach has some clear limitations. Ontology languages imply *declarative* descriptions, but offer little when it comes to *computational* or *procedural* descriptions. This is similar to the relationship between *formal descriptions* of computer programs (such as object diagrams in UML[1]) and the actual *implementation* of the program. In its implementation, the program contains not only an object structure but also an *algorithmic description* of how the program behaves. We found that when modeling critiquing support which not only analyzes task *structures* but also task *behaviors* over time, a pure declarative formalism for describing a task domain is rather restricting. For example:

1. When describing task types, you cannot integrate a description of how task types should be manipulated in the views of the planning tool. When describing a *transportation task* we could, for example, use the following parameters:

    - name,
    - description,
    - set of agents involved in the activity,
    - preferred start time,
    - preferred end time,
    - starting location, and
    - destination

    Start and end locations may be best viewed and modified through a map-like view presenting real geographical data, whereas time information may be best handled through a time-line view. Also, each type of parameter may need a special routine for specifying how it is presented and may be modified. A location may be presented as an outlined area in the geographical view, the set of agents assigned to a task may be represented by drawing lines from a set of graphical representations of selectable agents and tasks, or using any other method.

2. When relationships such as inheritance or sequential ordering are introduced between tasks, parameters in the related tasks can be affected non-deterministically, which makes it hard to declaratively state the exact effects

[1] Unified Modeling Language

of performing a task. For example, if a commanders decides that he needs to transport a number of units to a location (task A) at which he subsequently needs to establish an emergency shelter for refugees (task B), this course of action ought to introduce a relationship between the destination of task A and the location of task B. How should this property be defined using a standard ontology language? In propositional logics, which forms the underpinning of both HTNs and Loom, we could probably *unify* the values of the two locations to solve this. But what if we only want to introduce a relationship such as that the end time task A should *not be preceded* by the start time of task B? Then again, maybe task B can start before all units involved in task A have arrived, so maybe we only want to *visualize* this constraint, not *enforce* it.

3. When simulating the effects of actions, every action may be unique in its definition of `effect'. Also, the plan can be laid out manually by a human operator, which allows for much freedom regarding *which* and *how much* information is available to the simulation engine. For example, the task of transporting 100 people from one location to another may be planned with only one bus and this bus is only able to carry 50 people. Does the human planner with this information imply that the bus should be used beyond its capacity, should the vehicle be used several times, should we just leave some people behind or is it a mistake on his side? In an ontological definition of a transportation task, the procedural knowledge needed for simulating it would simply be missing. To add it elsewhere[2] introduces unnecessary complexity since a concept (in this case, the transportation task) is encoded using two different representations.

No doubt, there are benefits of using ontology languages for modeling purposes, especially when using special ontology tools such as Prot□g□ [62], where domain experts can describe their domain through a customized interface specially made for every domain. In heterogeneous knowledge-intensive environments, when there are problems establishing common definitions of a large set of concepts, ontological approaches to knowledge representation offer great promises.

However, the ComPlan is not first and foremost thought as an actor in such an environment. The examples above illustrate some of the problems we see when using ontological descriptions of tasks and resources for our specific type of planning application. Due to these problems, *we have decided not to incorporate a special-purpose knowledge representation language as part of our ComPlan application*. Hierarchical descriptions of activities and resources can be represented just as well using object-oriented programming languages, along with procedural definitions of activities. One may even generate stubs for these hierarchical structures using graphical modeling tools such as Argo/UML (see Section **??**).

[2]in programming language code, and as a part of the planning application

There is also another issue with the use of formal ontology languages for specifying domain knowledge: They require special expertise. Although there has been much research on how to automate or streamline the knowledge elicitation process, there are still requirements for special knowledge engineering competence at some level in the process of describing domain concepts. This competence may be harder to find than programming competence, and must therefore be justified by some considerable and measurable advantages. Seeing as both types of competence are required to develop domain models for a decision support tool which uses formal ontology language[3], there may be an issue to justify the extra resources needed to implement the domain model in Loom or a Hierarchical Task Network compared to modeling it directly in a programming language.

## Views

A core design feature of ComPlan is the use of different interconnected views of a planning situation. Each view is used to illustrate and manipulate one aspect of the plan, such as resource organization and allocation, task structure, scheduling or geographical aspects. These are the particular views we have implemented, but they are by no means an exclusive or exhaustive list of the aspects relevant to mission planning. They do, however, represent a sufficiently diverse visualization and interaction mechanisms to illustrate how views could assist a human planner develop better understanding of a situation.

In our design of the ComPlan, we have considered various interaction modes of critiquing systems and have studied what these modes imply for users. In Section **??** we classified decision support systems based on *initiative*. This is not a measure of *how responsive* an application is, but mostly it describes what a user is *allowed to do*. It determines whether a user is allowed to ignore restrictions implemented in the knowledge base of the support tool, whether he may choose the order for filling in information and which feedback he wants.

Although our aim is for the user to work as unrestricted by the knowledge available in the tool as possible, it is apparent that the tool will not be very useful unless certain information is provided. If planning for an emergency evacuation of hundreds of people, the time frame for doing this is important. However, before a user can manipulate timing information in the `time view', at least one task must be created in the `task view'. We could of course allow for tasks to be created in any view using shortcut commands or menu options as in any application, but the point of using views is really that there are separate concerns in a plan that one should be able to treat separately. In future research, we will have to investigate closer what views should really encapsulate: Perhaps they should only *present different aspects* of a plan but allow for direct manipulation of *all aspects in all views*?

In figure 1 we illustrate how the views in ComPlan are dependent on each other. For the time view to be at all useful, one must have determined which tasks

[3]as there may be procedural knowledge which needs to be represented in a programming language

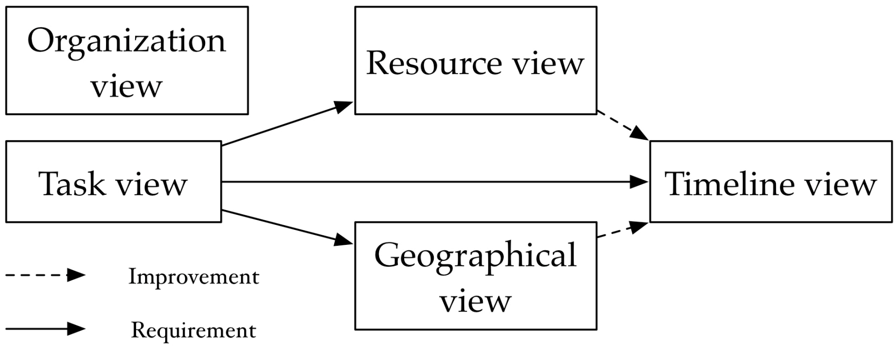


Figure 1: Dependencies between the different views. For example, the time view only becomes useful when information is supplied in the task view on which tasks we are about to perform. As more information becomes available, the time view is able to provide more feedback such as simulation-based feedback when both agents, locations and other parameters are set.

are part of the mission so that there are any missions to manipulate. However, without having set parameters for resources and geography, the time view can provide little feedback on which constraints are being violated and what problems may arise from the current mission plan. The dependencies are mostly related to increased feedback options, and do not restrict the user from manipulating a plan in just about any order. For example, the organizational view is used to navigate and determine the properties of available forces. It can be manipulated in parallel with any of the other views so that one can try the effects of assuming that a rescue helicopter can transport more patients than stated by default, or that the range or speed of some other resources differs from the usual.

The task view (Figure 2) shows a graph of the group of tasks planned for a particular mission. This group of tasks together forms a mission to be carried out by a subordinate commander.

As more information is available, more calculations and simulations can be performed in the time view to help the mission planner see potential problems. Hence, the time view will receive improved functionality as more information is provided in the other views. In Figure 4 we can see how the user has initiated a simulation and receives feedback that one of the allocated rescue helicopters may run out of fuel in mid-flight. Temporal restrictions introduced by the relationships introduced in the task view are here used with an *automatic update* policy which forces temporal restrictions to be in effect at all times during the mission. Therefore, all tasks are re-ordered automatically when a user manipulates one of them.

## Constraints

In automated planning, all planners use strict formalisms for describing planning domains, permitted operations, and resulting effects of operations. Thus, `con-

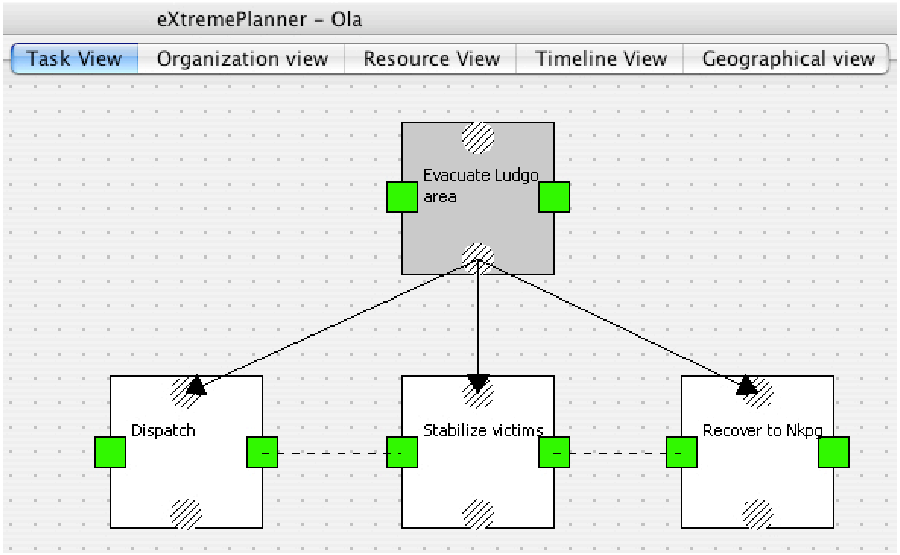


Figure 2: The task view showing a number of tasks ordered hierarchically as well as sequentially.

straints' can best be defined as the prerequisites necessary for atomic plan operations. If an automated planner is used as part of a mixed-initiative system for mission planning, the constraints put on a user are likely to be mandatory, that is, not to be violated by a user during planning. Such hard restrictions may be troublesome to maintain if the world model of the planning tool is not sufficient. Another view on how to use constraints comes from the Cognitive Systems Engineering field of research, where researchers have studied how human operators with time-critical real-time monitoring tasks, such as air traffic controllers (ATCs), work and what role constraints play in that context [45]. Thus, efficiently interpreting an information flow describing ongoing events is critical. For ATCs, relevant constraints relate, for instance, to runway and air space capacities. Any support system for such an application area needs to make absolutely sure that all relevant information is presented in as efficient a manner as possible. Two airplanes that occupy the same altitude segment may not necessarily collide, but information that they are almost at the same altitude (and about to cross each others paths) may be more important to present than if they are several miles apart. Thus, `constraints' in real time control applications need to be properly *presented*, but controllers are not always in a position to *preemptively* ensure they are not violated. In their 1999 article on ATC support systems that provide `management by exception'[4] Dekker and Woods note the following [45]:

[4]Management by exception means that the computer system is in control of the management situation but can relinquish control in case of exceptional circumstances. However, the system may be interrupted at any time by a human operator.

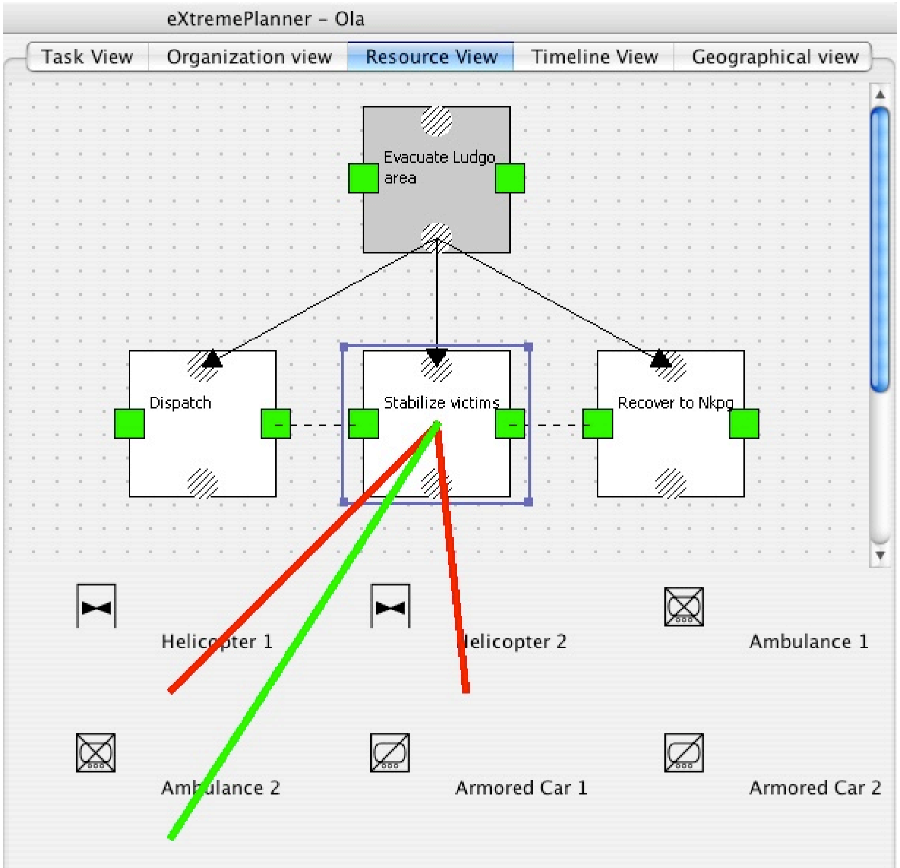


Figure 3: The resource view showing a number of tasks together with resources available. Resources associated with tasks are denoted by colored lines.

> 94 Management by exception traps human controllers in a dilemma: intervening early provides little justification for restrictions (and compromises larger air traffic system goals). But intervening late leaves little time for actually resolving the problem, which by then will be well under way (thereby compromising larger air traffic system goals).

So, many constraints in automated or mixed-initiative planning systems must be *respected* by the human operator, but constraints in real-time control systems need should first and foremost be *well presented*. As we see it, in crisis management situations there may be need to use constraints for both purposes. Some aspects of a plan may be more amenable to the policy of automatic enforcement used by project management tools and some mixed-initiative planning tools. Other aspects may resemble the uncertain constraints in the ATC scenario more, where visualization as support is favored compared to thorough computer-based analysis of possible errors. In our ComPlan design, we offer users the option of selecting be-

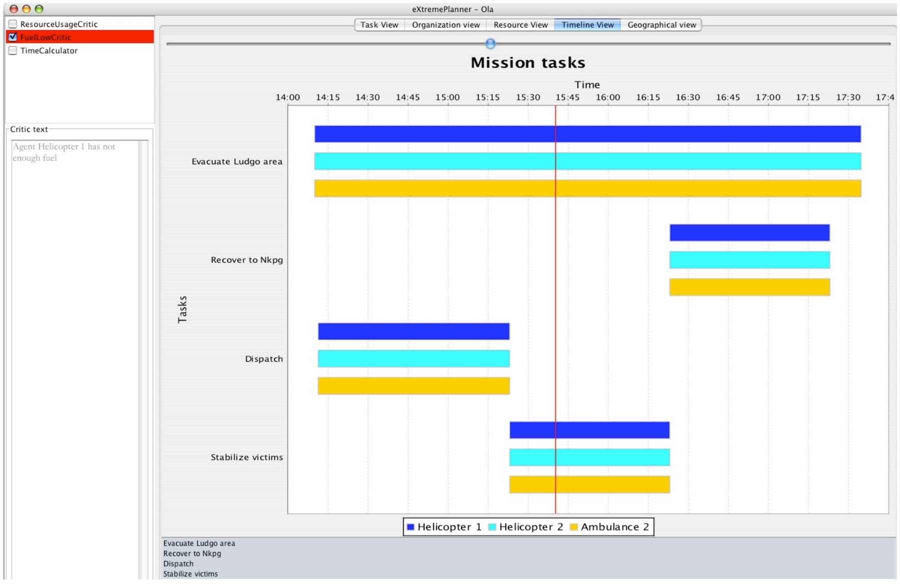


Figure 4: The timeline view showing a set of tasks as planned over time.

tween mandatory enforcement and visualization, by using constraint *policies* that can change the behavior of the constraint checking engine.

### Automatic enforcement

Although we are dealing with rather unpredictable real-life scenarios in mission planning, we believe that some constraints may still be considered `hard' constraints. If a mission planner has stated that task `transport patients to rendez-vous location A' should precede another task which takes place *at* rendez-vous location A, he may want this ordering property to hold even if he makes changes to when either task should begin or how much time they should take. By automatically monitoring and enforcing constraints, the planner can rely on certain properties of his plan. On the other hand, such a policy may not be suitable if it is later discovered that these restrictions are unnecessary or misleading.

### Visualization

Instead of automatically enforcing constraints on behalf of the user, the system may instead only visualize constraint information and possibly present critique based on this information. As Dekker and Woods argued in their 2002 article *MABA-MABA or Abracadabra? Progress on Human-Automation Co-ordination* [44], human operators may be better served by graphical representations of relationships between critical variables compared to being served computer interpretations. Humans are good at interpreting graphical patterns and if provided with relevant and well-presented

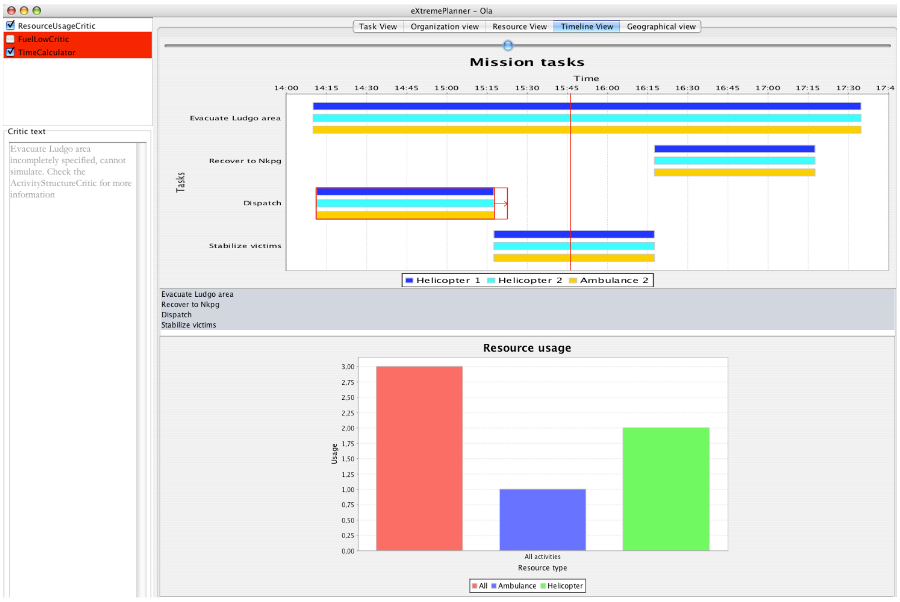


Figure 5: The timeline view, showing resource usage over time as part of a user-driven simulation. By pulling a slider, the user can simulate various simple aspects of a plan to better appreciate the effects of resource allocation for example.

information on tangible constraints, they may be better at interpreting such representations than most AI algorithms. Critique and visualization are still closely related though, because a critiquing engine can make a user aware of potential problems, but illustrate these problems using rich representations which make the reasoning of the critiquing engine more transparent. For example, the critiquing engine may assume that there is a problem with the relation between the fuel range of a vehicle and its current location. Following this, it may choose to inform the user of this problem by graphically representing the fuel range and distance to refueling facilities and *not* just tell the user that `the vehicle may run out of gas'. This representation has also been considered by others for illustrating constraints [44, 164, 166]. Currently, information on constraints can be represented in graphs and charts in the interactive simulation available through the time view (see Figure 5 for an example of charts used to aggregate information in the time view).

## Interactive simulation

Simulation can be a powerful mechanism for illustrating connections and constraints that are not visible in static information displays. Also, simulations are used by many decision-support tools to predict likely outcomes of different courses of action. Predictions are potentially troublesome, however, for the same reason as critiquing and option evaluation may be hazardous: the world model may not be accurate enough, and some parts of reality may just be very difficult to simulate. However, simulation offers great prospects for efficiently presenting time-related information. For this purpose, we have included the option of simulating events in the time view. Based on specific information such as fuel consumption estimation, the simulation engine simulates the behavior of each *resource* in the plan over time, as it is part of executing one or several tasks in parallel.

This kind of simulation can be useful not only for visualizing the development of simple resource parameters, such as the expected fuel consumption in units over time, but also for illustrating constraints and informing the user of potential problems. With `problems', we mean not only constraint *violations* but rather to illustrate how *fragile* a plan is with respect to various constraints over time. That is, how much freedom will this set of resources give me at various points in time further on? Since information may be scarce at the beginning, it is important that initial planning allows for much adaptation to new circumstances and information. Therefore, *sensitivity analysis* of the constraint model is an important instrument for planners. Currently, this is performed by simply illustrating raw simulation data on available resources at each point in time or critiquing messages whenever constraints are violated, but there are useful variations to information presentation which may be used in between those two alternatives.

# 5 Semantic Desktops

Document processing, part 1: I-KNOW -08

# 6 Communication Analysis

## Method

Three phases in analysis, interviews, prototyping and workshop evaluation

The features that a classifier can recognize and expose to the user as part of the model constructed depend on how the classifier constructs its model. A classifier typically builds an internal model by receiving a set of instances (messages) with a limited, fixed number of attributes and a single classification attribute [163]. A rule- or tree-based classifier may reveal combinations of attribute values or ranges of values that correspond to decision classes in a way that a human could understand and interpret. However, some features of a message flow may be inaccessible to such a classifier, if the features concern the relationship *between messages* (For example the timing between messages, the relation between the last recipient of a message and the current sender) or the relationships *between attributes* of a single

| **Indentification of** | **through** | | |
|---|---|---|---|
| | Interviews | Prototype | Workshop |
| Goals in analysis | • | | |
| Challenges | • | | • |
| Tool options | | • | • |
| Criteria | | • | • |

Table 1: An overview of how the three phases of this study have contributed to the conclusions in this paper

message (For example the relation between the roles of the sender and the recipient). Also, classifying with respect to text tends to be considered a problem in itself, requiring methods for classification other than those dealing with data types for which there exists a natural, total ordering. Texts can be analyzed with respect to string distances, keyword occurrences, grammatical structures et cetera, but with no universal, domain-independent, language-independent definition of similarity. This in turn means that the combination of message text and other discriminating attributes of messages is difficult to combine for classification purposes without explicitly describing how text features can be combined with other attributes, which would have to be domain-specific. In our analysis of the precision of a combined classifier, we concluded that the text-based classifier was much more important to classifying messages than the non-text-based one, but none of the classifiers had access to *both* text and other attributes. This means that patterns that emerge when submitting an order (the combination of certain phrases in the text, submitted from an individual who operates at a certain level in the organizational hierarchy) can elude the classifier. It would be interesting to investigate whether the representation of messages can be adapted to reveal the relations exemplified above to the classifier at a low cost in terms of computational and manual labor.

# 7 Discussion

> Plans are worthless, but planning is everything -- Dwight D. Eisenhower

## Requirements for planning support systems

### Assumptions in automated planning

To create a plan as a sequence of actions automatically through a planning system, there are some specific requirements:

There must be a set of operators which can be assembled in a sequence that forms a plan. These operators must have well-defined effects that can be interpreted as the preconditions that are necessary for other actions. There must also be an metric that can be applied to a plan to determine the utility of one sequence of actions compared to another.

### Human functions to support

The group of commanders involved in planning an operation are supposed to determine which they should use to achieve the given political goal and which physical resources they are supposed to employ to achieve the desired effects. Also, they need to understand how much they do and do not know about a situation, the basis for making assumptions about constraints, resources and hostile intentions and how to manage uncertainties during an operation.

The plan document in itself may be a fragile (by-)product of the process of understanding the constraints involved in planning

## Requirements for semantic desktop systems

### Assumptions in semantic desktop systems

To create a semantic desktop system capable of making inferences on objects in the application domain, one has to produce a formally sound ontological description of a domain in the language of the desktop ontology.

The OWL language of the IRIS semantic desktop system requires objects to belong to a specific class, and that classes are specified using necessary and sufficient conditions for membership.

The simplest form of ontological description of objects consists of classes with no required properties. Without object properties, an inference engine cannot classify objects automatically, but the ontology could still be used for filing purposes as folders in filesystems. With more elaborate class descriptions, where objects must have attribute values that make their classification unique, inference engines can determine the relationships between objects and classes, and between classes and other classes, to support the organization of knowledge.

Harvesting objects from the products (e.g. documents, e-mail or browser data) or interactions (e.g. sending e-mail, opening documents, clicking on links) in a semantic desktop requires that all domain-specific semantic objects are available in machine-readable format. Interactions with a desktop environment are defined by the environment, and objects in the form of files are naturally opened, saved and sent through the semantic desktop system which means that they are accessible for inspection at a low level. However, to perform reasoning tasks related to the domain of work relevant to a user, there must be a mapping between objects relevant to a desktop system and objects relevant to a user. In Chapter 5 we describe how concepts and objects that were relevant to reason about in a $C^2$ context could be found in a documents. In general, to capture the semantics behind user-produced interactions or documents would require either:

1. a restricted, domain-specific workflow management system for communication or document production system for documents that directs users to use only expressions that are machine-readable, or
2. an annotation mechanism for everything that is to be machine-translated (documents, messages), embedded in the desktop environment
3. well-structured documents or messages that follow typographical, machine-comprehensible standards which relate semantic objects to the structure of a document or well-defined patterns within documents or messages.

In Chapter 5 we describe a system for managing documents that had not been authored using dedicated authoring tools, and with no annotation procedures. Due to the types of documents and the characteristics of the semantic objects contained in these documents, we were able to extract objects of relevance for the planning process and demonstrate how they can be manipulated in representations that are dedicated to each type of object (*unit synchronization orders* and *military locations*).

### Human functions to support

When planning, human planners collaborate to develop courses of action by creating and exchanging a set of plan documents. In these documents, there are cross-references to other documents regarding units, locations, targets, timing and background information. Some information may be duplicated in several documents to make the documents easier to read, and parts of each document is based on information in other parts. As several actors independently make changes to their respective documents, misunderstandings and errors are almost inevitable over the course of a few days. One planner may make assumptions regarding the status of a synchronization matrix (that it is nearly finalized) and decide to work on a plan to implement the tasks assigned to his or her units. A few hours later, changes are made to the synchronization matrix that are not propagated properly and lead to confusions later on. Alternatively, a secure point of debarkation may be marked as only tentative and not definitely decided, but misunderstood by naval officers who start planning for debarkation at the suggested location only to discover that the location is changed later on.

One compounding factor causing the problems in the planning scenario we have described is the use of office documents for structuring information. However, in an open environment with high demands for interoperability and explicit demands on using off-the-shelf products that are likely to continue to be maintained and upgraded, standard tools for document preparation are likely to prevail.

## Requirements for machine learning systems

### Assumptions in machine learning systems

Machine learning works in either supervised or unsupervised mode: as a classification of instances in a fix number of classes given known classifications, or as clustering of data in groups given a proximity metric to apply. In supervised mode, a classifier can produce a model of a set of observations which is possibly a condensed, generalized representation of the patterns in data which correspond to classifications made. There are many heuristics to apply regarding exactly how to generalize from the given training examples and represent the condensed model, as propositional rules, decision trees, decision networks, decision tables et cetera. All these models make assumptions on how data sets can be generalized, and fit different domains depending on these assumptions.

The patterns that can be detected by machine learning systems have to do with relationships among attribute values of single instances in a dataset. Most machine learning approaches use only a fix number of attributes and do not consider patterns that involve sequences of observations. Patterns that can be found as combinations of patterns in text and other attributes are usually not considered. For a pattern to be detected, there has to be either a set of examples of classified instances, with examples of all the possible classes that may be found, which the supervised learning algorithm uses to create a classification model. Alternatively, for unsupervised learning, there has to be a well-defined distance metric for comparing instances with one another so that clustering instances can be performed for all instances.

The performance of a machine learning system is usually measured as a function of the precision attainable with a system. In some instance, the precision itself may be enough to determine the utility of a classifier. For filtering junk e-mail, however, precision requirements are not symmetric, as the cost of classifying a message as junk in error is much higher than not classifying a junk message as such.

### Human functions to support

In the third case study presented, we describe the work of exploring relationships in data from $C^2$ scenarios. The analysis of how researchers explore data sets of communications in command and control is a meta-analysis of the work in command and control. In the workflow presented in paper 7, the researchers describe how they selected research material for closer study and drew conclusions from material they had processed. Their descriptions of an iterative, explorative process, in which selection of data sets and visualization of different kinds of patterns was central to their work.

# Bibliography


[1] Mark S. Ackerman. The intellectual challenge of CSCW: The gap between social requirements and technical feasibility. *Human-Computer Interaction*, 15:179--203, 2000.

[2] D. S. Alberts, J. J. Gartska, and F. P. Stein. *Network Centric Warefare: Developing and Leveraging Information Superiority*. National Defense University Press, Washington, DC, 2000.

[3] D.S. Alberts and R.E. Hayes. *Power to the Edge: Command, Control in the Information Age*. CCRP Publication Series, 2003.

[4] Pär-Anders Albinsson and Johan Fransson. Representing military units using nested convex hulls -- coping with complexity in command and control. In *Proceedings of the First Swedish-American Workshop on Modeling and Simulation*, Orlando, USA, October 30--31 2002.

[5] James Allen and George Ferguson. Human-machine collaborative planning. In *Proceedings of the Third International NASA Workshop on Planning and Scheduling for Space*, Houston, Texas, 2002. URL http://www.cs.rochester.edu/~ferguson/papers/.

[6] David Anderson, Emily Anderson, Neal Lesh, Joe Marks, Brian Mirtich, David Ratajczak, and Kathy Ryall. Human-guided simple search. In *Proceedings of the Seventeenth National Conference on Artificial Intelligence*, Austin, Texas, 2000. URL http://www.dominik.net/research/hugss/.

[7] Stephen J. Andriole. *Handbook of Decision Support Systems*. TAB Books Inc., 1989.

[8] Michael Argyle. *The Social Psychology of Work*. The Penguin Press, London, UK, 1972.

[9] Stefan Arnborg, Joel Brynielsson, Henrik Artman, and Klas Wallenius. Information awareness in command and control: Precision, quality, utility. In *Proceedings of the International Conference on Information Fusion*, 2000.

[10] Anders Arpteg. *Intelligent Semi-Structured Information Extraction: A User-Driver Approach to Information Extraction*. PhD thesis, Department of Computer and Information Science, Linköping university, Linköping, Sweden, 2005.

[11] John Arquila and David Ronfeldt. Swarming: The future of conflict. Technical report, RAND, National Defense Research Institute, 2000.

[12] Paolo Avesani, Anna Pereni, and Francesco Ricci. Interactive case-based planning for forest fire management. *Applied Intelligence*, 13:41--57, 2000.

[13] Franz Baader, Diego Calvanese, Deborah McGuiness, Daniele Nardi, and Peter Patel-Schneider, editors. *The Description Logic Handbook*. Cambridge University Press, 2003.

[14] Olle Bälter and Candace L Sidner. Bifrost inbox organizer: giving users control over the inbox. In *NordiCHI '02: Proceedings of the second Nordic conference on Human-computer interaction*, pages 111--118, New York, NY, USA, 2002. ACM Press. ISBN 1-58113-616-1. doi: http://doi.acm.org/10.1145/572020.572034. URL http://doi.acm.org/10.1145/572020.572034.

[15] John A. Bateman. On the relationship between ontology construction and natural language: a socio-semiotic view. *International Journal of Human--Computer Studies*, 43(5/6):929--944, 1995.

[16] Allen W. Batschelet. Effects-based operations: A new operational model? Technical report, US Army War College, 2002.

[17] Marcel A. Becker and Stephen F. Smith. Mixed-initiative resource management: The AMC Barrel Allocator. In *Proceedings of the 5th International Conference on Artificial Intelligence Planning and Scheduling (AIPS-2000)*, Breckenridge, Colorado, USA, 2000. URL http://www.cs.cmu.edu/~ozone/barrel/barrel.html.

[18] Micheline Bélanger and Adel Guitouni. A decision support system for coa selection. In *Proceedings of the 5th International Command and Control Research and Technology Symposium*, 2000.

[19] Marie Bienkowski. Demonstrating the operational feasibility of new technologies. *IEEE Expert*, 10(1):27--33, February 1995.

[20] Jim Blythe and Yolanda Gil. A problem-solving method for plan evaluation and critiquing. In *proceedings of the International Knowledge Acquisition Workshop*, 1999. URL http://www.isi.edu/~blythe/papers/pdf/psm.pdf.

[21] A. Boukhtouta, A. Bedrouni, J. Berger, F. Bouak, and A. Guitouni. A survey of military planning systems. In *Proceedings of the 9th International Command and Control Research and Technology Symposium (ICCRTS)*, 2004.

[22] John Boyd. A discourse on winning and losing. Maxwell Air Force Base, AL: Air University Library Document No. M-U 43947 (Briefing slides), 1987.

[23] Berndt Brehmer. Dynamic decision making: Human control of complex systems. *Acta Psychologica*, 81(3):211--241, December 1992.

[24] Berndt Brehmer. The Dynamic OODA Loop: Amalgamating Boyd's OODA Loop and the Cybernetic Approach to Command and Control. In *Proceedings of the 2005 Command and Control Research and Technology Symposium*, 2005.

[25] John L. Bresina, Ari K. Jónsson, Paul H. Morris, and Kanna Rajan. Mixed-initiative planning in MAPGEN: Capabilities and shortcomings. In *Proceedings of the IJCAI-2005 Workshop in Mixed-Initiative Planning and Scheduling*, Edinburgh, Scotland, 2005. URL http://www.cs.rochester.edu/research/mipas2005/final-drafts/bresina-jonsson-morris-rajan.pdf.

[26] Leonard A. Breslow and David W. Aha. NaCoDAE: Navy conversational decision aids environment. Technical Report AIC-97-018, Navy Center for Applied Research in Artificial Intelligence (NCARAI), Washington, DC, USA, 1998. URL http://citeseer.ist.psu.edu/breslow98nacodae.html.

[27] R. Brown. *Group Processes - Dynamics Within and Between Groups*. Blackwell, Cambridge, Massachusetts, 1993.

[28] Joel Brynielsson and Stefan Arnborg. Bayesian games for threat prediction and situation analysis. In *Seventh International Conference on Information Fusion*, 2004.

[29] Mark H. Burstein and Drew V. McDermott. Issues in the development of human-computer mixed-initiative planning systems. In B. Gorayska and J.L.Mey, editors, *Cognitive Technology: In Search of a Humane Interface*, pages 285--303. Elsevier Science B.V., 1996.

[30] Janis A. Cannon-Bowers, Eduardo Salas, and Sharolyn Converse. Shared mental models in expert team decision making. In N. John Castellan, editor, *Individual and group decision making: current issues*, pages 221--246. Lawrence Erlbaum Associates, Hillsdale, N.J., 1993.

[31] Rafael Capurro. Hermeneutics and the phenomenon of information. In *Metaphysics, Epistemology and Technology. Research in Philosophy and Technology.*, volume 19, pages 79--85. Elsevier Science Publishers Ltd., 2000.

[32] Arthur K. Cebrowski and John J. Garstka. Network-centric warfare: Its origin and future. *U.S.Naval Institute Proceedings*, 124(1):28--35, 1998.

[33] Anup Chalamalla, Sumit Negi, L. Venkata Subramaniam, and Ganesh Ramakrishnan. Identification of class specific discourse patterns. In *CIKM '08: Proceeding of the 17th ACM conference on Information and knowledge management*, pages 1193--1202, New York, NY, USA, 2008. ACM. ISBN 978-1-59593-991-3. doi: http://doi.acm.org/10.1145/1458082.1458240.

[34] Alan Chalmers. *What Is This Thing Called Science?* Queensland University Press, third edition, 1999.

[35] Adam Cheyer, Jack Park, and Richard Giuli. IRIS: Integrate. relate. infer. share. In *Proceedings of the Semantic Desktop and Social Semantic Collaboration Workshop (SemDesk-2006)*, Athens, GA, USA, November 6 2006.

[36] Aaron V. Cicourel. The integration of distributed knowlede in collaborative medical diagnosis. In J. Galegher, R.E. Kraut, and C. Egido, editors, *Intellectual Teamwork: Social and Technological Foundations of Cooperative Work*. Lawrence Erlbaum Associates, Hillsdale, New Jersey, 1990. http://www.erlbaum.com/60.htm.

[37] Delwyn N. Clark. A literature analysis of the use of science management tools in strategic planning. *The journal of the operational research society*, 43(9): 853--870, 1992.

[38] Clausewitz. *On War*. Wordsworth Editions, Hertfordshire, England, 1997.

[39] Robin Cohen, Coralee Allaby, Christian Cumbaa, Mark Fitzgerald, Kinson Ho, Bowen Hui, Celine Latulipe, Fletcher Lu, Nancy Moussa, David Pooley, Alex Qian, and Saheem Siddiqi. What is initiative? *User Modeling and User-Adapted Interaction*, 8:171--214, 1998.

[40] G. Cortellessa, A. Cesta, A. Oddi, and N. Policella. User interaction with an automated solver - the case of a mission planner. *PsychNology*, 2(1):140--162, 2004. URL http://www.istc.cnr.it/showabstract.php?bibid=84.

[41] Mary L. Cummings. *Designing Decision Support Systems for Revolutionary Command and Control Domains*. PhD thesis, School of Engineering and Applied Science, University of Virginia, 2004.

[42] Douglas R. Cutting, David R. Karger, Jan O. Pedersen, and John W. Tukey. Scatter/gather: A cluster-based approach to browsing large document collections. In *Proceedings of the 15th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1992.

[43] Antoine Henri (Baron) de Jomini. *The Art of War*. Greenhill, 1996.

[44] S. W. A. Dekker and D. D. Woods. MABA-MABA or abracadabra? progress on human-automation co-ordination. *Cognition, Technology & Work*, 4(4):240--244, 2002. URL http://www.springerlink.com/content/q2legq6x4cvw4wtw/.

[45] Sidney W. A. Dekker and David D. Woods. To intervene or not to intervene: The dilemma of management by exception. *Cognition, Technology & Work*, 1(2):86--96, September 1999. URL http://www.springerlink.com/content/ttug9eukr261ckb4/.

[46] Xin (Luna) Dong and Alon Halevy. A platform for personal information management and integration. In *Proceedings of the Second Biennial Conference on Innovative Data Systems Research*, 2005.

[47] Hubert Dreyfus. Why heideggerian ai failed and how fixing it would require making it more heideggerian. Memo on the state of AI, 2006.

[48] Magdalini Eirinaki and Michalis Vazirgiannis. Web mining for web personalization. *ACM Transactions on Internet Technology*, 3(1):1--27, 2003. doi: http://doi.acm.org/10.1145/643477.643478.

[49] Mica R. Endsley. Toward a theory of situation awareness in dynamic systems. *Human Factors*, 37(1):32--64, 1995.

[50] Mica R. Endsley. *Situation Awareness Analysis and Measurement*, chapter Theoretical Underpinnings Of Situation Awareness: A Critical Review. Lawrence Erlbaum Associates, 2000.

[51] Max D. Engelhart, Edward J. Furst, Walker H. Hill, and David R. Krathwohl. *Taxonomy of Educational Objectives: Book 1 Cognitive Domain*. Longman, 1956.

[52] Kutluhan Erol, James Hendler, and Dana S. Nau. UMCP: A sound and complete procedure for hierarchical task-network planning. In *Proceedings of the 2nd International Conference on AI Planning Systems (ICAPS)*, pages 249--254, 1994.

[53] Juan Fdez-Olivares, Luis Castillo, Óscar García-Pérez, and Francisco Palao. Bringing users and planning technology together: Experiences in SIADEX. In *Proceedings of the Sixteenth International Conference on Automated Planning and Scheduling (ICAPS 2006)*, Cumbria, UK, 2006. URL https://siadex.ugr.es/public/docs.htm.

[54] George Ferguson and J. Allen. Mixed-initiative dialogue systems for collaborative problem solving. In *Proceedings of the AAAI Fall Symposium on Mixed-Initiative Problem Solving Assistants (FS-05-07)*, pages 57--62, Arlington, Virginia, 2005. URL http://www.cs.rochester.edu/~ferguson/papers/.

[55] George Ferguson, James Allen, and Brad Miller. TRAINS-95: Towards a mixed-initiative planning assistant. In *Proceedings of the Third Conference on Artificial Intelligence Planning Systems (AIPS-96)*, pages 70--77, Edinburgh, Scotland, 1996. URL http://www.cs.rochester.edu/~ferguson/papers/.

[56] Frederico T. Fonseca and James E. Martin. Toward an alternative notion of information systems ontologies: Information engineering as a hermeneutic enterprise. *Journal of the American Society for Information Science and Technology*, 56(1):46--57, 2005.

[57] Leslie Zimmerman Foor and Drew J. Asson. Spike: A dynamic interactive component in a human-computer long-range planning system. In *3rd International NASA Workshop on Planning and Scheduling for Space*, Houston, Texas, 2001. URL http://www.stsci.edu/resources/software_hardware/spike/.

[58] K. Forbus, J. Usher, and V. Chapman. Sketching for military courses of action diagrams. In *Proceedings of Intelligent User Interfaces Conference*, Miami, Florida, 2003.

[59] Johan Fransson and Pär-Anders Albinsson. Communication visualization - an aid to military command and control evaluation. In *Human Factors and Ergonomics Society Annual Meeting Proceedings*, pages 590--594, 2001.

[60] Jolene Galegher. Technology for intellectual teamwork: Perspectives on research and design. In J. Galegher, R.E. Kraut, and C. Egido, editors, *Intellectual Teamwork: Social and Technological Foundations of Cooperative Work*, pages 1--20. Lawrence Erlbaum Associates, Hillsdale, New Jersey, 1990. http://www.erlbaum.com/60.htm.

[61] Liqiang Geng, Scott Buffett, Bruce Hamilton, Xin Wang, Larry Korba, Hongyu Liu, and Yunli Wang. Discovering structured event logs from unstructured audit trails for workflow mining. In Jan Rauch, Zbigniew Ras, Petr Berka, and Tapio Elomaa, editors, *Foundations of Intelligent Systems*, volume 5722 of *Lecture Notes in Computer Science*, pages 442--452. Springer Berlin / Heidelberg, 2009. URL http://dx.doi.org/10.1007/978-3-642-04125-9_47.

[62] John H. Gennari, Mark A. Musen, Ray W. Ferguson, Wiliam E. Grosso, Monica Crubézy, Henrik Eriksson, Natalya F. Noy, and Samson W. Tu. The evolution of Protégé: an environment for knowledge-based systems development. *International Journal of Human-Computer Studies*, 58: 89--123, 2002. URL http://smi-web.stanford.edu/auslese/smi-web/reports/SMI-2002-0943.pdf.

[63] Melinda T. Gervasio, Wayne Iba, and Pat Langley. Case-based seeding for an interactive crisis response assistant. In *Case-Based Reasoning Integrations:*

*Papers from the 1998 Worksop. AAAI Technical report WS-98-15*, pages 61--66. AAAI Press, 1998.

[64] Yolanda Gil. Plan representation and reasoning with description logics. Course literature in CS541 - Artificial Intelligence Planning, 2004. URL http://www.isi.edu/~blythe/cs541/.

[65] Yolanda Gil and Jim Blythe. PLANET: A shareable and reusable ontology for representing plans. In *AAAI 2000 workshop on representational issues for real-world planning systems*, Austin, Texas, 2000. URL http://www.isi.edu/isd/HPKB/planet/planet-ws00.pdf.

[66] Yolanda Gil et al. A knowledge acquisition tool for course of action analysis. In *Proceedings of the Fifteenth Annual Conference on Innovative Applications of Artificial Intelligence*, pages 43--50, Acapulco, Mexico, August 12--14 2003. URL http://www.qrg.cs.northwestern.edu/papers/Files/975.pdf.

[67] Daniel R. Gonzalez. *Evolution of CTAPS and the Air Campaign Planning Process*. RAND, 1996.

[68] Dik Gregory. Delimiting expert systems. *IEEE Transactions on Systems, Man & Cybernetics*, 16(6):834--843, November/December 1986.

[69] Larry Ground, Alexander Kott, and Ray Budd. A knowledge-based tool for planning of military operations: the coalition perspective. In *Proceedings of the Second International Conference on Knowledge Systems for Coalition Operations*, 2002.

[70] Sture Hägglund. Introducing expert critiquing systems. *The knowledge engineering review*, 8(4):281--284, 1993.

[71] Sajjad Haider and Alexander H. Levis. On finding effective courses of action in a complex situation using evolutionary algorithms. In *Proceedings of the 10th International Command and Control Research and Technology Symposium*, 2005.

[72] Stanley M. Halpin. The human dimensions of battle command: A behavioral perspective on the art of battle command. Research Report 1696, U.S. Army Research Institute for the Behavioral and Social Sciences, 1996.

[73] C. C. Hayes, A. D. Larson, and U. Ravinder. Weasel: A mixed-initiative system to assist in military planning. In *Proceedings of the IJCAI-2005 Workshop in Mixed-Initiative Planning and Scheduling*, Edinburgh, Scotland, 2005. URL http://www.cs.rochester.edu/research/mipas2005/final-drafts/hayes-larson-ravinder.pdf.

[74] Martin Heidegger. *Being and Time*. Wiley-Blackwell, 1962.

[75] Erik Hollnagel. *Barriers and accident prevention*. Ashgate, 2004.

[76] Erik Hollnagel and Andreas Bye. Principles for modelling function allocation. *International Journal of Human-Computer Studies*, 52:253--265, 2000.

[77] Erik Hollnagel and David A. Woods. *Joint cognitive systems : foundations of cognitive systems engineering*. CRC Press, Boca Raton, Florida, USA, 2005.

[78] E Hutchins. *Cognition in the Wild*. MIT Press, 1995.

[79] Northrup Fowler III, Stephen E. Cross, and Chris Owens. The arpa-rome knowledge-based planning and scheduling initiative. *IEEE Expert*, 10(1): 4--9, February 1995.

[80] P. M. Jones. Cooperative support for distributed supervisory control: Issues, requirements, and an example from mission operations. In *Proceedings of the ACM International Workshop on Intelligent User Interfaces*, pages 239--242, Orlando, USA, 1993.

[81] Josiah R. Collens Jr. and Bob Krause. Theater battle management core system systems engineering case study. Technical report, Lockheed Martin Integrated Systems and Solutions, 2005.

[82] Arthur E. Kirkpatrick, Bistra Dilkina, and William S. Havens. A framework for designing and evaluating mixed-initiative optimization systems. In *Proceedings of the IJCAI-2005 Workshop in Mixed-Initiative Planning and Scheduling*, Edinburgh, Scotland, 2005. URL http://www.cs.rochester.edu/research/mipas2005/final-drafts/kirkpatrick-dilkina-havens.pdf.

[83] Gary A. Klein, Judith Orasanu, Roberta Calderwood, and Caroline E. Zsambok, editors. *Decision Making in Action: Models and Methods*. Ablex Publishing corporation, 1993.

[84] Raymond Kosala and Hendrik Blockeel. Web mining research: a survey. *SIGKDD Explorations*, 2(1):1--15, 2000. ISSN 1931-0145. doi: http://doi.acm.org/10.1145/360402.360406.

[85] Alexander Kott, Victor Saks, and Albert Mercer. A new technique enables dynamic replanning and rescheduling of aeromedical evacuation. *AI Magazine*, 20(1):43--53, 1999. URL http://portal.acm.org/citation.cfm?id=295240.295957&coll=&dl=ACM&CFID=15151515&CFTOKEN=6184618.

[86] Alexander Kott, Larry Ground, Ray Budd, Lakshmi Rebbapragada, and John Langston. Toward practical knowledge-based tools for battle planning and scheduling. In *Proceedings of the Eighteenth National Conference on Artificial Intelligence*, pages 894--899, Edmonton, Alberta, Canada, July 28 - August 1 2002. URL http://www.geocities.com/alexkott/papers/IAAI-2002-CADET.pdf.

[87] Alexander Kott, Robert Rasch, and Kenneth D. Forbus. AI on the Battlefield: An Experimental Exploration. In *Proceedings of the 14th Conference on Innovative Applications of Artificial Intelligence*, 2002.

[88] N. Kushmerick. *Wrapper Induction for Information Extraction*. PhD thesis, University of Washington, 1997.

[89] N. Kushmerick, T. Lau, M. Dredze, and R. Khoussainov. Activity-centric email: A machine learning approach. In *Proc. American Nat. Conf. Artificial Intelligence*, 2006. NECTAR paper.

[90] Nicholas Kushmerick and Tessa Lau. Automated email activity management: An unsupervised learning approach. In *Proceedings of the Conference on Intelligent User Interfaces*, 2005.

[91] Patrik Larsson and Arne Jönsson. Automatic handling of frequently asked questions using latent semantic analysis. In *Proceedings of the IJCAI Workshop on Knowledge and Reasoning in Practical Dialogue Systems*, 2009.

[92] Joel S. Lawson Jr. Command control as a process. *IEEE Control Systems Magazine*, 1981.

[93] Paul E. Lehner. On the role of artificial intelligence in command and control. *IEEE Transactions on Systems, Man & Cybernetics*, 16(6):824--833, November/December 1986.

[94] Ola Leifler. Combining Technical and Human-Centered Strategies for Decision Support in Command and Control --- The ComPlan Approach. In *Proceedings of the 5th International Conference on Information Systems for Crisis Response and Management*, May 2008.

[95] Ola Leifler and Henrik Eriksson. A model for document processing in semantic desktop systems. In *Proceedings of I-KNOW '08, The International Conference on Knowledge Management*, Graz, Austria, September 3--5 2008.

[96] Ola Leifler and Henrik Eriksson. Domain-specific knowledge management in a semantic desktop. In *Proceedings of I-KNOW '09, The International Conference on Knowledge Management*, September 2009.

[97] Ola Leifler and Henrik Eriksson. Communication analysis tools in command and control - open-endedness and transparency as design criteria. Submitted for publication, 2010.

[98] Ola Leifler and Henrik Eriksson. Message classification as a basis for studying command and control communications - an evaluation of machine learning approaches. Submitted for publication, September 2010.

[99] John Levine, Austin Tate, and Jeff Dalton. O-P$^3$: Supporting the planning process using open planning process panels. *IEEE Intelligent Systems*, 15(5): 52--62, 2000.

[100] John Li, Cleo Condoravdi, and Adam Pease. From visual to logical representation -- a GIS-based sketching tool for reasoning about plans. Technical report, Teknowledge Inc., January 9 2000. URL http://citeseer.nj.nec.com/437877.html.

[101] Pattie Maes. Agents that reduce work and information overload. *Communications of the ACM*, 37(7):30--40, 1994. ISSN 0001-0782. doi: http://doi.acm.org/10.1145/176789.176792. URL http://portal.acm.org/ft_gateway.cfm?id=176792&type=pdf&coll=Portal&dl=ACM&CFID=65494353&CFTOKEN=22746799.

[102] John C. Mallery, Roger Hurwitz, and Gavan Duffy. Hermeneutics. In *Encyclopedia of Artificial Intelligence*. John Wiley & Sons, New York, NY, USA, 1987.

[103] Ching man Au Yeung, Nicholas Gibbins, and Nigel Shadbolt. User-induced links in collaborative tagging systems. In *Proceeding of the 18th ACM conference on Information and knowledge management (CIKM'09)*, pages 787--796, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-512-3. doi: http://doi.acm.org/10.1145/1645953.1646053.

[104] Christofer J. Matheus, Mieczyslaw M. Kokar, and Kenneth Baclawski. A core ontology for situation awareness. In *Proceedings of the 6th International Conference on Infromation Fusion*, Cairns, Queensland, Australia, July 8--11 2003.

[105] Raul Medina-Mora, Terry Winograd, Rodrigo Flores, and Fernando Flores. The action workflow approach to workflow management technology. In *CSCW '92: Proceedings of the 1992 ACM conference on Computer-supported cooperative work*, pages 281--288, New York, NY, USA, 1992. ACM. ISBN 0-89791-542-9. doi: http://doi.acm.org/10.1145/143457.143530. URL http://portal.acm.org/ft_gateway.cfm?id=143530&type=pdf&coll=GUIDE&dl=GUIDE&CFID=27157872&CFTOKEN=13039602.

[106] Perry L. Miller. Medical plan-analysis: The attending system. In *Proceedings of the 1983 International Joint Conference on Artificial Intelligence*, 1983.

[107] Thomas E. Miller, Laura G. Militello, and Jennifer K. Heaton. Evaluating air campaign plan quality in operational settings. In *ARPI 1996 Proceedings*, 1996.

[108] Tom Mitchell. *Machine Learning*. McGraw Hill, 1997.

[109] Oskar Morgenstern and John Von Neumann. *Theory of Games and Economic Behavior*. Princeton University Press, May 1944. URL http://www.amazon.com/exec/obidos/redirect?tag=citeulike07-20&path=ASIN/0691003629.

[110] Magnus Morin. *Multimedia Representations of Distributed Tactical Operations*. PhD thesis, Institute of Technology, Linköpings universitet, 2002.

[111] Alice M. Mulvehill and Joseph A. Caroli. JADE: A tool for rapid crisis action planning. In *Proceedings of the 1999 Command and Control Research and Technology Symposium*, 1999.

[112] Héctor Muñoz-Avila, D.W. Aha, L. Breslow, and D. Nau. HICAP: An interactive case-based planning architecture and its application to noncombatant evacuation operations. In *Proceedings of the Ninth Conference on Innovative Applications of Artificial Intelligence*, pages 879--885, Orlando, USA, 1999. AAAI Press.

[113] Karen Myers, Peter A. Jarvis, and Thomas Lee. CODA: Coordinating human planners. In *Proceedings of the 6th European Conference on Planning (ECP-01)*, Toledo, Spain, 2001. URL http://www.ai.sri.com/pub_list/913.

[114] Karen L. Myers. Abductive completion of plan sketches. In *Proceedings of the Fourtenth National Conference on Artificial Intelligence (AAAI97)*. AAAI Press, 1997. URL http://citeseer.ist.psu.edu/myers97abductive.html.

[115] Karen L. Myers, W. Mabry Tyson, Michael J. Wolverton, Peter A. Jarvis, Thomas J. Lee, and M. desJardins. PASSAT: A user-centric planning framework. In *Proceedings of the Third International NASA Workshop on Planning and Scheduling for Space*, Houston, Texas, October 2002. URL http://www.ai.sri.com/pub_list/925.

[116] Karen L. Myers, Peter A. Jarvis, W. Mabry Tyson, and Michael J. Wolverton. A mixed-initiative framework for robust plan sketching. In *Proceedings of the 13th International Conference on Automated Planning and Scheduling*, Trento, Italy, 2003. URL http://www.ai.sri.com/project/PASSAT.

[117] K.L. Myers. Domain metatheories: Enabling usercentric planning. *Proceedings of the AAAI-2000 Workshop on Representational Issues for Real-World Planning Systems*, 2000.

[118] Ian M. Neale. First generation expert systems: a review of knowledge acquisition methodologies. *The knowledge engineering review*, 3(2):105--145, 1988.

[119] *Guidelines for Operational Planning (GOP) -- Guideline document for NATO countries*. North Atlantic Treaty Organisation, 2000.

[120] Joseph D. Novak and Alberto J. Cañas. The theory underlying concept maps and how to construct them. Technical report, Cornell University, 1982.

[121] J. Orasanu and T. Conolly. The reinvention of decision making. In G. A. Klein, J. Orasanu, R. Calderwood, and C. E. Zsambok, editors, *Decision Making in Action*. Ablex Publishing corporation, Norwood, New Jersey, 1993.

[122] Mats Persson. Visualization of information spaces for command and control. In *ROLF 2010 -- The Way Ahead and The First Step*. Gotab Erlanders, Stockholm, 2000.

[123] Suzanne D. Pinson, Jorge Anacleto Louçã, and Pavlos Moraitis. A distributed decision support system for strategic planning. *Decision Support Systems*, 20:35--51, 1997.

[124] Mike Pool, Ken Murray, Julie Fitzgerald, Mala Mehrotra, Robert Schrag, Jim Blythe, Jihie Kim, Hans Chalupsky, Pierluigi Miraglia, Thomas Russ, and Dave Schneider. Evaluating expert-authored rules for military reasoning. In *K-CAP '03: Proceedings of the 2nd international conference on Knowledge capture*, pages 96--104, New York, NY, USA, 2003. ACM Press. ISBN 1-58113-583-1. doi: http://doi.acm.org/10.1145/945645.945661.

[125] Stephen Potter, Austin Tate, and Gerhard Wickler. Using i-x process panels as intelligent to-do lists for agent coordination in emergency response. In Bartel Van de Walle and Murray Turoff, editors, *Proceedings of the Third International ISCRAM Conference*, Newark, NJ, USA, 2006.

[126] Gregg Rabideau, Russell Knight, Steve Chien, Alex Fukunaga, and Anita Govindjee. Iterative repair planning for spacecraft operations using the ASPEN system. In *Proceedings of the International Symposium on Artificial Intelligence, Robotics and Automation for Space*, 1999. URL http://www-aig.jpl.nasa.gov/public/planning/aspen/.

[127] Dnyanesh Rajpathak and Enrico Motta. An ontological formalization of the planning task. In *Proceedings of the International Conference on Formal Ontology in Information Systems*, 2004.

[128] Robert Rasch, Alexander Kott, and Kenneth D. Forbus. Incorporating AI into military decision making: An experiment. *IEEE Intelligent Systems*, 18 (4):18--26, 2003.

[129] Jens Rasmussen. Deciding and doing: Decision-making in natural contexts. In *Decision Making in Action: Models and Methods*. Ablex Publishing corporation, 1993.

[130] Horst W. J. Rittel and Melvin M. Webber. Dilemmas in a general theory of planning. *Policy Sciences*, 4(2):155--169, 1973.

[131] Karol G. Ross, Gary A. Klein, Peter Thunholm, John F. Schmitt, and Holly C. Baxter. The recognition-primed decision model. *Military Review*, pages 6--10, July-August 2004. URL http://www.au.af.mil/au/awc/awcgate/milreview/ross.pdf.

[132] Mehran Sahami, Susan Dumais, David Heckerman, and Eric Horvitz. A bayesian approach to filtering junk e-mail. In *Proceedings of the AAAI-98 Workshop on Learning for Text Categorization*, 1998.

[133] Magnus Sahlgren and Jussi Karlgren. Terminology mining in social media. In *Proceeding of the 18th ACM conference on Information and knowledge management (CIKM'09)*, pages 405--414, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-512-3. doi: http://doi.acm.org/10.1145/1645953.1646006.

[134] Leo Sauermann, Gunnar Aastrand Grimnes, Malte Kiesel, Christiaan Fluit, Heiko Maus, Dominik Heim, Danish Nadeem, Benjamin Horak, and Andreas Dengel. Semantic Desktop 2.0: The Gnowsis Experience. In I. Cruz, editor, *Proceedings of the International Semantic Web Conference (ISWC 2006)*, volume 4273 of *Lecture Notes in Computer Science*, pages 887--900. Springer Verlag, 2006.

[135] Donald A. Schön. *Educating the Reflective Practitioner*. Jossey-Bass, 1987.

[136] Yuval Shahar, Silvia Miksch, and Peter Johnson. *Artificial Intelligence in Medicine*, volume 1211/1997 of *Lecture Notes in Computer Science*, chapter A task-specific ontology for the application and critiquing of time-oriented clinical guidelines, pages 51--61. Springer Berlin/Heidelberg, 1997.

[137] Yuval Shahar, Silvia Miksch, and Peter Johnson. The Asgaard Project: a task-specific framework for the application and critiquing of time-oriented clinical guidelines. *Artificial Intelligence in Medicine*, 14(1--2):29--51, 1998.

[138] Lawrence G. Shattuck and David D. Woods. Communication of intent in military command and control systems. In C. McCann and R. Pigeau, editors, *The Human in Command: Exploring the Modern Military Experience*, pages 279--292. Kluwer Academic/Plenum Publishers, 241 Borough High Street, London, 2000.

[139] Barry G. Silverman. *Critiquing Human Error -- A Knowledge Based Human-Computer Collaboration Approach*. Academic Press, London, 1992.

[140] Barry G. Silverman and Gregory Wenig. Engineering expert critics for cooperative systems. *The knowledge engineering review*, 8(4):309--328, 1993.

[141] Benjamin D. Smith, Barbara E. Engelhardt, Darren H. Mutz, and John P. Crawford. Automated planning for the modified antarctic mission. In *IEEE Aerospace Conference*, March 2001. URL http://www-aig.jpl.nasa.gov/public/planning/mamm/.

[142] Stephen F. Smith, Ora Lassila, and Marcel Becker. Configurable, mixed-initiative systems for planning and scheduling. *Advanced Planning Technology*, 1996. URL http://www.ozone.ri.cmu.edu/publications/ad-pl-tech96.html.

[143] Stephen F. Smith, David W. Hildum, and David R. Crimm. Comirem: An intelligent form for resource management. *IEEE Intelligent Systems*, 20(2): 16--24, 2005. ISSN 1541-1672. URL http://doi.ieeecomputersociety.org/10.1109/MIS.2005.23.

[144] Diane H. Sonnenwald and Linda G. Pierce. Information behavior in dynamic group work contexts: interwoven situational awareness, dense social networks and contested collaboration in command and control. *Information Processing and Management*, 36:461--479, 2000.

[145] Steffen Staab and R. Studer, editors. *Handbook on Ontologies*. International Handbooks on Information Systems. Springer Berlin/Heidelberg, 2004.

[146] Pieter Jan Stappers and John M. Flach. Visualizing cognitive systems: Getting past block diagrams. In *Proceedings of the 2004 IEEE Conference on Systems, Man & Cybernetics*, 2004.

[147] Adelheit Stein, Jon Atle Gulla, and Ulrich Thiel. User-tailored planning of mixed initiative information-seeking dialogues. *User Modeling and User-Adapted Interaction*, 9(1--2):133--166, 1999. URL http://www.springerlink.com/content/v176130346151830/.

[148] Lucy A. Suchman. *Plans and situated actions -- The problem of human-machine communication*. Cambridge University Press, 1987.

[149] Lucy A. Suchman. *Human-Machine Reconfigurations. Plans and Situated Actions*. Cambridge University Press, 2nd edition edition, 2007.

[150] William Swartout and Yolanda Gil. EXPECT: Explicit representations for flexible acquisition. In *Proceedings of the Ninth Knowledge Acquisition for Knowledge-Based Systems Workshop (KAW'95)*, Banff, Canada, February 26-March 3 1995. AAAI.

[151] Juha Takkinen. *From Information Management to Task Management in Electronic Mail*. PhD thesis, Linköping studies in Science and Technology, 2002.

[152] Austin Tate, S.J. Buckingham Shum, J. Dalton, C. Mancini, and A.M. Selvin. Co-opr: Design and evaluation of collaborative sensemaking and

planning tools for personnel recovery. Technical report, Open University Knowledge Media Institute, 2006.

[153] George Tecuci. Training and using disciple agents: A case study in the military center of gravity analysis domain. *AI Magazine*, 24(4):51--68, 2002.

[154] Carnegie Mellon University The Robotics Institute. Intelligent coordination and logistics laboratory. http://www.ozone.ri.cmu.edu/, 2006. URL http://www.ozone.ri.cmu.edu/.

[155] Peter Thunholm. *Military Decision Making and Planning: Towards a New Prescriptive Model*. PhD thesis, Stockholms universitet, 2003.

[156] David Traum, Jeff Rickel, Jonathan Gratch, and Stacy Marsella. Negotiations over tasks in hybrid human-agent teams for simulation-based training. In *Proceedings of the Second Joint Conference on Autonomous Agents and Multiagent Systems*, pages 441--448, Melbourne, Australia, July 14--18 2003.

[157] A. Valente, Yolanda Gil, and William Swartout. INSPECT: An intelligent system for air campaign plan evaluation based on EXPECT. Technical report, USC -- Information Sciences Institute, 1996. URL http://citeseer.nj.nec.com/valente96inspect.html.

[158] Martin L. Van Creveld. *The Art of War : War and Military Thought*. Cassell, London, 2000. Martin van Creveld. ill. (some col.) ; 28 cm.

[159] Wil van der Aalst and Kees Max van Hee. *Workflow management: models, methods, and systems*. MIT Press, Cambridge, MA, USA, 2002.

[160] Lijie Wen, Jianmin Wang, and Wil M. P. van der Aalst. A novel approach for process mining based on event types. *Journal of Intelligent Information Systems*, 32:163--190, 2009.

[161] Norbert Wiener. Cybernetics. *Bulletin of the American Academy of Arts and Sciences*, 3(7):2--4, 1950.

[162] Terry Winograd and Fernando Flores. *Understanding computers and cognition : a new foundation for design*. Ablex Publishing corporation, Norwood, New Jersey, USA, 1986.

[163] Ian H. Witten and Eibe Frank. *Data mining : Practical Machine Learning Tools & Techniques, Second Edition*. Morgan Kaufmann Series in Data Management Systems. Morgan Kaufmann Publishers Inc., 2005.

[164] Rogier Woltjer. On how constraints shape actions. Master's thesis, Graduate School for Human-Machine Interaction, Division of Quality and Human-Systems Engineering, Department of Mechanical Engineering, Linköpings universitet, SE-58183 Linköping, Sweden, 2005.

[165] Rogier Woltjer. *Functional Modeling of Constraint Management in Aviation Safety and Command and Control*. PhD thesis, Linköping studies in Science and Technology, Linköping, Sweden, 2009.

[166] D. D. Woods. Paradigms for intelligent decision support. In *Intelligent Decision Support in Process Environments*, pages 153--173. Springer Verlag, Berlin, 1986.

PAPER I

Ola Leifler. Combining Technical and Human-Centered Strategies for Decision Support in Command and Control --- The ComPlan Approach. In *Proceedings of the 5th International Conference on Information Systems for Crisis Response and Management*, May 2008

# Combining Technical and Human-Centered Strategies for Decision Support in Command and Control: The ComPlan Approach

**Ola Leifler**
Linköpings universitet
olale@ida.liu.se

**ABSTRACT**

ComPlan (A Combined, Collaborative Command and Control Planning tool) is an approach to providing knowledge-based decision support in the context of command and control. It combines technical research on automated planning tools with human-centered research on mission planning. At its core, ComPlan uses interconnected views of a planning situation to present and manipulate aspects of a scenario. By using domain knowledge flexibly, it presents immediate and directly visible feedback on constraint violations of a plan, facilitates mental simulation of events, and provides support for synchronization of concurrently working mission planners. The conceptual framework of ComPlan is grounded on three main principles from human-centered research on command and control: *transparency*, *graceful regulation,* and *event-based feedback*. As a result, ComPlan provides a model for applying a human-centered perspective on plan authoring tools for command and control, and a demonstration for how to apply that model in an integrated plan-authoring environment.

**Keywords**

Decision support, mixed-initiative planning, critiquing, cognitive systems engineering.

## INTRODUCTION

This work presents the ComPlan approach to support plan authoring in command and control. ComPlan combines results from research on both technical as well as human-centered research on mission planning. Our approach is technically related to mixed-initiative planning systems and critiquing systems. Our specific contribution lies in using results from human-centered research on cognitive systems engineering and military decision theory to create a support tool that matches the work process and representation of plan elements of planners.

Designing knowledge-based support tools for planning in military staff and civilian emergency management teams is a challenging task. It can be cumbersome to model all aspects of command and control situations correctly and therefore, many researchers have explored mixed-initiative planning systems as well as critiquing systems to extend the capabilities and usefulness of classical AI systems in realistic planning situations. We can describe research in these areas as a *technical* approach to providing decision support for mission planners. An alternative approach is to study what mission planners actually spend their time on and what requirements their work situation puts on support systems. Such a *human-centered* approach is taken by cognitive systems researchers and researchers who study military decision making. Both approaches have strong merits, and they complement each other well in describing how we can move forward towards usable and capable decision support for command and control. To be successful in realistic settings, however, we argue that decision-support systems research needs to draw from both human-centered as well as technical approaches, and we describe three guiding principles that we maintain are important for plan authoring tools.

In the following sections, we present technical and human-centered research on decision support for mission planning and use these results to support the ComPlan concept.

## BACKGROUND

Our work on ComPlan builds on research in both *mixed-initiative planning*, *critiquing systems*, *cognitive systems engineering*, and *military decision theory*. In this section, we present related work from these disciplines.

#### Mixed-initiative planning

Mixed-initiative planning systems have successfully been deployed for solving logistical problems (Ferguson, Allen, Miller, 1996), to plan space missions (see for example Cortellessa, Cesta, Oddi and Policella, 2004; Bresina, Jónsson, Morris and Rajan, 2004), to help mission commanders plan military operations (see for example Smith, Hildum and Crimm, 2005; Hayes, Larson and Ravinder, 2005), and to plan for large-scale fire fighting (Fdez-Olivares, Castillo, García-Pérez and Palao, 2006).

When using mixed-initiative planning systems, human planners and automated planning systems support each other's actions by producing different parts of the final plan. Some mixed-initiative planning tools allow the user to decide on an overall course of action and suggest methods for dividing a plan into smaller, more specific fragments (see for example Myers, Tyson, Wolverton , Jarvis, Lee and desJardins, 2002; Fdez-Olivares et al. 2006).

Others involve users in modifying plan constraints, search heuristics or solution criteria (Anderson, Anderson, Lesh, Marks, Mirtich, Ratajczak and Ryall, 2000) to control the search for a solution. Most tools incorporate several of these techniques to allow continuous cooperation between users and an AI planner until a final plan is produced.

All of these support systems assume that the internal domain model of the tool is consistent with real situations, and that complete plan specifications can be produced using a human to fill in slots in a template. In ComPlan, we have taken a somewhat different approach. Although plan constraints can be used to maintain plan consistency, they also serve to highlight problems. The user can choose how to use constraints during planning, with the intention that planning should not be restricted by assumptions made in the internal model.

#### Critiquing Systems

Compared to mixed-initiative planning, critiquing systems (Silverman, 1992) present a different approach to using domain models for supporting mission planners. Instead of offering solutions, they compare computer-generated solutions to human ones and only present critical differences. Critiquing in military command and control has been studied in several projects (see, for example, Valente, Gil and Swartout, 1996).

In a way, mixed-initiative planning systems compare to automated planners in much the same way critiquing systems compare to expert systems. Both can be considered to be relaxations of completely automated problem solvers and interact with the user much more closely than their automated counterparts to maintain trust in the system and to capitalize better on the joint capabilities of both human and computer when solving difficult, real-life problems.

In ComPlan, we demonstrate that a critiquing system for planning and a mixed-initiative planning system can complement each other well. Knowledge, which can be used for planning and simulation, can also be used to highlight constraint violations in the same framework.

#### Cognitive systems engineering

Cognitive system engineers study how to design efficient support systems for humans, considering how humans think and behave. As a result of such studies, cognitive systems engineers have devised models for intelligent support systems.

For command and control, cognitive systems researchers stress that intelligent support systems should neither *emulate an expert* nor *supply solutions* to problems (Woods, Johannesen and Potter, 1991, Hollnagel and Woods, 2005). Also, cognitive systems researchers have posited that any participant in a planning process, whether human or computer, needs to make its contribution conspicuous and intelligible (Dekker and Woods, 1999). A planning application should make it clear what actions it performs when modifying a plan and help human planners interpret both the reasoning as well as the results of a joint planning process. As a consequence of this, Dekker and Woods (2002) argue that *event-based information*, *simulation of predicted events* and pattern-based representations should make for effective support rather than automation of command and control.

In ComPlan, we visualize information, provide domain-dependent feedback and manage constraints using the same knowledge source and mechanisms.

### Military decision theory

Planning military missions involves generating a plan on the one hand, but on the other, it is also very much about improving one's understanding of a situation (Shattuck and Woods, 2000). As an example of this, Ferguson, Allen, and Miller (1996) found in a study that only 23% of the utterances in problem-solving dialogues of human planners actually refer to suggesting courses of action. The rest of the communication pertained to establishing a common understanding of the situation and discussing strategy. Based on this analysis, one could argue that a planning support tool should be built to support *all* relevant activities that human planners actually spend their time on, and not only those that involve creating a plan.

However, it is not that easy to establish exactly which work to support in military planning. Traditional planning models established in military doctrine such as the NATO Guidelines for Operational Planning (GOP) (North Atlantic Treaty Organisation, 2000) or the US Field Manual 101-5 (Department for the Army, 1997) *pre*scribe how to plan military operations at various levels, but they may not accurately *de*scribe how planning is performed in practice.

One recently developed model of decision making, the Recognition Planning Model (RPM) (Ross, Klein, Thunholm, Schmitt, Baxter, 2004), describes the activities of military commanders when they plan military operations. The prescribed model for planning as stated by the GOP declares that the planner should gather information without committing to any particular course of action, then form at least three different courses of action, evaluate them in parallel and select the most appropriate one for execution. According to the RPM, this is not always the case. Rather, military commanders tend to commit to a single alternative early in the planning process and use various techniques to adapt the plan to the current situation. This means, generating many different options for complete plans may not be most useful to planners, which has also been noted in field studies on mission planning (Ross et al., 2004).

### Guiding principles

One of the toughest challenges when devising intelligent decision support for mission planners may be to incorporate high-level knowledge-based reasoning in a manner that is acceptable for end-users and offers efficient and clear support. Dynamic situations, where goals and means to solve them can change frequently puts high demands on the design of support tools.

We have elicited three principles from research on command and control that have influenced our work on ComPlan: *transparency*, *graceful regulation* and *event-based feedback*.

#### *Transparency*

When faced with the challenge of creating a domain model for such a complex domain as mission planning for crisis situations, there are basically two options for researchers. The first option is to improve the knowledge elicitation process and support domain experts when describing a domain of interest (see, for example, Kim, 1999; Blythe, Kim, Ramachandran and Gil, 2001).

The other option is to make the domain model open for inspection and modification by the end users. This is an approach which has been used in, for example, the SIADEX system (Fdez-Olivares, J., Castillo, L., García-Pérez, Ó. and Palao, F., 2006). Creating an open knowledge base is also a prerequisite for the support system to act as a "good team player" as described by Dekker and Woods (1999). They claim that *transparency* in the reasoning process, and hence in the knowledge base underlying the reasoning, is necessary for intelligent support systems to be successful.

In the case of an automatic planner, the reasoning process is more or less by design obscured from the user's view although researchers have studied extensively how to use, for example, dialogue interfaces to support human-computer collaboration (see, for example, Myers, K. L., 2003; Ferguson, G., and Allen, J., 2005; Muñoz-Avila, H., Aha, D. W., Breslow, L. and Nau, D., 1999). Still, making formal reasoning comprehensible in partially automated planning systems presents a major challenge. In ComPlan, we have opted to use domain knowledge in applications we believe are easier to inspect than partially automated planning, such as maintaining constraints based on visually represented dependencies and presenting visual feedback (critique) on plan modifications. Also, domain knowledge in ComPlan can be inspected through the use of multiple views representing different aspects of a plan.

#### *Graceful regulation*

The concept of graceful regulation alludes to graceful degradation: the ability of a system to function even in the absence of some components. In our case, the user may at one time restrict the use of a certain constraint and still benefit from other functionality in ComPlan. Some parts of the internal domain model may be known to hold most of the time, yet not always. In such a situation, the support tool must be prepared to accept plan modifications that conflict with its internal reasoning. Otherwise, it will be of little use in situations when its' model is invalid. An alternative to modeling a domain well enough so that these situations do not occur and not consider uncertain knowledge as part of the domain model is to offer *different strategies* when using the knowledge. ComPlan offers the option of switching between *active* and *passive* use of constraints as well as disabling them completely. By setting a constraint as passive, it only notifies the user of possible constraint violations instead of enforcing the declared or implied constraints of the plan. The user may also opt to make a passive constraint active again.

#### *Event-based feedback*

As Dekker and Woods (2002) argue, event-based feedback is an important mechanism for providing decision support in command and control. Researchers on mixed-initiative planning have also noted that support for synchronizing the distributed work of multiple planners may in fact be more important to the success of a military staff compared to semi-automatic plan generation (Myers, K. L., Jarvis, P. A., Lee, T., 2001). Events that trigger feedback in ComPlan include all plan modifications made locally and also those made by other planners, so that feedback can be provided on all plan modification events in a staff environment. Events that trigger feedback could in principle correspond to any directly observable change in the plan or environment. In a mixed-initiative plan generation system, events that modify the plan come primarily from the plan engine itself, which makes event-based feedback more difficult to implement since the user interaction does not directly correspond to changes in the plan.

### THE COMPLAN MODEL

With ComPlan, we believe we have created a model for plan support systems that creates an intersection between the cognitive needs as stated by researchers on cognitive systems engineering and military decision theory on the one hand, and the technical opportunities exploited for support exploited previously in mixed-initiative planning and critiquing systems.

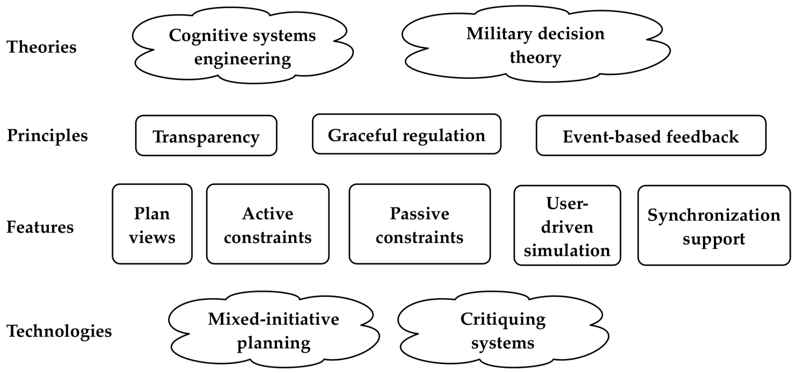


**Figure 1. The main features of the ComPlan model are justified by three guiding principles elicited from human-centered research on Command and Control and based on research on mixed-initiative planning systems and critiquing systems.**

Figure 1 illustrates the concepts introduced by ComPlan and their relation to research on cognitive systems engineering and military decision theory. In the following sections, we present each of them and their relation to previous research.

### Plan views

*Plan views* present plan information to human planners in a way that is conceptually natural to them. In

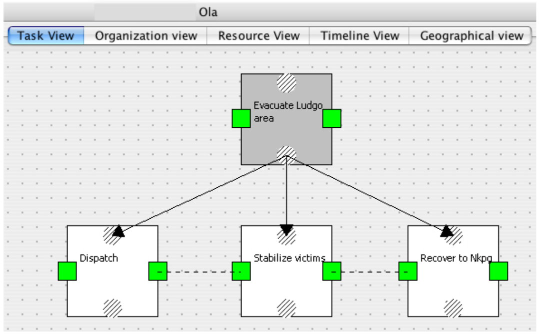


**Figure 2. The *task view* describing a set of tasks to be carried out during a mission, with dependencies marked with lines between them.**

ComPlan, views are accessible and editable in parallel, and use direct, graphical representations close to the ones used in traditional military planning. In our current implementation of the ComPlan model, they include

- the *task view*: a visual, *graph-based representation* of the relationships between tasks in a plan (see Figure 2),
- the *organizational view* which presents the domain knowledge of resources available for the current planning situation,
- *timelines* for tasks and resources that provide an overview of the timing of the plan (see Figure 5), and
- a *map* which presents spatial information and geographical constraints.

In these views, we present users with the option of using feedback mechanisms (passive constraints) that are specific to each view. Each critic presents information either graphically as overlays on the ComPlan interface or in natural language. These *passive constraints* are based on the same technique used to enforce *active constraints*, which help a planner maintain certain aspects of a plan automatically.

### Active constraints

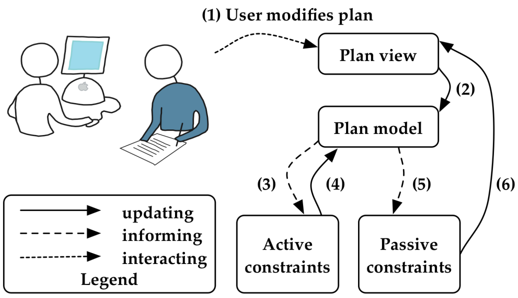


**Figure 3. Ilustration of user interaction with ComPlan during planning. As the user modifies the plan, ComPlan manages active constraints to keep the plan in a consistent state and thereafter notifies passive constraints of the current state**

Active constraints are provided for planners to facilitate plan consistency when making changes. This functionality comes from the active use of constraints in some mixed-initiative planning systems and provides useful help for planners to check the consistency of the plan. As part of planning with a mixed-initiative planning system, users can modify or add constraints in a plan. These constraints may affect the timing of a task, its use of resources or relation to other tasks. In other systems, they can be used to restrict an automated planner when it searches for new plans, or to enforce constraints in the face of plan manipulations initiated by the user. Bresina et al. (2005) call constraints that enforce relationships during user interaction *active constraints* when describing the MAPGEN mixed-initiative system, which is also our intended meaning of the term. By analogy, passive constraints denote constraints that do not enforce relations but rather notifies the user in case of perceived inconsistencies.

Figure 3 shows how ComPlan responds to user interactions and in particular, the relation between active and passive constraints. Active constraints keep the plan in a consistent state as users make changes. Following the updates by both the user and the active constraints, passive constraints can immediately update the current plan view with visual feedback.

**Passive constraints**

All user interactions trigger an evaluation of the plan with respect to both active and passive constraints. Passive constraints, or *feedback mechanisms*, are evaluated after active constraints are processed (see Figure 2) and may, at the user's consent, present information on specific problems related to plan structure, timing or resource allocation. Passive constraints can change *policy* and become active, and active constraints can become passive. The reason for this is to achieve *graceful regulation* of the level of support our tool provides in case the knowledge base is not correct with respect to the current domain.

The analysis performed by our constraints is based on a straightforward model of a planning situation, where much of the underlying assumptions used by the system are exposed to the user and can be modified at run-time. We do not use an extensive knowledge base but have instead opted for a solution where as much of the domain knowledge as practically possible can be inspected and modified through the tool. We base this decision on the principle of *transparency*, which has been stressed as important by both Hollnagel (2005) and Dekker and Woods (2002) in the context of intelligent decision support systems. In ComPlan, all numerical settings on values such as the range and fuel consumption of vehicles, the distances between locations that the timing of missions can be set explicitly through the plan authoring interface.

Apart from using policies for constraint use, we promote transparency by using domain knowledge in user-driven interactive *simulations*.

**Simulation**

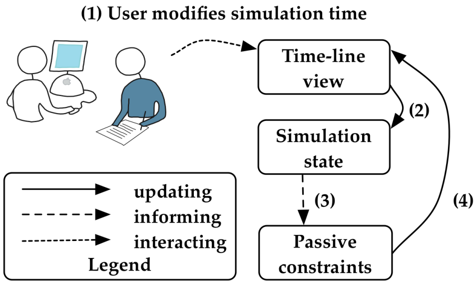


**Figure 4. Illustration of the workflow during simulation in Complan. The simulation is directed by the user through a slider that represents the current time in the simulation.**

Our concept of interactive simulation uses a set of deterministic constraints in a user-controlled[1] simulation that provides immediate access to plan consequences and also feedback based on those consequences. Figure 4 provides an overview of this process. Whenever the user modifies the current time in the simulation, the simulation engine updates the simulation state and notifies constraints and visualization components accordingly. As a result, the user is notified of potential problems if there are any or presented with a visual presentation of the projected state at the selected point in time.

[1] The simulation is *manually* driven forwards or backwards by the user.

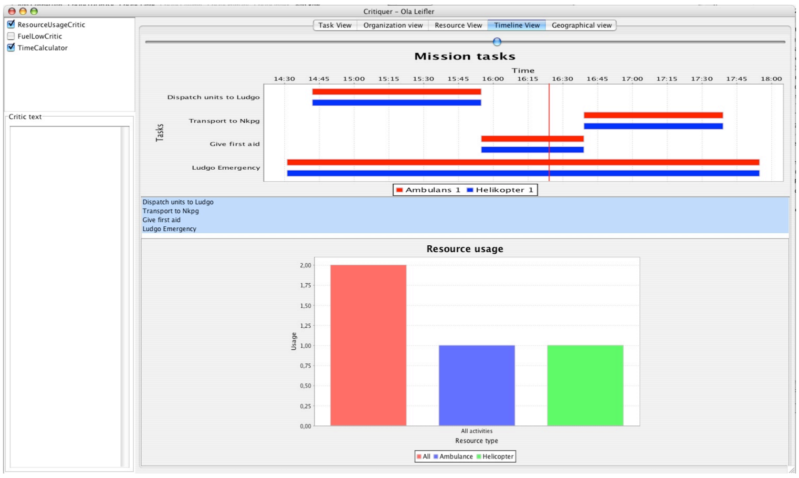


**Figure 5. Screenshot of a simulation performed by the user in the time-line view. The user controls the simulation directly by moving a slider, and receives information immediately on some specific properties of plan at that point in time.**

In Figure 5, we see an example of such a simulation in ComPlan. As a consequence of the user using a slider, the simulation engine advances the simulation to the corresponding point in time, indicated by a vertical bar in the time line. The user has requested information on resource usage, thus a bar chart with this information is displayed below the time line.

Interactive simulation can be useful when a large number of tasks and resources are being planned for. Trying out and simulating many different approaches helps the user evaluate scheduling options. It can also be useful when a mission that has already begun needs to be re-planned and the consequences of small modifications to an existing plan need to be evaluated quickly.

Although it is important to provide simulation of future events and supporting efficient exploration of options for a *single planner*, this form of support is not sufficient in a scenario where many different people collaborate on planning an operation. Therefore, we have devised a mechanism for informing one instance of ComPlan of the actions taken by other instances on the same network, so that several planners can work jointly on planning an operation.

### Synchronization support

A joint staff needs to make sure resources are not oversubscribed and that there are no conflicting intentions among planners. To support the synchronization of work performed by several concurrently working planners, we use the same model for constraint propagation and feedback as described in Figure 3 extended with support for remote planners.

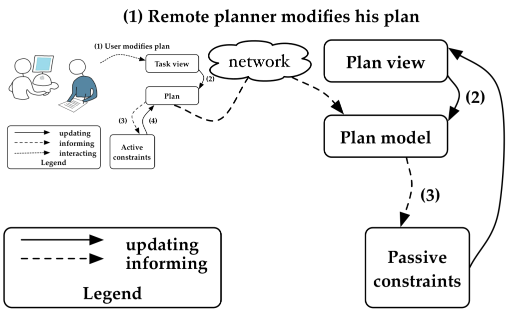


**Figure 6. An illustration of using the actions of remote planners as input to the constraint engine.**

Figure 6 illustrates how planning actions performed in one instance of ComPlan on a network can propagate to other instances. Actions from a remote ComPlan instance do not lead to updates of a local plan, since this would violate the principle of *transparency*, but rather, only passive constraints are fired so that the local view can reflect conflicts between the locally developed plan and plans made by others.

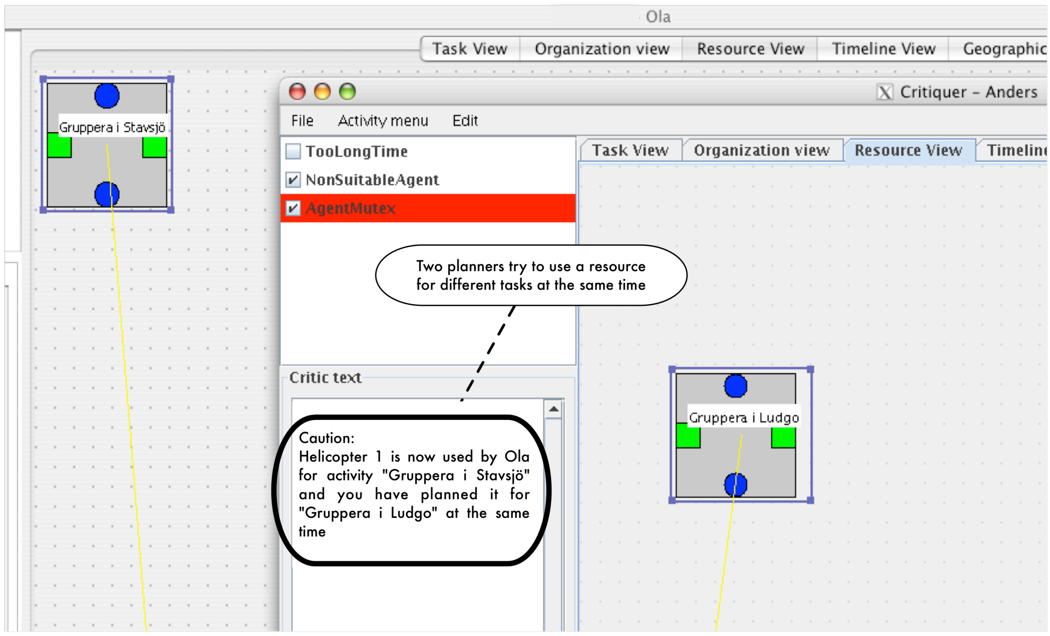


**Figure 7. Illustration of how a critic notifies the user of a conflict in resource usage between two planners.**

Figure 7 illustrates how two instances of ComPlan operate together when on a local network. Once aware of each other, plan modifications performed in one instance of ComPlan are forwarded to remote instances where they are treated as if a local user performed them. In the figure, there is a conflict of resource usage that is signaled as a text message to the local user.

## DISCUSSION

### Plan views

Cognitive systems researchers have long stressed the importance of embedding support systems in the context of work, so as to make use of naturally occurring cognitive artifacts and *enhance* existing practice, not *replace* it (Woods, 1986; Hollnagel and Woods, 2005). In a sense, automated planning systems *replace* some activities during planning by suggestion courses of action based on an initial set of parameters. In experiments with the Recognition-Primed Decision Model, researchers found that commanders were much more interested in visualization support than tools for generating plans (Ross et al. 2004).

ComPlan allows manual planning with the option of maintaining constraints determined by the user. This simplifies the existing practice of calculating the time requirements of different plan options but does not automate any of the cognitive activities during planning. Although ComPlan makes extensive use of multiple, concurrent, directly editable views, the concept has been discussed and implemented partially by other researchers on mixed-initiative systems.

When describing opportunities for future research in mixed-initiative planning, Burstein and McDermott (1996) name specifically "different perspectives of conveying information" as an important area. In research closely related to the agenda for mixed-initiative planning systems, Jones (1993) describes views as important means of enhancing collaboration in mission planning contexts. There are different interpretations of this concept in plan authoring tools. Levine, Tate and Dalton (2000) use views to present human planners with information related to the planning process in the O-Plan plan authoring system and Kim and Blythe (2003) use the concept of views to present time-related information of process models in the KANAL critiquing tool.

Even in the absence of multiple concurrently available views that a planner can switch between, several mixed-initiative tools allow hierarchical specifications of plans using one interface, and some manner of seeing the consequences of the plan as it unfolds using another (see for example Smith, Hildum and Crimm, 2005; Fdez-Olivarez et al., 2006; Myers et al., 2003; Foor and Asson, 2001).

### Simulation

Burstein and McDermott (1996) argue in their research agenda for mixed-initiative planning that simulation should be an integral component for visualizing plans, and that simulation should resemble a movie showing the unfolding of the plan. Using simulations in such ways suggest that simulation results should be *graphical* and used primarily to *emphasize information* relevant to a human planner, not to achieve optimal solutions by iteratively simulating different plans. Modeling and simulation play a central part in military planning as a human process described by the Recognition Planning Model (RPM) (Ross et al., 2004), which characterizes military planning as a process where commanders initially choose a template plan as their preferred option. This template is often based on earlier experiences and is adapted as more information arrives. An important step in the development of a plan is *war gaming*, whereby a plan is subjected to "what if" scenarios and tested iteratively. War-gaming is often performed, individually or in groups, as a mental *simulation* of hypothetical future events to allow better understanding of the possible outcomes of a plan. This is also the intended use of simulations in ComPlan.

In emergency response applications, modeling and simulation have been used for many different purposes, although until recently, such simulations have been isolated from other information systems. Recently, however, researchers have begun to argue for the integration of simulation as part of larger frameworks for decision support (Jain and McLean, 2003), much as simulation is integrated in ComPlan.

### Collaboration support

Myers, Jarvis and Lee claim that for many military planning situations, collaboration tools could turn out to be more useful than automated plan generation systems due in part to the difficulty of modeling the domain (Myers, Jarvis and Lee, 2001). Myers et al. presented a tool, CODA, where planners subscribe to certain types of changes made by others to a common plan. By subscribing to such information, CODA helps planners collaborate through their planning tools between joint meetings. For example, two planners who decide to use the same exclusive resources for a period of time during a mission can be notified of this inconsistency through CODA. In comparison to CODA however, ComPlan can analyze *all* remote plan modifications and not only specific types of changes.

## RELATED WORK

Related work on creating models for plan-authoring tools has mostly been based on technical research on automated planning (mixed-initiative planning) and critiquing.

| System | Plan views | Passive constraints | Active constraints | User-driven simulation | Synchronization support |
|---|---|---|---|---|---|
| PASSAT | | ● | ● | | |
| ComiRem | | | ● | | |
| INSPECT | | ● | ● | | |
| ComPlan | ● | ● | ● | ● | ● |

**Table 1. A Comparison of Planning Systems**

Table 1 presents an overview of previous projects in the area of plan authoring support. Most other related tools are similar in the sense that automatic constraint management or text-based feedback based on complete plan specifications has been the principal concern.

## CONCLUSIONS

ComPlan demonstrates how to combine results from human-centered research with concepts from mixed-initiative planning systems and critiquing systems when supporting mission planners in crisis situations. The concepts in ComPlan are based on three important principles elicited from research on command and control, and the selection and implementation of support techniques in ComPlan have been chosen based on how they support these principles In doing so, we provide a model for more user-centric application of decision support technologies, which can be used to verify claims by command and control researchers regarding how to design decision support systems.

## ACKNOWLEDGEMENTS

This work was supported by grants from the Swedish National Defence College.

## REFERENCES

1. Anderson, D., Anderson, E., Lesh, N., Marks, J., Mirtich, B., Ratajczak, D. and Ryall, K (2000) Human-guided simple search, *Proceedings of the Seventeenth National Conference on Artificial Intelligence*, Austin, Texas.
2. Blythe, J., Kim, J., Ramachandran, S. and Gil, Y. (2001) An integrated environment for knowledge acquisition, *Proceedings of the 2001 International Conference on Intelligent User Interfaces*, Santa Fe, NM, USA.
3. Bresina, J. L., Jónsson, A. K., Morris, P. H. and Rajan, K. (2005) Mixed-initiative planning in MAPGEN: Capabilities and shortcomings, *Proceedings of the IJCAI-2005 Workshop in Mixed-Initiative Planning and Scheduling*, Edinburgh, Scotland.
4. Burstein, M. H. and McDermott, D. V. (1996) Issues in the development of human-computer mixed-initiative planning systems, *Cognitive Technology: In Search of a Humane Interface*, Elsevier Science B.V.
5. Cortellessa, G., Cesta, A., Oddi, A. and Policella, N. (2004) User interaction with an automated solver - the case of a mission planner, *PsychNology*, 2,1,140–162.
6. Department of the Army (1997) *Field Manual 101-5: Staff Organisation and Operations*, Washington, D.C., USA.
7. Dekker, S. W. A. and Woods, D. D. (2002) MABA-MABA or abracadabra? Progress on human-automation co-ordination, *Cognition, Technology & Work*, 4, 4, 240–244.
8. Dekker, S. W. A. and Woods, D. D. (1999) To intervene or not to intervene: The dilemma of management by exception, *Cognition, Technology & Work*, 1, 2, 86–96.

9. Fdez-Olivares, J., Castillo, L., García-Pérez, Ó. and Palao, F. (2006) Bringing users and planning technology together: Experiences in SIADEX, *Proceedings of the Sixteenth International Conference on Automated Planning and Scheduling*, Cumbria, United Kingdom.
10. Ferguson, G., Allen, J. and Miller, B. (1996) TRAINS-95: Towards a mixed-initiative planning assistant, *Proceedings of the Third Conference on Artificial Intelligence Planning Systems*, Edinburgh, Scotland.
11. Foor, L. Z. And Asson, D. J. (2001) Spike: A Dynamic Interactive Component In a Human-Computer Long-range Planning System, *Proceedings of the Third International NASA Workshop on Planning and Scheduling for Space,* Houston, TX, USA.
12. Hayes, C. C., Larson, A. D. and Ravinder, U. (2005) Weasel: A mixed-initiative system to assist in military planning, *Proceedings of the IJCAI-2005 Workshop in Mixed-Initiative Planning and Scheduling*, Edinburgh, Scotland.
13. Hollnagel, E. and Woods, D. A. (2005) Joint cognitive systems : Foundations of cognitive systems engineering, CRC Press.
14. Jain, S. and McLean, C. (2003) Simulation for emergency response: A framework for modeling and simulation for emergency response, *Proceedings of the Thirty-Fifth Conference on Winter Simulation*, New Orleans, LA, USA.
15. Jones, P. M. (1993) Cooperative support for distributed supervisory control: Issues, requirements, and an example from mission operations, *Proceedings of the ACM International Workshop on Intelligent User Interfaces*, Orlando, FL, USA.
16. Kim, J. (1999) Deriving expectations to guide knowledge base creation, *Proceedings of the 12th Workshop on Knowledge Acquisition, Modeling and Management*, Banff, Canada.
17. Kim, J. and Blythe, J. (2003) Supporting plan authoring and analysis, *Proceedings of the 2003 International Conference on Intelligent User Interfaces*, Miami, FL, USA.
18. Levine, J., Tate, A. and Dalton, J. (2000) O-$P^3$: Supporting the planning process using open planning process panels, *IEEE Intelligent Systems*, 15, 5, 52–62.
19. Myers, K., Jarvis, P. A. and Lee, T. (2001) CODA: Coordinating human planners, *Proceedings of the 6th European Conference on Planning*, Toledo, Spain.
20. Myers, K. L. et al. (2002) PASSAT: A user-centric planning framework, *Proceedings of the Third International NASA Workshop on Planning and Scheduling for Space*, Houston, TX, USA.
21. Myers, K. L., Jarvis, P. A., Tyson, W. M. and Wolverton, M. J. (2003) A mixed-initiative framework for robust plan sketching, *Proceedings of the Thirteenth International Conference on Automated Planning and Scheduling*, Trento, Italy.
22. North Atlantic Treaty Organisation (2000) Guidelines for Operational Planning (GOP): Guideline document for NATO countries.
23. Ross, K. G., Klein, G. A., Thunholm, P., Schmitt, J. F. and Baxter, H. C. (2004) The recognition-primed decision model, *Military Review*, July–August, 6–10.
24. Shattuck, L. G. and Woods, D. D. (2000) Communication of intent in military command and control systems, *The Human in Command: Exploring the Modern Military Experience*, Kluwer Academic/Plenum Publishers.
25. Silverman, B. G. (1992) Survey of expert critiquing systems: Practical and theoretical frontiers, *Communications of the ACM*, 35, 4, 106–127.
26. Smith, S. F., Hildum, D. W. and Crimm, D. R. (2005) Comirem: An intelligent form for resource management, *IEEE Intelligent Systems*, 20, 2, 16–24.
27. Valente, A., Gil, Y. and Swartout, W. (1996) INSPECT: An intelligent system for air campaign plan evaluation based on EXPECT, Information Sciences Institute.
28. Woods, D. D. , Johannesen, L. and Potter, S. S. (1991) Human interaction with intelligent systems: An overview and bibliography, *SIGART Bulletin*, 2, 5, 39–50.
29. Woods, D. D. (1986) Paradigms for intelligent decision support, *Intelligent Decision Support in Process Environments*, 153–173.

PAPER II

Ola Leifler and Henrik Eriksson. A model for document processing in semantic desktop systems. In *Proceedings of I-KNOW '08, The International Conference on Knowledge Management*, Graz, Austria, September 3--5 2008

# A Model for Document Processing in Semantic Desktop Systems

**Ola Leifler**
(Linköping University, Sweden
olale@ida.liu.se)

**Henrik Eriksson**
(Linköping University, Sweden
her@ida.liu.se)

**Abstract:** There is a significant gap between the services provided by dedicated information systems and general desktop systems for document communication and preparation. This situation is a serious knowledge-management problem, which often results in information loss, poor communication, and confusion among users. Semantic desktops promise to bring knowledge-based services to common desktop applications and, ultimately, to support knowledge management by adding advanced functionality to familiar computing environments. By custom tailoring these systems to different application domains, it is possible to provide dedicated services that assist users in combining document handling and communication with structured workflow processes and the services provided by dedicated systems. This paper presents a model for developing custom-tailored document processing for semantic-desktop systems. Our approach has been applied to the domain of military command and control, which as based on highly-structured document-driven processes.
**Key Words:** semantic desktop, document-driven processes, semantic documents, planning
**Category:** H.5.3, H.5.4, I.7.1, I.7.5, M.1, M.4

## 1 Introduction

Information handling and knowledge communication in dedicated application systems, such as accounting and planning systems, are typically designed to support users in achieving common organizational goals. Normally, such dedicated information systems assume well-defined information processes and workflows. Conversely, there are generic desktop applications that support users in a wide variety of tasks such as communication via e-mail, document preparation through word processors and information navigation through web browsers. Such general desktop systems do not make any commitments to a particular work process.

Unfortunately, there is a significant gap between dedicated information systems and common desktop applications for everyday work. Often, users must communicate outside dedicated information systems, for example when preparing and sending documents to one another. In case users have needs for communicating or storing information that cannot be managed within dedicated information systems, users resort to generic desktop systems. In such situations, it is difficult to keep the activities in dedicated and generic systems synchronized. In document-driven activities such as planning and reporting, where document preparation and use are core activities, it is especially important to bridge the core process of using dedicated support systems with the document authoring task and to keep the document flow manageable.

Important elements of knowledge management are document preparation, communication, archival, and retrieval. Many knowledge-intensive activities are document-driven in the sense that they focus on document authoring and document use to support human tasks such as planning, decision making, and information dissemination. Today, users take advantage of computer-based office programs such as Microsoft office and OpenOffice to create and edit documents. However, these document-processing environments mainly provide support for text-based editing tasks rather than semantic services to support knowledge management.

*Semantic desktops* are inspired by the semantic web in the sense that the semantic desktop brings semantic-web technologies to the user's desktop [Cheyer et al., 2006, Sauermann et al., 2006, Dong and Halevy, 2005]. A principal idea behind the semantic desktop approach is to support working and reasoning with semantic entities that are normally scattered across several different resources. One of the advantages of semantic desktops is that they promote both formal and informal work processes and flexible information flows. Furthermore, these systems avoid the rigid structures sometimes imposed by traditional dedicated information systems. However, without in-depth knowledge of the application domain (i.e., specific information about the content, structure, and purpose of the documents), it is difficult to provide semantic services beyond general document indexing and search.

Our approach is to extend a pre-existing semantic desktop with domain-specific functionality that enables relevant document analysis based on both the document structure (e.g., outline, tables, and diagrams) and textual content (e.g., keywords, terms, and distinguishing phrases). The development of such semantic-desktop extensions includes modeling of the document workflow, definition of domain concepts in ontologies, and implementation of document-analysis components. The implementation of extensions to a semantic desktop can involve the addition of domain-specific document tracking, indexing and visualization. Naturally, it is necessary to precede this extension development with a traditional requirements analysis identifying the relevant services that the environment should provide.

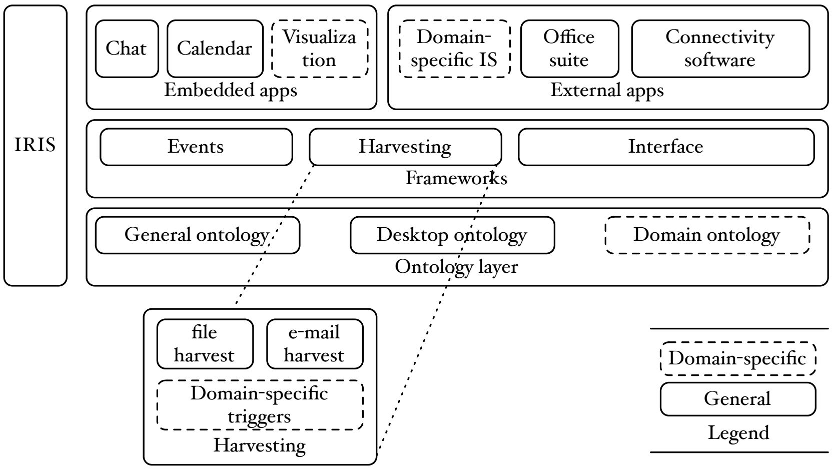


Figure 1: The IRIS semantic desktop model as a three-layered architecture with the ontology at the bottom, the frameworks used to interact with the ontology in the middle and applications at the top.

## 2 Background: Semantic desktop systems

Semantic Desktops [Cheyer et al., 2006, Sauermann et al., 2006] introduce technologies from research on the *semantic web* [Berners-Lee et al., 2001] to the computer desktop with the goal to empower users through improved information-management techniques. The semantic-desktop effort aims at managing information more intelligently through the use of powerful logical formalisms [Baader et al., 2003] for reasoning about semantic entities on the desktop.

Figure 1 presents an overview of the semantic-desktop model. It is based on the notion of an underlying formal ontology, which contains concepts and relationships that pertain to the use of a computer desktop environment. The concepts in the ontology describe interacting with applications, opening and editing files, reading e-mails etc. Also, it contains concepts that relate to semantic entities people refer to in their daily work, and which is what the semantic desktop is built to support. To give an example of such a semantic entity, we often refer to a *person*, although on a computer desktop, a person may be manifested in e-mails as the sender or recipient, in documents as the author, and in calendars as the participant in an event. All these application-specific references are identifiers of the same semantic object, however.

IRIS is a semantic-desktop environment [Cheyer et al., 2006] that contains functionality for harvesting references such as people, tasks and other common concepts from desktop resources. However, the flow of information between parts of an organization may contain much richer semantics than that which can be described using such general concepts. Especially in highly-regulated activities, such as military command and control and medicine, additional information can be inferred from the context of work.

Like many other semantic-desktop environments, IRIS is an extensible framework where functionality can be added at all levels. To support the use of dedicated information systems along with generic desktop applications, the environment needs extensions primarily in the layers concerned with managing the ontology and ensuring that the semantics in the document flow is merged with information from dedicated information systems. The semantics of documents can be retrieved by injecting a context-aware IRIS plug-in that responds to events on the semantic desktop, such as the arrival and submission of documents via e-mail, and the creation and modification of documents. Together with a workflow model based on domain-specific documents, we propose to use semantic desktops as the foundation for new methods of merging and reasoning with information sources.

## 3 Document workflow modeling

The document flow within an organization is an important part of the knowledge-management process. Figure 2 illustrates the information flow between two levels of management. Both levels of management work in parallel to evaluate and plan activities, as well as monitor their progress and coordinate common resources. One way to characterize the interaction and workflow between these levels of management is through the documents that are created, modified, and accessed as part of the process. All traceable actions can be described as communication acts with specific senders and recipients within the organization, or with specific documents as attachments to e-mail communication. Moreover, for each specific domain, the contents of the documents communicated can be described in more detail, such as the character of the *situation brief* submitted by middle management. In the command-and-control domain, such situation briefs can contain descriptions of the current restrictions related to geographical, logistical, legal, and temporal conditions as well as medical restrictions. Also, the directive produced from upper management may contain a decision table with references to objective specifications, due dates for individual tasks, and allocated resources.

In domains where regulations or prevailing practice stipulate a well-defined structure for documents and communications, along with current trends towards standardized formats for documents such as OpenDocument and the proliferation of powerful toolkits for information extraction [Cunningham et al., 2002], a semantic desktop can be augmented with functionality that takes advantage of this structure and uses it to facilitate consistent and efficient use of information.

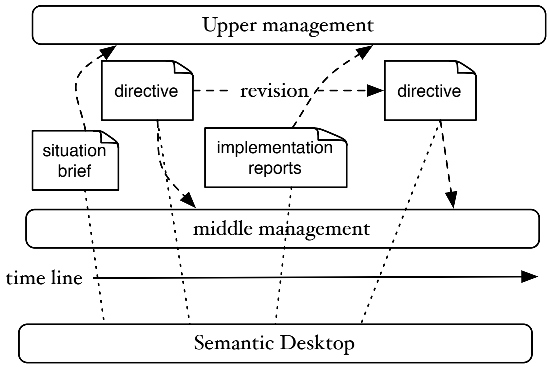


Figure 2: Document flow as part of the collaboration between two levels of management. The documents involved in this communication carry auxiliary information, such as specific file names, structured content, format, author, sender, and recipients. This type of information can assist the information-extraction process.

## 4 Domain-specific semantic desktop information management

Recent standards in document formats (e.g., the OpenDocument and Office Open XML formats) along with advances in natural-language processing simplify the inspection of generic desktop documents. Today, it is a relatively straightforward task to recognize the structure of a document and use that structure to harvest information on its contents. In addition, natural-language-processing tools can classify words and phrases as being references to specific semantic entities, such as people and locations. Together, these advances corroborate the case for the semantic desktop as a viable support tool for knowledge management.

The use of domain-specific information can support users with reconciliation of resources, efficient information navigation, and improved communication in the following manner:

- *Reconciliation of resources* can be performed using the mechanisms already in place in semantic desktops for reifying information [Dong and Halevy, 2005], with the only addition that the source of the information must originate from the dedicated information system as well as text documents. The IRIS environment in particular is designed to act as an open platform that merges semantic references from other applications and other computers through the use of web services.

- *Efficient information navigation* can be supported by IRIS through generic views that present information about the objects on the desktop, such as people and tasks. However, because IRIS is an open platform, it is possible to add views for displaying and navigating among references to locations, resources, teams, and so on. Furthermore, developers can connect these views to the ontology via specific events that inform each view of updates to the specific kind of semantic entities that the view manages.

- *Improved communication* can also be supported by the semantic desktop by using the shared knowledge base. The knowledge base is set up to be shared over a network, along with an event-driven architecture that provides users with notifications of changes or additions to specific categories of semantic entities.

## 5 Implementation for command and control

To demonstrate its viability, we have successfully used our approach in a military command-and-control scenario, where we harvest domain-specific location references from standard OpenOffice documents and provide navigation among those references via a map-based user interface that supports information navigation. As another example, we have identified documents with a specific structure and purpose[1], and extracted domain-specific references to tasks in those documents, based on the structure of the document and the existence of a tabular definition of tasks and responsibilities, to reason about temporal dependencies between tasks. Furthermore, we are currently employing our approach to facilitate communication analysis by modeling the transactions between members of staff.

## 6 Discussion

The use of semantic-desktop environments promotes interoperability across organizations because most of the information interchange is based on e-mail and standard document formats rather than dedicated protocols. This design enables tracking and analysis of documents from outside sources and collaboration partners. In a way, the documents act as user interfaces to the tracking and analysis programs because they affect the actions of these systems.

In this work, we have addressed the analysis of standard documents. However, we believe that it is possible to augment documents with additional metadata to achieve *semantic documents* [Eriksson and Bång, 2006, Eriksson, 2007]. Such semantic document can contain ontologies with concepts that are linked to words, sentences, paragraphs, and other parts of the document. It is possible to add this information to standard document formats, such as PDF, to allow analysis programs to access metadata directly without extracting them from the document text. For example, by representing the content of decision tables in the

[1] to synchronize the use of resources over time during a mission

documents as ontology classes and instances, a semantic-desktop environment could compare documents directly by performing ontology matching. In other words, semantic documents have the potential of retaining semantic information available at the time of document preparation.

A long-term goal for domain-specific semantic-desktop approach it to make it available to a broad range of applications by lowering the effort to custom tailor the environment. In the area of knowledge acquisition, researchers have developed metatools for instantiating domain-specific knowledge-acquisition tools from high-level descriptions [Gennari et al., 2002]. A similar approach could potentially be used to create domain-specific extensions for semantic-desktop environments. For example, we believe that a meta-level tool could generate a domain-specific extension from a combination of domain ontologies and document templates. Naturally, there are many design alternatives for making domain-specific commitments in future generalized semantic-desktop environments and for developers to specify the domain-specific aspects.

Our evaluation of semantic desktops as a viable approach for supporting document-driven staff work has currently been conducted through technical implementation work, based on analyses from participating in and observing actual staff work. The results of our work indicates that semantic desktops can be used successfully to extract and reason about information critical to commanders and represented as parts of the contents or the structure of documents. In military staff exercises, we have identified issues related to the management of temporal and spatial information in documents that was a major cause of concern. Since effective sense making is crucial to a staff, and since that process implies primarily interpreting information jointly, means for organizing the desktop content according to semantic entities instead of the physical structure of documents would make valuable contributions to the work environment of commanders.

## 7 Conclusion

Semantic desktop systems have the potential to enhance document-driven knowledge-management processes through more effective management of semantic entities in documents and communications. One of the main advantages of the semantic-desktop approach is that it supports the users in their daily activities without introducing traditional application systems that require separate streams of information handling. To maximize the benefit of semantic desktops, however, we believe that it is necessary to adapt these systems to the knowledge-management environment of the users and the tasks that the users are performing daily.

A remaining challenge for domain-specific semantic desktops is to find ways to customize the system to the documents and work patterns particular to the application area. Specifically, it is important to streamline the adaptation of document analysis for new document types.

## Acknowledgments

This work was supported by the Swedish National Defence College.

## References

[Baader et al., 2003] [Baader et al., 2003] Baader, F., Calvanese, D., McGuiness, D., Nardi, D., and Patel-Schneider, P., editors (2003). *The Description Logic Handbook*. Cambridge University Press.

[Berners-Lee et al., 2001] [Berners-Lee et al., 2001] Berners-Lee, T., Hendler, J., and Lassila, O. (2001). The semantic web. *Scientific American*, 284(5):28–37.

[Cheyer et al., 2006] [Cheyer et al., 2006] Cheyer, A., Park, J., and Giuli, R. (2006). IRIS: Integrate. relate. infer. share. In *Proceedings of the Semantic Desktop and Social Semantic Collaboration Workshop (SemDesk-2006)*.

[Cunningham et al., 2002] [Cunningham et al., 2002] Cunningham, H., Maynard, D., Bontcheva, K., and Tablan, V. (2002). GATE: A framework and graphical development environment for robust NLP tools and applications. In *Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics*.

[Dong and Halevy, 2005] [Dong and Halevy, 2005] Dong, X. L. and Halevy, A. (2005). A platform for personal information management and integration. In *Proceedings of the Second Biennial Conference on Innovative Data Systems Research*.

[Eriksson, 2007] [Eriksson, 2007] Eriksson, H. (2007). The semantic document approach to combining documents and ontologies. *International Journal of Human–Computer Studies*, 65(7):624–639.

[Eriksson and Bång, 2006] [Eriksson and Bång, 2006] Eriksson, H. and Bång, M. (2006). Towards document repositories based on semantic documents. In *Proceedings of the Sixth Conference on Knowledge Management*, pages 313–320, Graz, Austria.

[Gennari et al., 2002] [Gennari et al., 2002] Gennari, J. H., Musen, M. A., Ferguson, R. W., Grosso, W. E., Crubézy, M., Eriksson, H., Noy, N. F., and Tu, S. W. (2002). The evolution of Protégé: an environment for knowledge-based systems development. *International Journal of Human-Computer Studies*, 58:89–123.

[Sauermann et al., 2006] [Sauermann et al., 2006] Sauermann, L., Grimnes, G. A., Kiesel, M., Fluit, C., Maus, H., Heim, D., Nadeem, D., Horak, B., and Dengel, A. (2006). Semantic Desktop 2.0: The Gnowsis Experience. In Cruz, I., editor, *Proceedings of the International Semantic Web Conference (ISWC 2006)*, volume 4273 of *Lecture Notes in Computer Science*, pages 887–900. Springer Verlag.

# PAPER III

Ola Leifler and Henrik Eriksson. Domain-specific knowledge management in a semantic desktop. In *Proceedings of I-KNOW '09, The International Conference on Knowledge Management*, September 2009

# Domain-specific semantic information management in a Semantic Desktop

Ola Leifler[1], Henrik Eriksson[1]

Department of Computer and Information Science, Linköpings universitet, Sweden
{olale|her}@ida.liu.se

**Abstract.** Semantic Desktops hold promise to provide intelligent information-management environments that can respond to users' needs. A critical requirement for creating such environments is that the underlying ontology reflects the context of work properly. For specialized work domains where people deal with rich information sources in a context-specific manner, there may be a significant amount of domain-specific information available in text documents, emails and other domain-dependent data sources. We propose that this can be exploited to provide much more effective information management by a semantic desktop. In this paper, we present extensions to an existing semantic desktop through content- and structure-based information extraction, domain-specific ontological extensions as well as visualization of semantic entities. Our extensions are motivated by needs in military command and control, where domain-specific, well-structured knowledge is available in documents and communications, but scattered across the desktop. The consistent and efficient management of these resources by a group of co-workers is critical to success. Using a domain-specific semantic desktop, we argue that decision makers will have a much better chance of successful sense making in command and control.

## 1 Introduction

Inconsistent or improper use of information is a major cause of concern in environments where the correct, timely, joint assessment of a situation is paramount to success.

Although specialized information systems have been deployed to alleviate problems with information in many information-intensive environments, it may be that the continued, and perhaps inevitable, proliferation of standard desktop application alongside specialized systems presents the toughest challenge ahead for systems designers.

As part of information management at the strategic level of a larger organisation, documents are communicated within as well as between different levels of management. Being able to correctly assess the information contained in those documents as well as in supporting information systems is critical.

The very use of text documents for communication can be seen as a potential source of errors in this scenario, but word processors, being effective at producing any kind of text document, are universal pieces of software, are not likely to be replaced by custom text entry applications any time soon. A possible remedy to the problem of using different information sources, however, is the use of semantic desktops which can present relevant domain-specific information, from documents as well as other information sources, in a consistent manner for users[1].

Semantic Desktops [5, 15, 23, 8, 22] introduce Semantic Web [2] technologies to a desktop environment in an effort to enhance their reasoning capabilities and empower users through improved information management techniques. There are also several free and commercial software products that attempt to address users' needs to manage information sources by introducing advanced text-based search technologies that can harvest textual information from several sources[2]. Together, desktop search engines and semantic desktop projects provide a platform for building applications that support users at a higher level than previously possible.

Although existing semantic desktops have integrated the use of common concepts such as *people*, *places*, *events* and *tasks* in their ontologies, researchers have also consciously worked to enable semantic desktops to act as open platforms where services and data sources can be added according to people's needs, and the ontologies can be extended to accommodate new circumstances. For reasoning effectively about domain-specific contexts, semantic desktop ontologies do not only need extensions in the form of new classes and properties, but also adaptations for reasoning about domain-specific uses of the ontology concepts already in place, such as a the use of the concept *unit*[3] or the specific connection between concepts such as geographical locations and tasks in military command and control.

We argue in this paper that (1) the ontology-based information management in current semantic desktops need to be extended with domain-specific concepts in contexts where both general and specialized applications co-exist on the desktop, and (2) implementing such extensions creates a semantic desktop environment that can provide significant benefits to information management for command and control.

The remainder of this paper is organized as follows: We outline the process of strategic decision making within the context of command and control in Sec-

[1] Domain-specific information in this case refers to all information that is specific to the domain, and not information that only manifests itself in a particular application. This distinction is important since even general word processors can be used in domain-specific ways in the sense that in content and form, documents can follow a domain-specific format and contain domain-specific information which can be processed and used.

[2] e.g. Google Desktop,Spotlight for Mac OS X, Instant Search for MS Vista and Beagle for the Gnome Desktop

[3] which can refer to a subdivision within a larger military group, a standardized quantity, or a piece of furniture depending on the application domain

tion 2 in relation to the problem of managing domain information correctly. Section 3 explains the rationale for using the IRIS semantic desktop environment in this setting, why we believe it is important to adapt it for domain-specific applications, and how these adaptations are implemented for our scenario in particular. Next, we present an overview of related research in Section 4, discuss the implications of our domain adaptions in Section 5. Finally, we conclude with the contribution of our work in Section 6.

## 2 Command and control

In strategic decision-making environments, which we exemplify with military command and control, correct sense-making is critical to success [3], which in turn requires correct interpretation of available data and well-directed search for new data. Therefore, tools for efficiently navigating the information space available can provide critical advantages. Current computer environments for decision makers include general office applications that co-exist with specialized information systems. This fact can be a seen as an inevitable source of confusion if the same information is represented in both raw text as well as specialized systems. However, text documents produced in such an environment generally have a well-defined structure, and the content follows strict conventions on the use of domain-specific keywords, named entities and typical phrases. Thus, documents in the domain of command and control are amenable to analysis with respect to both contents and structure, which makes it a viable approach to use a semantic desktop as the basis for harvesting and presentation of critical, domain-specific information in documents.

As an example of the information management process in command and control, consider Figure 1, where two levels of command communicate in part by submitting documents to one another prior to launching an operation.

The tactical command reports to the operational command how they would prefer to solve the task given to them (1). This is taken as input by the higher command when they outline the final plan of the operation (2). Finally, the tactical command receives a plan that they are responsible to execute (3), which may or may not take their prior suggestions into account. Suggestions by the tactical staff may either be disregarded deliberately by higher command, or they may be overlooked due to communication errors and misunderstanding in the process. Such misunderstandings often cause problems that can jeopardize the success of an operation.

If the information in documents could be used to explicitly represent an operation, this could help commanders assess documents and messages better and help them relate information in them to that which already exists in specialized sensor systems, geographical systems and report tools.

### 2.1 Information systems in command and control

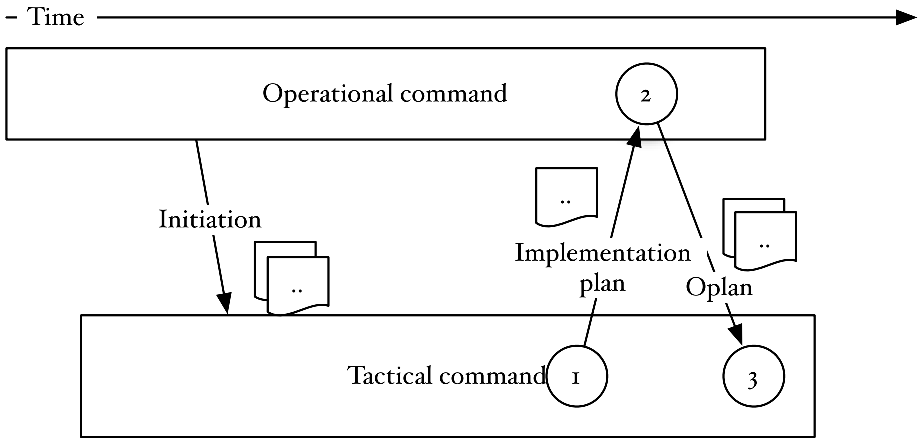


**Fig. 1.** The information flow between the operational and tactical levels of staff. Both staffs work simultaneously with developing their understanding of the current situations and their options. As part of this process they exchange and modify a set of plan documents. In particular, the tactical command develop their suggestion for how to implement the operational objectives prior to the final operation plan.

During the last decade, researchers have done extensive research on the use of intelligent systems for automating strategic decision making [24, 12, 1].

Although recent research has attempted to bridge the gap between automated systems and simpler tools [17] to improve the acceptance of using intelligent support systems, mission plans are still crafted as a set of structured text documents that each describe separate aspects of a plan such as the task organization, rules of engagement, available time frames, force composition, and relevant intelligence information. Figure 1 describes the process of planning as a sequence of communication acts in which structured documents are exchanged between two levels of command. From this picture, we can deduce that, prior to the delivery of the final *operational plan* from the Operational command, the subordinate staff at the tactical level will have to provide an *implementation plan* as part of their preparations. Also, as prescribed in standard documents on planning[4], these documents will have to contain certain information in a distinct format. Figure 2 provides an example of this format in the form of a *synchronization matrix*, which lists the responsibilities of military units at the various stages of the operation. This matrix can be found in specific files during the planning process.

Documents and communications in command and control thus generally follow a well-specified pattern in terms of content and structure. Also, when communicating with one another, members of staff also identify themselves as the function they perform in the staff by code, such as *J4*: responsible for the logistics directorate at the joint, operational level of command. Taken to-

[4] such as the NATO GOP [19]

| ENHET/SKEDE | SPOD | STORÅN | FLYGPLATSEN | Kommmentarer |
|---|---|---|---|---|
| BDE TF MAIN | | - Grp BDE TF HQ MAIN RIMBBO | - Grp BDE TF HQ MAIN RIMBBO | |
| VJ | - Lastar av<br>- Organiserar<br>-anf förberedelser | - Tar tg<br>- VALLENTUNA-E4<br>- Öppnar väg77 | | |
| VL | | - Lastar av<br>- Org<br>Anf förb | Tar Uppsala fplats | |

**Fig. 2.** A synchronization matrix describing the responsibilities of units at the various stages of a fictional operation.

gether, much domain-specific information can be made available to a semantic desktop-based environment for reasoning about the current work context. In military planning, reasoning efficiently with the information available and being able to ask relevant questions about available data is crucial to the success of a mission. In particular, Jensen found in a study on the performance of command teams that information supply was much less important than the *interpretation* of the information given [14]. We argue that semantic desktops can in such situations be used as a help to assess the situation, but only if they present *domain-specific* semantic information derived from documents and communications and not only general, computer-oriented or task-neutral information. By extending the IRIS semantic desktop environment, we demonstrate how to provide such domain-specific information and use it to support situation analysis.

## 3 The IRIS semantic desktop

The IRIS semantic desktop [5] is a modular, service-based semantic desktop project which features an OWL-based object store, integration via plug-ins with external applications such as web browsers, integrated applications such as a calendar as well as a set of support applications for navigating and manipulating the underlying ontology. The applications included within IRIS are general in the sense that they are not specific to a particular work context. This is a feature of IRIS since it is intended to be a general framework for information management. For the purpose of creating domain-specific support in information-intensive work domains, however, we have identified a need for more powerful mechanisms for handling semantic entities that can be found in documents and communications.

In Section 2, we presented an example where the tactical and operational command are supposed to work jointly with the operation plan in preparing for an operational plan. In the example, the operational command receives an implementation plan from the tactical command. Suppose that the operation command fails to recognize that the implementation plan contains a critical piece of information: a suggested SPOD[5] location. Also, suppose that the tactical command of one unit fails to understand their own role in the different stages of the operation and acts prematurely, thus jeopardizing the operation.

Using the semantic desktop, we can aid the decision makers in this case by dealing with issues such as *What locations are referred as* ***Secure Point of Debarkation*** *in the current mission?* and *What are the different stages of the operation and what are the responsibilities of each unit during the first stage?*. To provide commanders with technical assistance with these issues, we propose the following use of the semantic desktop:

– When the operational command receives information from the tactical command that they have prepared an initial implementation plan (step 1 in Figure 1), the operational command opens the implementation plan in the form of a text document and reviews it (step 2 in Figure 1). In doing so, the semantic desktop triggers domain-specific information extraction, which updates the ontology with locations referred to in the document. One of these locations has no previous connection to this document, causing a property update and a notification to the user via messages in the domain-specific information view related to geographical information.
– As part of writing the final operational plan, the operational command formulates a synchronization matrix (see Figure 2) to answer the questions *who* is responsible for *what when*. The file containing the synchronization matrix is classified as a synchronization file by the semantic desktop, causing structure-based harvesting of information on tasks, responsibilities and timing.
  Both levels of staff shares the same semantic desktop environment with the same underlying ontology, so changes of the task descriptions at the operational level are also reflected at the tactical level. Whenever documents containing synchronization information are manipulated at either level, an explicit task representation is updated to reflect these changes (see Figure 7 for an example).

To effectively provide commanders with support in the situations described above requires features that are beyond what is provided by current semantic desktops: (1) the ability to *classify words and phrases* by their semantic class, and (2) the ability to *recognize domain-specific structures* of work-related documents.

As a response to these specific challenges, we have extended the IRIS semantic desktop with *named entity recognition* and *document structure analysis*. We have also extended the IRIS ontology to accommodate our specific domain.

[5] Secure Point of Deployment, denoting a location which can be used as a bridge head and for debarkation point of land-based units arriving by sea.

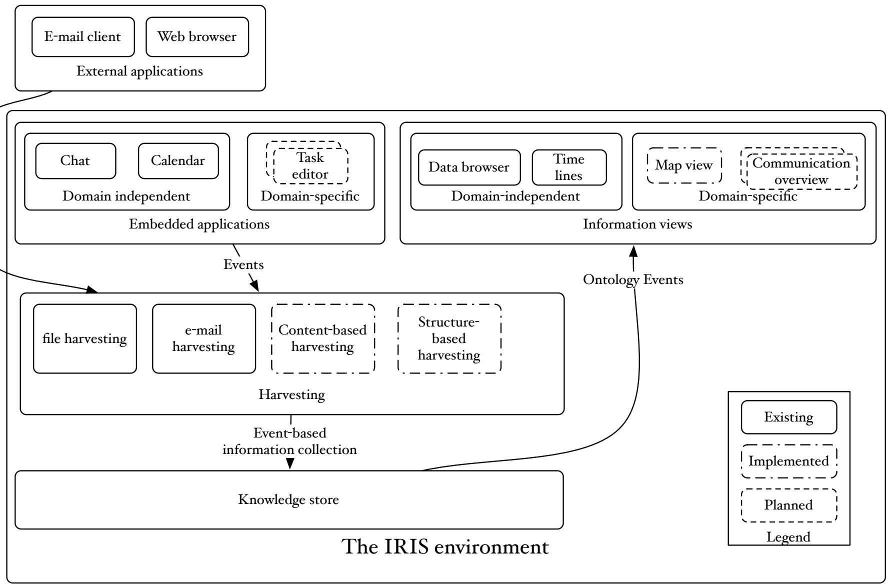


**Fig. 3.** An overview of the IRIS semantic desktop enhanced with domain-specific additions.

### 3.1 Named entity recognition

Current IRIS functionality for harvesting information from files and populating the ontology database relies on the use of the SEMEX [4] information management framework. SEMEX is domain-agnostic as well as IRIS in general, in that no information that is specific to a particular work context is encoded using the current harvesting techniques. For named entity recognition, we have therefore deployed natural language processing components from the GATE framework [6] in IRIS. Using GATE, one can define a grammar of named entities and use the common features of GATE to process documents, annotating them using a set of annotation rules. There are a great number of pre-existing rules defined, and there is even the possibility of linking named entities to classes in an ontology. When annotating documents with GATE, annotations can have attributes that relate to the context of the annotation. This way, we can set the properties of the ontology objects created from the annotation by using such attributes. For the purpose of our current model, connections are primarily made between the annotations and the documents within which they are contained.

### 3.2 Document structure analysis

Apart from the natural language components in GATE, we have set up a mapping between the document structure of Open Document [20] files containing synchronization matrices and the corresponding semantic entities found in the

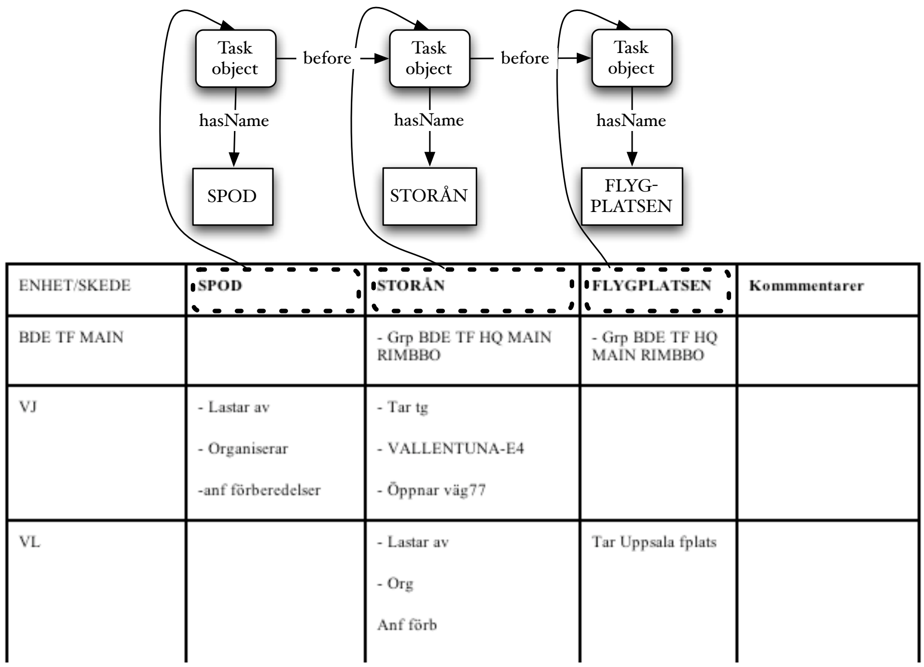


**Fig. 4.** A synchronization matrix contains information about all the stages of an operation, the units involved in each stage and their responsibilities against one another. From these descriptions, we deduce a set of semantic objects and their inter-relationships in terms of temporal ordering.

matrix. This addition to the IRIS harvesting environment was created using primarily XPath and could be instantiated from a general XPath-based OpenDocument parser. Figure 4 describes how specific parts of a synchronization matrix are used to create semantic objects denoting stages of an operation.

The information from named entity analysis and document structure analysis is harvested from documents automatically in IRIS, and is used for navigating domain-specific information more efficiently using special information views.

### 3.3 Domain-specific ontology

The current IRIS ontology is written in OWL-DL [7] and contains concepts related both to the activities performed by applications in a desktop environment, [6], as well as to what users reason about in the environment[7].

[6] such as opening files, launching applications and reading e-mail
[7] such as people, tasks, and places

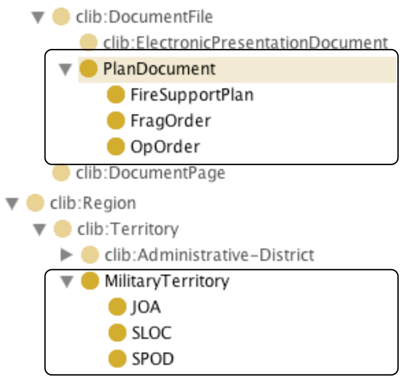


**Fig. 5.** A section of the IRIS ontology, as seen from the Protégé ontology editor. The light-colored classes pertain to the general IRIS ontology, whereas the domain-dependent additions darker. For instance, `PlanDocument` entities represent domain-specific document types that represent document types that are created during the course of managing a military operation.

Creating a domain-specific semantic desktop application, the IRIS environment needs to accommodate both specific *concepts* of a domain, but also specific *uses* of general concepts in that same domain.

Figure 5 shows how domain-specific classes in our scenario relate to the general ontology in IRIS. There are a number of specific classes that describe the work context of command and control, but there are also a number of generally used concepts that carry a specific meaning in this domain. One such example is a *task*, in which a set of military units perform a mission with a specific military purpose. Although the general concept of a task already exists within the IRIS ontology, it is supposed to deal with tasks manifested as items in a calendar, associated with e-mail correspondence or in other ways associated with office work. In the military domain, however, a task may be manifested in an order document which is to be sent to subordinate units. Also, the agents involved in the task may not refer to other people in the same computer environments but rather disparate the military units in the field responsible for carrying out the order. Those units may in turn be referred to primarily in specific places in the documents outlining the task organization, and not as users in a joint computer system. Although there are such differences, it may still not be necessary or even relevant to use a subclass for military tasks, since visualization of tasks and reasoning about their inter-relationships can still be performed using the existing functionality, with the only caveat that the *has-agent* property of a military task is set by harvesting a specific type of document and not by examining calendar och e-mail information.

### 3.4 Using domain-specific content in IRIS

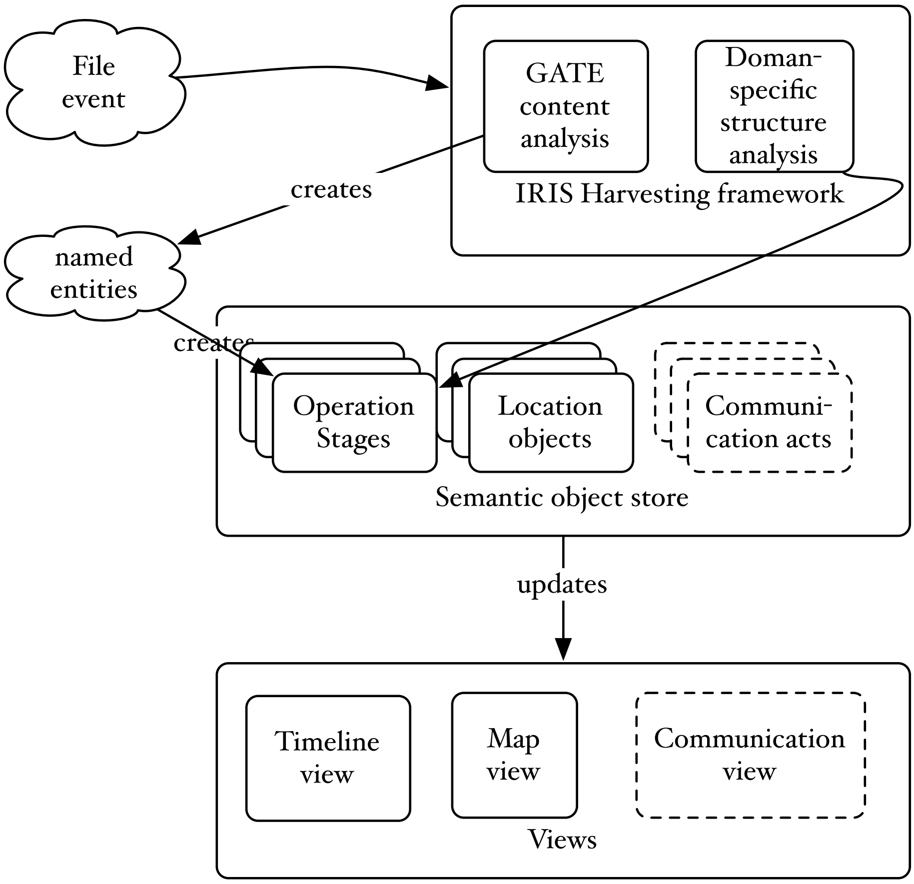


**Fig. 6.** An overview of the harvesting mechanism, once initialized. The object store is updated with information on the stages of an operation and the locations involved in the operation. The communication acts between members of staff is another partially domain-specific piece of information that we are currently adding. Each type of data can be presented using one of the information views.

Once domain-specific harvesting has been set up by coupling named entity recognition in GATE to ontology classes in IRIS, and by recognizing structure-dependent semantic entities and their relations in office documents, this information is processed by IRIS in an event-driven manner as shown in Figure 6. The IRIS framework can automatically notify interested parties of ontology events, so the harvested information can be used to synchronize views on the actual contents of documents.

In Figure 7, a time line describes the stages of an operation, as harvested from a synchronization matrix. Another example of how domain-specific, map-related information can be used to assess the current situation is shown in Fig-

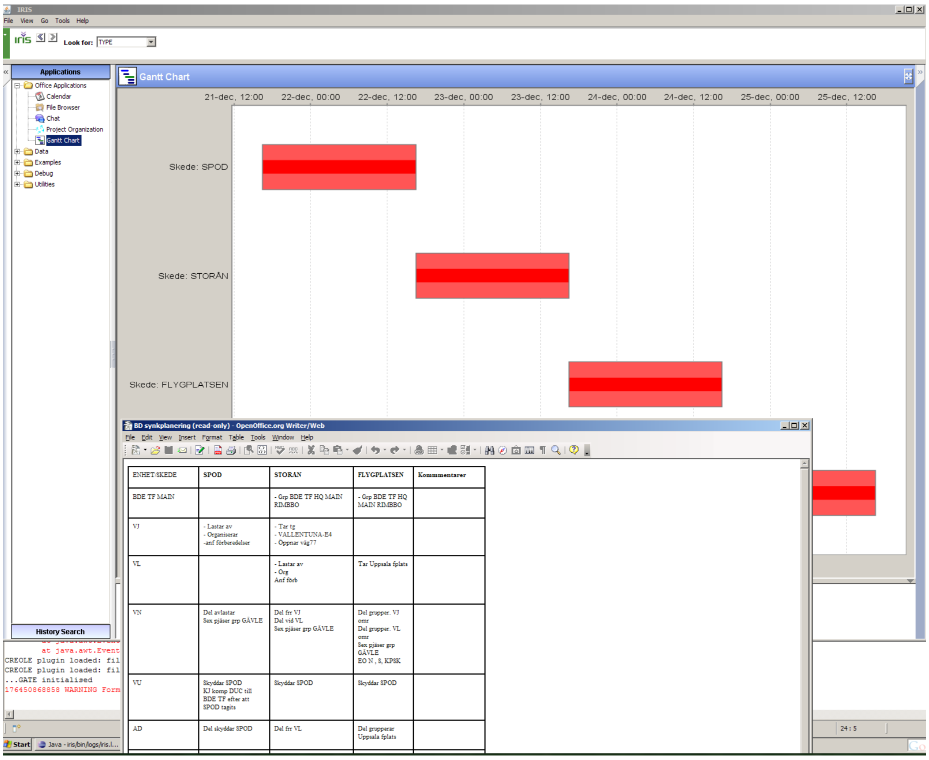


**Fig. 7.** The IRIS environment contains a graphical component to show tasks in a time line. By automatically harvesting synchronization matrices, we can populate the time line with tasks that correspond to those being planned for.

ure 8 where the Secure Point of Debarkation is marked on a map and provides direct access to the documents in which it is referred through the IRIS notification mechanism.

## 4 Related Work

Reeve and Han [21] surveyed how extraction of semantic information can be performed using automated methods. These methods do not require any modification to the editing environment for producing documents, and attempt to minimize the user effort when extracting semantic information. Related to the area of information extraction, Kushmerick [16] introduced the concepts of wrapper induction, in which a system attempts to extract semantically important parts of an the XML tree such as a web page.

In other projects aimed at defining the semantics of documents, researchers have integrated ontology editing capabilities within a document editing environment [11, 9, 25, 18] to provide support for creating semantic annotations

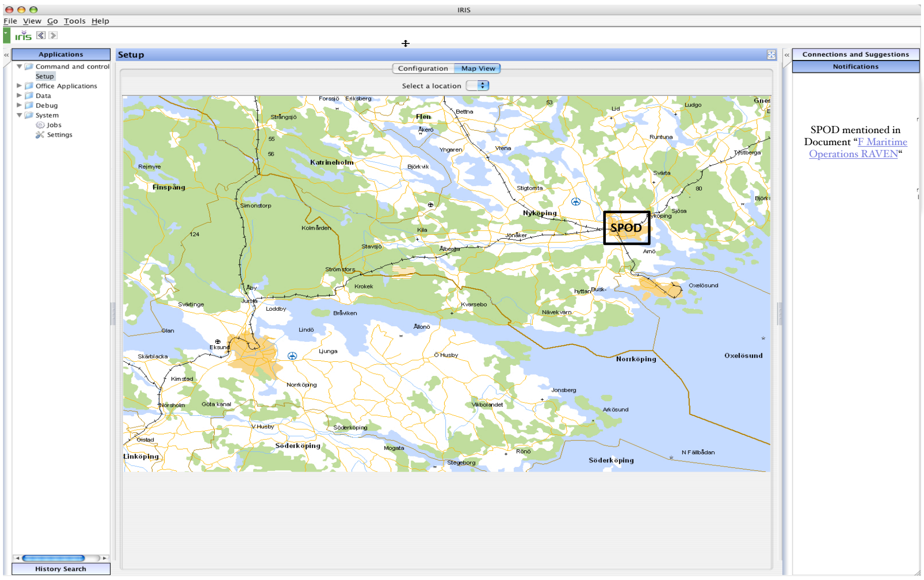


**Fig. 8.** Map view showing a geographic reference plotted on a map and linked to a document. When documents are manipulated, the set of geographical references is updated by using GATE for harvesting location references. Since such location references can include code names or functional locations such as SPOD in the military context, we cannot rely on GIS services to automatically plot them. However, once their correspondence to locations on a map has been established, we can provide visual cues and links to documents that refer to the locations using the IRIS notification framework.

when writing a document. These projects differ in what platform for document production they support, how integrated and transparent the annotation mechanism is and how the annotations are manifested. However, they share an ambition to create document preparation environments in which documents are augmented with annotations with a maximal ratio of expressiveness to human effort.

Franz, Staab and Arndt [10] have demonstrated how to integrate different, application-specific information sources in a semantic desktop framework. Although they refer to *domain*-specific information sources, their framework primarily deals with *application*-specific ones.

## 5 Discussion

In this paper, we have outlined how to extend a semantic desktop environment with a domain-specific ontology to support information visualization and navigation. Semantic Desktops are open platforms since the goal of a semantic desktop is to merge information from different sources and provide an intelligent framework for reasoning with pre-existing entities. Until now, however, few domain-specific applications have been demonstrated.

Our approach to extending the semantic desktop with a richer context model provides a specific advantage for military commanders. However, in any domain where semantic entities are available in well-defined locations in documents or as well-defined components of natural language, the semantic desktop can be extended in a similar manner, allowing an ontology which much more closely captures the context of work. Also, by reasoning about semantic entities found within textual documents, the semantic desktop can create a unified view of entities referred to in such texts and specialized information systems.

### 5.1 Future work

We see two major directions for future technical work, one of which we are currently pursuing.

First, we have observed that apart from using the information inherent in documents on the desktop (or in a shared environment) correctly, *communicating* effectively is crucial in staff work. To support communication with a semantic desktop, future efforts will study how a domain-specific work-flow ontology can be used to support reasoning with and provide feedback to commanders when using text-based communication.

Second, to streamline the creating of domain-specific information harvesting in a semantic desktop, the system should be able to guide the user through a process of defining semantic entities along with procedures for how to extract them from documents and messages. This process could consist of selecting or creating ontology classes and properties, and defining the mapping between these classes and their representation in documents, by either selecting or modifying grammar rules in GATE for recognizing natural language components, or by wrapper induction techniques for recognizing structure elements as proposed by Hogue and Karger [13].

## 6 Conclusions

This article describes the need to enhance semantic desktops with domain-specific ontologies for supporting information-intensive strategic decision making in heterogeneous desktop environments, and describes a model for doing this using natural language processing and document structure analysis. We have demonstrated how to extend an existing semantic desktop (IRIS) this way which provides the ability to reason about semantic entities contained in text documents. Using such techniques, a semantic desktop can accommodate richer environments than otherwise possible, and will also be able to reconcile information from domain-specific systems with structured, general text documents.

## 7 Acknowledgements

This work was sponsored by the Swedish National Defence College.

## References


1. Marcel A. Becker and Stephen F. Smith. Mixed-initiative resource management: The AMC Barrel Allocator. In *Proceedings of the 5th International Conference on Artificial Intelligence Planning and Scheduling (AIPS-2000)*, Breckenridge, Colorado, USA, 2000.
2. T. Berners-Lee, J. Hendler, and O. Lassila. The semantic web. *Scientific American*, 284(5):28–37, 2001.
3. Berndt Brehmer. Understanding the functions of $C^2$ is key to progress. *The International $C^2$ Journal*, 1(1):211–232, 2007.
4. Yuhan Cai, Xin Luna Dong, Alon Halevy, Jing Michelle Liu, and Jayant Madhavan. Personal information management with SEMEX. In *SIGMOD '05: Proceedings of the 2005 ACM SIGMOD international conference on Management of data*, pages 921–923, New York, NY, USA, 2005. ACM Press.
5. Adam Cheyer, Jack Park, and Richard Giuli. IRIS: Integrate. relate. infer. share. In *Proceedings of the Semantic Desktop and Social Semantic Collaboration Workshop (SemDesk-2006)*, 2006.
6. H. Cunningham, D. Maynard, K. Bontcheva, and V. Tablan. GATE: A framework and graphical development environment for robust NLP tools and applications. In *Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics*, 2002.
7. Mike Dean and Editors Guus Schreiber. Owl web ontology language reference. `http://www.w3.org/TR/2004/REC-owl-ref-20040210/`, February 10 2004.
8. Xin (Luna) Dong and Alon Halevy. A platform for personal information management and integration. In *Proceedings of the Second Biennial Conference on Innovative Data Systems Research*, 2005.
9. Henrik Eriksson. An annotation tool for semantic documents (system description). In *Proceedings of the 4th European Semantic Web Conference (ESWC 2007)*, Innsbruck, Austria, 2007.
10. Thomas Franz, Steffen Staab, and Richard Arndt. The x-cosim integration framework for a seamless semantic desktop. In *K-CAP '07: Proceedings of the 4th international conference on Knowledge capture*, pages 143–150, New York, NY, USA, 2007. ACM.
11. Tudor Groza, Siegfried Handschuh, Knud Möller, and Stefan Decker. SALT - semantically annotated latex for scientific publications. In *Proceedings of the 4th European Semantic Web Conference (ESWC 2007)*, Innsbruck, Austria, 2007.
12. C. C. Hayes, A. D. Larson, and U. Ravinder. Weasel: A mixed-initiative system to assist in military planning. In *Proceedings of the IJCAI-2005 Workshop in Mixed-Initiative Planning and Scheduling*, Edinburgh, Scotland, 2005.
13. A. Hogue and D. Karger. Wrapper Induction for End-User Semantic Content Development. *Proceedings of First International Workshop on Interaction Design and the Semantic Web, ISWC*, 2004.
14. Evan Jensen. Sensemaking in military planning: a methodological study of command teams. *Cognition, Technology & Work*, July 2007.

15. David R. Karger, Karun Bakshi, David Huynh, Dennis Quan, and Vineet Sinha. Haystack: A customizable general-purpose information management tool for end users of semistructured data. In *Proceedings of the Second Biennial Conference on Innovative Data Systems Research*, 2005.
16. N. Kushmerick. *Wrapper Induction for Information Extraction*. PhD thesis, University of Washington, 1997.
17. Ola Leifler. Combining technical and human-centered strategies for decision support in command and control — the complan approach. In *Proceedings of the 5th International Conference on Information Systems for Crisis Response and Management*, December 2008.
18. Ronny Löfqvist. Utökning av LaTeX med stöd för semantisk information (extension of LaTeX with support for semantic information). Master's thesis, Linköping university, 2007.
19. North Atlantic Treaty Organisation. *Guidelines for Operational Planning (GOP) – Guideline document for NATO countries*, 2000.
20. OASIS. Open document format for office applications (opendocument) v1.1. http://www.oasis-open.org/specs/#opendocumentv1.1, February 2007.
21. L. Reeve and H. Han. Survey of semantic annotation platforms. *Proceedings of the 2005 ACM symposium on Applied computing*, pages 1634–1638, 2005.
22. Leo Sauermann. The gnowsis semantic desktop for information integration. In *The Semantic Web – ISWC 2005*, 2005.
23. Leo Sauermann, Gunnar Aastrand Grimnes, Malte Kiesel, Christiaan Fluit, Heiko Maus, Dominik Heim, Danish Nadeem, Benjamin Horak, and Andreas Dengel. Semantic desktop 2.0: The gnowsis experience. In I. Cruz, editor, *Proceedings of the International Semantic Web Conference (ISWC 2006)*, volume 4273 of *Lecture Notes in Computer Science*, pages 887–900. Springer Verlag, 2006.
24. Stephen F. Smith, David W. Hildum, and David R. Crimm. Comirem: An intelligent form for resource management. *IEEE Intelligent Systems*, 20(2):16–24, 2005.
25. Marcelo Tallis. Semantic word processing for content authors. In *Proceedings of the Knowledge Markup & Semantic Annotation Workshop*, Florida, USA, 2003.

# PAPER IV

Ola Leifler and Henrik Eriksson. Message classification as a basis for studying command and control communications - an evaluation of machine learning approaches. Submitted for publication, September 2010

**Journal of Intelligent Information Systems manuscript No.**
(will be inserted by the editor)

# Message Classification as a basis for studying command and control communications - An evaluation of machine learning approaches

**Ola Leifler · Henrik Eriksson**

the date of receipt and acceptance should be inserted later

**Abstract** In military command and control, success relies on being able to perform key functions such as communicating intent. Most staff functions are carried out using standard means of text communication. Exactly how members of staff perform their duties, who they communicate with and how, and how they could perform better, is an area of active research. In command and control research, there is not yet a single model which explains all actions undertaken by members of staff well enough to prescribe a set of procedures for how to perform functions in command and control. In this context, we have studied whether automated classification approaches can be applied to textual communication to assist researchers who study command teams and analyze their actions.

Specifically, we report the results from evaluating machine leaning with respect to two metrics of classification performance: (1) the precision of finding a known transition between two activities in a work process, and (2) the precision of classifying messages similarly to human researchers that search for critical episodes in a workflow.

The results indicate that classification based on text only provides higher precision results with respect to both metrics when compared to other machine learning approaches, and that he precision of classifying messages using text-based classification in already classified datasets was approximately 50%. We present the implications that these results have for the design of support systems based on machine learning, and outline how to practically use text classification for analyzing team communications by demonstrating a specific prototype support tool for workflow analysis.

**Keywords** Command and control; classification; exploratory sequential data analysis; workflows mining; random indexing; text clustering

O. Leifler · H. Eriksson
Dept. of Computer and Information Science
Linkping University
SE-581 83 Linkping, Sweden
Tel. +46 13 281000
E-mail: {ola.leifler,henrik.eriksson}@liu.se

## 1 Introduction

Although successful command and control is essential to the success of crisis management and military operations, our understanding of how command and control is performed is still limited (Brehmer 2007). Studying command teams present commanders and researchers with great challenges. First, commanders need to accomodate shifting circumstances and uncertain information about the environment in their work process which makes the work process inherently dynamic (Klein et al. 1993). Second, the use of electronic communications and new media in command teams yields large amounts of data (e.g. text communications, audio, video, computer logs) that are difficult for researchers to process.

An important aspect of analyzing command and control is to find critical episodes in the workflow that warrant further study. Currently, most analyses of electronic communication in both situated and distributed teamwork are conducted manually through the use of *classification schemes* (Silverman 2006). A consequence of the significant effort required by manually classifying communications is that only a part of teams' communication patterns can be explored. The prospect of using automatic support for finding relations in command and control communications is therefore appealing.

This paper presents an evaluation of automated approaches for classifying text messages in the workflows of command and control teams by comparing a selection of classifiers with respect to their precision of classifying messages similarly to human experts. Our selection of classification approaches to compare was justified by the requirements of a widely used method for studying command and control, Exploratory Sequential Data Analysis (ESDA) (Sanderson and Fisher 1994). The results from our evaluation are twofold: first, we identify a classification approach which is suitable for use in an ESDA application, and second, based on the precision results attained, we outline how the classification approach could be used to support the study of command and control workflows.

In the following sections, we describe command and control research and the rationale for investigating machine learning approaches for supporting it in Section 2. Section 3 presents research on extracting patterns related to workflows and related concepts from texts. In Section 4 we present the specific data sets we have applied our classification approaches on. We present our classification approaches in Section 5 and the results of classifying messages in Section 6. Based on these results, we discuss their implications on the design of support tools for analyzing command and control communications and present an implementation that uses automatic classification of text messages in Section 7, and Section 8 concludes this paper.

## 2 Background

Command and control researchers investigate *how groups and group members perform their tasks*, identify *performance measures* for the group and study how they could *improve their performance* (Brehmer 2007). There are several frameworks for understanding teams and teamwork (e.g. (Argyle 1972; Salas et al. 2008)). A common representation of team workflows is to use graphs, where nodes represent tasks and arcs denote transitions between tasks. Such graph-based workflow models have been suggested for the analysis and support the coordination of work in various professional settings (Medina-Mora et al. 1992; van der Aalst and van Hee 2002).

One example of a workflow model that aims to describe how members of command teams perform their tasks is the Dynamic Observe-Orient-Decide-Act model in Figure 1.

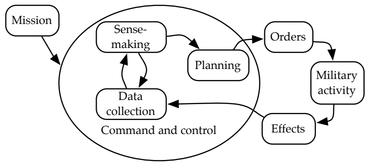


**Fig. 1** The Dynamic Observe-Orient-Decide-Act loop by Brehmer as an abstract model of a workflow in command and control with tasks and transitions between them (Brehmer 2005).

DOODA describes a set of tasks with transitions from one task to another. These tasks can be overlapping or iterating, such as the tasks of *sensemaking* (Weick 1995) and data collection in DOODA. At one point, however, there is a transition from sensemaking to planning, when the commander's intent is formulated and communicated to subordinate units. Irrespective of whether this model accurately describes command and control at a sufficient level of detail for correlating the activities in the model to the observable activities in a command staff, the model could be used as a *hypothesis* for analyzing staff work. If we believe, according to a model such as DOODA, that the staff should begin with *data collection*, and we know that messages of certain types denote a transition to the *sensemaking* step in the DOODA process, then the presence or presence of such types of messages would be part of a researcher's work of establishing performance measures for a command staff.

In general, we can interpret the task of understanding command and control as three separate tasks. First, understanding *how command teams and team members perform their tasks* means constructing a general workflow model such as DOODA from command and control scenarios. Second, establishing direct *performance measures* is synonymous to relating the workflow model to the estimated outcome of scenarios as defined by indirect measurements of scenario outcome (for example, performance scores in computer simulations (Johansson et al. 2003) or evaluations by human experts of team performance in role-playing exercises (Jensen 2007)). Third, *improving performance* is equal to, in each particular scenario, using those performance measures to relate staff actions to the proposed workflow. The process by which researchers establish a workflow and relate staff actions to it from recorded scenario data is based on two principal activities: (1) labeling communication acts with a categorization scheme (Thorstensson et al. 2001), and (2) looking for higher level patterns of episodes (tasks) with the labelled communication acts to focus the search for critical points that have affected the outcome of the scenario (Sanderson and Fisher 1994; Albinsson et al. 2004).

The work of labeling messages according to message categories is the most time-consuming step, with vast quantities of communication data to sift through iteratively, first searching for commonalities that can lead to classification schemes, and later by applying classification schemes to all utterances and reducing the amount of data to a set of episodes based on the classification. This is also the activity for which we evaluate the use of machine learning techniques.

## 3 Related work

The problem of inferring activities from text-based communications has been studied previously by Kushmerick and Lau (Kushmerick and Lau 2005). Their approach was based on searching for specific syntactic patterns originating from the use of computer software (ecommerce systems). Those patterns were in turn used to organize messages into workflows. Patterns originating from the use of computer systems has also been studied by the workflow management community (van der Aalst et al. 2003) where workflows have been elicited from interactions with workflow management systems or other software systems. Both these approaches concern the mining of machine-generated patterns, not patterns originating from human activities.

Regarding the recognition of human activities from text, Scerri et al. (2008) have proposed a model for human workflow management in a semantic desktop environment that relies on the detection or tagging of speech acts in e-mail. Their approach is based on Speech Act recognition performed by a speech act extraction web service which uses grammar patterns for detecting speech acts. Their stated application is to support individuals by monitoring unresolved issues in e-mail conversations such as unanswered questions. Other researchers have described an approach to workflow mining from unstructured data which relies on the existence of a fixed, known number of activity types or named entities in messages for determining which activity a message pertains to (Wen et al. 2009; Geng et al. 2009). Mainly, however, the problem of extracting patterns from e-mail has been studied for the purpose of filtering spam (Sahami et al. 1998) which is essentially equivalent to considering whether a message is at all related to any kind of activity the user is engaged in.

Several projects have attempted to elicit patterns of a domain-specific discourse, mainly from questions and responses sent between customers and company support lines for the purpose of helping customer support identify previous, relevant answers to new questions (e.g. (Larsson and Jönsson 2009; Chalamalla et al. 2008)).

In document management, researchers have studied approaches to relate specific domain knowledge in the form of concepts, objects and relations to textual documents (McDowell and Cafarella 2006; Eriksson 2007) and based on such semantic documents, some projects have studied how to create support for information management in team workflows by using domain-specific document features (Franz et al. 2007; Leifler and Eriksson 2009).

## 4 Material

We used three data sets to establish how well machine learning approaches would classify messages compared to human classification. Our data sets came from three command and control scenarios (Labeled *ALFA -05*, *C3Fire -05* and *LKS* from the projects they originate from) in which crisis management teams had used free text-based means of communication for coordinating their work (fending off forest fires in *ALFA -05* and *C3Fire -05*, and defending against information warfare in *LKS*). In all settings, the participants engaged in activities they were likely to encounter in their profession and the settings used had authentic chains of command and scenario descriptions. The tasks in each scenario were conducted as simulated exercises where the participants collaborated in teams to solve a task. Their performance had been assessed by the staff leading the exercises, which in all cases consisted of researchers studying team performances.

### 4.1 ALFA -05

The *ALFA -05* dataset consisted of 849 text messages exchanged between seven commanders in a simulated crisis response scenario (Trnka et al. 2006). During the scenario, commanders operated at three levels of command, in two administrative areas (approximately county-sized areas) and played a role-playing simulation exercise (Trnka and Jenvald 2006) in which there was initially a forest fire but subsequently also an evacuation from a zoo as well as a search and rescue operation. Participants communicated with one another through a text-based messaging system designed for use in micro-world simulations with the $C^3$Fire simulation environment (Johansson et al. 2003), although it shared the basic features of an e-mail messaging system without the use of subject lines or other auxiliary e-mail headers. The scenario was played over the course of one day.

Each message had been assigned one of 19 different classes by hand. These classes fall into four speech-act-related categories tied to the functions of command and control (Trnka et al. 2006). The four categories were *questions*, *information*, *commands* and *other messages*. When researchers had looked for patterns in the ALFA -05 dataset, they had studied both the general proportions of messages of each class sent to and from the participants in the scenario, but they had also studied specific sequences of speech acts, such whether as a set of *information* and *question*-labelled message exchanges had preceded a *command*.

### 4.2 C3Fire -05

The C3Fire -05 dataset was similar to ALFA -05 with regard to the scenario played and the categorization used. It consisted of 619 messages. One of the main differences was that it was categorized by two independent researchers with a 77.86% agreement between the two on which category to assign each message (the agreement was 87.02% when considering only the four main categories described in the section above). Only those messages which had been classified similarly by the two researchers were selected for classifier comparison. The other main difference compared to ALFA -05 messages was the participants of the study, who were domain experts in the ALFA -05 scenario and students in the C3Fire -05 scenario.

### 4.3 LKS

The *LKS* dataset consisted primarily of 113 e-mail messages exchanged during a training exercise concerning information warfare at the Swedish Defense Research Institute. All participants were experts in the domain and the exercise served the dual purpose being of exercise for them as well as a study of performance indicators in command and control. The scenario was role-played over the course of two days and the participants received instructions from their higher command to engage in intelligence operations for the first day to find information about and locate potential terrorists and monitor, and repel threats during an evacuation of a VIP during the second day of their operation.

Due to these instructions, we categorized the e-mail exchanges pertaining to the first day as *intelligence* and those from the second as *evacuation*, which was consistent with the expected outcome of the exercise. The manual classifications of both datasets were used as validation of the automatic classification approaches we report in this paper.

## 5 Method

To verify that the information in messages could be used for distinguishing contextually significant classes of messages from one another[1] consistent with how command and control researchers would classify messages, we message meta-data to our datasets that we believed to be relevant to classification. With these datasets, we conducted a comparison between several classification approaches by using standard methods for evaluating Machine Learning algorithms.

Messages in a military command and control workflow usually contain domain-specific attributes such as the rank and role of participants. Also, researchers may classify according to the appearance of question marks and the grammatical structure of messages. To understand how these attributes affect automated classification, we compared the impact on classification results of encoding these attributes as part of the message instances. We also evaluated the relative significance of non-text attributes in relation to the text by using a combined classifier that would use a text-based classifier and a non-text classifier in combination for classification. The combined classifier would also provide information on the relative contributions of a non-text classifier compared to a text-based one.

Apart from domain-specific message attributes which are likely to influence human classifications of messages, we considered the influence of a numerical attribute with statistically significant differences of attribute values across the categories of messages: message length. To establish whether a significant difference in message lengths would be used by a classifier when building a classifier model, we studied whether a standard discretization approach (Fayyad and Irani 1992) (as required by the classifiers we evaluated) would generate meaningful nominal interval values and if so, what precision results the classifiers would attain.

We also considered the precision of a random classifier and used that as a baseline for comparing the results of using our selected classification algorithms. If our classifier would not find a meaningful distance measure for the purpose of classifying with respect to message categories (in the ALFA -05 dataset) or belonging to different stages in the scenario workflow (in the LKS dataset), the classifier would basically choose a class at random. The precision it could attain for each decision class could then be described as a function of the proportion of instances of each decision class in the training data.

The precision of the algorithm is expressed as the number of times the algorithm answers correctly, divided by the total number of questions asked. Thus, it is the sum of the number of correct classifications with respect to each of the classes. A completely random classifier, given a dataset $U$ and a function $d$ for mapping messages to the domain of decision classes $\{c_1, c_2, \ldots, c_l\}$ where the sizes of each class is $|c_i| = |\{x \in U : d(x) = c_i\}|$ would attain precision of Equation 1.

$$\Sigma_{i=1}^{l} \frac{|c_i|}{|U|}^2 \tag{1}$$

The LKS dataset consisted of two classes, evenly distributed with 61 messages from day one and 52 from day 2. The random precision would be $(61/113)^2 + (52/113)^2 = 0.5032$, close to 50%. Given the distribution of decision classes in Table 1, random precision attainable in the ALFA -05 dataset was $0.23^2 + 0.39^2 + 0.17^2 + 0.24^2 = 0.29$. Classification results of approximately 29% in ALFA -05 would therefore be attributed to the distribution of messages and not to the message contents. For the C3Fire -05 dataset, the distributions of

[1] such as identifying the two tasks in the LKS dataset or the message classes related to speech acts in the ALFA -05 and C3Fire -05 datasets

**Table 1** Frequency of messages in each of the four message categories of the ALFA -05 dataset.

| Category | Proportion |
|---|---|
| Questions | 23% |
| Information | 39% |
| Orders | 17% |
| Other messages | 24% |

**Table 2** Message lengths in all categories

| | Information | Commands | Questions | Other |
|---|---|---|---|---|
| **Mean** | 110 | 76 | 90 | 55 |
| **Median** | 92 | 58 | 68 | 32 |

classes was more even for both sets of classifications from the two researchers, resulting in random precision of 24.04% and 25.07% respectively.

When evaluating the different approaches to classify messages, we used a stratified cross-validation (Witten and Frank 2005) on each dataset. To accomodate the execution times of text-based classification, we decided to use a 3-times 3-fold stratified cross-validation on our datasets for evaluation. The results were stable when confirmed with a train-and-test procedure on each dataset.

### 5.1 Message lengths

The messages from the four main categories of the ALFA -05 dataset were compared with one another with respect to the lengths of the messages in each category. Since the different message categories contained a different number of messages and the message lengths could not be assumed to be normally distributed, we compared the differences with a non-parametric Mann-Whitney U-test. All categories of messages were compared to one another pairwise. All pairs of categories displayed significant differences in message lengths ($p < 0.002$) and the mean and median values differed as outlined in Table 2.

### 5.2 Classifier selection

The classification schemes we used for both text classification and non-text classification on our datasets were selected based on two primary criteria:

1. the models built as part of learning patterns in data should be *accessible to human inspection*, and
2. they should be computationally *tractable for interactive use* in both scenarios.

The first criterion, accessibility, was considered important because of the prospect of using the resulting classifier model as a basis for a support tool for command and control researchers. In ESDA analysis, exploration means using various data sources in combination to detect patterns of team activity. For a computer-based support tool in this process, establishing trust is critical, and understanding the basis for making classifications could even be more important than high precision for classification, depending on the role of a classifier. The second criterion, computational tractability for interactive use, was considered important for the practical use of automatic classification. In data exploration tools such as

MacSHAPA (Sanderson et al. 1994) and MIND (Thorstensson et al. 2001), researchers navigate scenario data looking for critical episodes by scanning a timeline according to which all scenario data is logged to find incidents that are important for further study. When using such tools, researchers expect interaction with data to be smooth and allow fast manipulations due to the labor-intensive task of finding critical episodes. For an automatic classifier to contribute in such exploration, it would have to build a classifier model fast enough not to interrupt the closer study of data.

Based on these criteria, we selected a text-based classification scheme that would connect important terms as well as the relationships between terms during the process of classification, with the intention of using those terms as part of a workflow analysis tool. Also, it would have to handle the datasets we had with little computational overhead. Based on these criteria, we decided to use the Random Indexing (RI) (Kanerva et al. 2000) vector space model as the primary method of text classification. RI assigns random vectors of a fixed dimensionality to words and texts to create the vector model for measuring similarity between texts (Kanerva et al. 2000). Prior to building the RI model, we filtered the messages so that commonly used, domain-independent words (stop words) would not taint our results. In addition to the RI-based text classification method, a String Subsequence Kernel was also used for analyzing the grammatical structure of messages (see Section 5.4) and for comparison of the RI text classification results. For non-text classification, we used four different classifiers, representing four classes of inference mechanisms:

1. J48, a classifier based on decision-trees (Quinlan 1993)
2. a Decision Table classifier (Kohavi 1995)
3. PART, a rule-based classifier (Frank and Witten 1998)
4. a Nave Bayes classifier (John and Langley 1995)

The first three classifiers were selected based on the accessibility of the models they construct, and the fourth, the Bayesian classifier, was selected due to previously reported results on classifying messages with respect to workflow-related activity types (Geng et al. 2009) with a Bayesian classifier. We conducted the evaluation within the WEKA knowledge analysis framework (Hall et al. 2009), within which we also implemented an RI-based text classifier.

### 5.3 Dataset features

The ALFA -05 and C3Fire -05 datasets differed in several aspects from the LKS dataset:

- All messages had been categorized by hand according to a scheme with 19 speech-act-related categories,
- there were many more participants,
- there were many more texts in both ALFA -05 and C3Fire -05 compared to the LKS scenario, and
- each participant in the ALFA -05 scenario belonged to a certain position in an organization.

We knew that, when analyzing the ALFA -05 scenario by hand to find critical transitions in the workflow, researchers had made use of meta-information that was not encoded directly in the messages. Therefore, we decided to add 4 such additional features for better non-text classifier performance:

```
tack ↦ nn.neu.sin.ind.nom
5914 ↦ rg.utr/neu.plu.ind/def.nom
nn.neu.sin.ind.nom ↦ 1
rg.utr/neu.plu.ind/def.nom ↦ 0
```

**Fig. 2** An example of substitution of phrase structure for message text.

1. The *direction in the chain of command* was the first feature we added, based on the conjecture that the correct classification of a message (such as it being a command or information) would be related to the roles of those involved in the communication. Commands usually travel downwards in the chain of command whereas information usually flows up, from ground units to their superior officers. We therefore encoded the direction in the chain of command as a specific message attribute.
2. The *rank of the sender and recipient* of messages was added as an absolute value in contrast to the relative value of direction in the chain of command (direction being equal to the difference in rank between sender and recipient). The ranks were encoded as nominal values.
3. The *occurrence of question marks* was added to the attributes of messages, with the conjecture that messages with question marks would be labelled *questions* more often than messages of other classes. This was a nominal, binary feature, indicating whether 1 or more question marks occurred in each message.
4. We reasoned that *the phrase structure* of the messages might be related to the manual classification, so that a message classified as a question would display a different sequence of phrase structure tokens than a message classified as an order. The phrase structure of a message was encoded as a substitute for the message text and evaluated using a string-based classifier.

### 5.4 Phrase structure classification

Typically, questions have one particular grammatical structure whereas orders have another. Therefore, as an alternative to using the message text in itself, we extracted the phrase structure of each message in the ALFA -05 dataset and replaced the message text with its (shallow) grammatical structure, so that the message would consist not of a sequence of words but rather of a sequence of phrase grammar indices. The sequence of indices was treated as text that was classified with a kernel-based classifier using a String Subsequence Kernel (SSK) as its kernel function (Lodhi et al. 2002).

Although SSKs are computationally expensive, the texts that we subjected it to were relatively small. We therefore decided to investigate whether an SSK-based approach would attain reasonable results with respect to execution time.

A kernel-based classifier maps a document to a higher-order vector space just as RI. The difference compared to the RI method of reducing the vector space to a more computationally manageable size is that an SSK uses a set of sub-sequences of the text as the feature space that it maps documents into and uses for comparison. A defining feature of an SSK is that it treats strings (documents) as more similar if substrings occur in the same order in both strings. This feature makes it more suitable than RI for comparing whether the grammatical structure of two documents is similar, or phrased differently, if sequences of grammar tokens come in the same order in two texts. The formal alphabet used for an SSK should represent the words available in the *dictionary*, where the dictionary is a set of all distinguishable tokens in the input. When comparing the grammatical structure of texts, we therefore used a

**Table 3** Mean classifier precision results from a 3-by-3 stratified cross-validation on the ALFA -05 datasets

| | J48 | Decision Tables | PART | Nave Bayes |
|---|---|---|---|---|
| Precision (%) | **49,15** | 49,00 | 47,11 | 48,37 |

**Table 4** Mean classifier precision results in percent from a 3-by-3 stratified cross-validation on the ALFA -05 and C3Fire -05 datasets

| | RI | SSK |
|---|---|---|
| **ALFA -05** | 48,96 | **58,85** |
| **C3Fire -05** | **45,67** | 40,25 |

dictionary of phrase structure parts. To extract them, we mapped words to grammar parts and then to simple indices that simplifies the work of the SSK. Figure 2 shows how we mapped a simple acknowledgement message "tack 5914" ("thank you 5914") to a sequence of phrase grammar tokens and subsequently to indices. Ideally, each phrase grammar part should represent a single, unique letter in an alphabet to maximize kernel performance. When used with an SSK-based kernel classifier, we considered the use of indices in a phrase grammar vector as an appropriate approximation that would preserve the discriminating features of a phrase grammar structure.

## 6 Results

We began our classification evaluation by investigating the relative importance of text-based to non-text-based classification when combined in a meta-classifier. The meta-classifier used assigned a weight to each as an indication of the precision of each classifier during training.

### 6.1 Non-text classifier comparison

The four non-text classifiers presented in Section 5.2 were evaluated with respect to classification precision when tested against the man-made categorizations in the ALFA -05 scenario using stratified cross-validation. Table 3 presents the results from applying the classifiers on ALFA -05 dataset. All classifiers performed similarly on the data set and had access to all directly available attributes explained in Section 4.1 and the domain-specific message attributes described in Section 5.3.

### 6.2 Text-based classifier comparison

We compared the Random Indexing-based classifier to the SSK classifier on both the ALFA -05 and C3Fire -05 datasets, as shown in Table 4. The results were inconclusive, as the RI classifier outperformed the SSK on the C3Fire -05 dataset, whereas the SSK classifier performed better on the ALFA -05 dataset. However, the execution time of the SSK classifier was prohibitively high with the ALFA -05 dataset, requiring several hours to build a classification model.

**Table 5** Relative importance of text-based and non-text-based classifiers when determining the class of messages in the ALFA -05 workflow as factors used in the linear regression model of the Stacking meta-classifier.

| Message category | Text | Non-text |
|---|---|---|
| Questions | 0.9204 | -0.0239 |
| Information | 0.9224 | 0.0037 |
| Orders | 0.9304 | -0.0119 |
| Other messages | 0.9299 | 0.0937 |

**Table 6** Prediction precision results from a set of classifier evaluations under different conditions. In each condition, a combined classifier was evaluated on the ALFA -05 dataset with messages tagged as belonging to one of the four different categories.

| Condition | Precision |
|---|---|
| combined classifier | 51% |
| phrase structure classification | 54% |

## 6.3 Comparison of text-based and non-text-based classification

As text-based classification cannot readily be combined with non-text classification, we wished to establish the relative precision of text-based compared to non-text-based classification. To this end, we studied the linear regression model built by a Stacking meta-classifier (Seewald 2003) combining both types of classifiers, which gave us indications that text was the most important feature for classification. In Table 5, we see the relative weights attached to each classifier by the Stacking algorithm when classifying messages of the ALFA -05 dataset as belonging to one of the four categories. The weights signify the relative performance of the algorithms during the training phase of the evaluation. The text-based classifier had much higher precision than the non-text-based one and therefore contributed to a much larger degree to the overall predictions made by the meta-classifier.

Furthermore, we conducted a 3x3 stratified cross-validation of four combined Stacking classifiers which differed with respect to the non-text classifier used. The non-text classifiers in Section 6.1 were combined with an RI-based text classifier and evaluated with respect to classification precision on the ALFA -05 dataset. All four approaches showed similar results (49.20% precision, $\sigma = 3.17$), which indicates that the text-based classifier, common to all four approaches, determined outcome of the combined classifier, as suggested by the regression model in Table 5. Having established that the precision results of a combined classifier on the ALFA -05 dataset did not depend on the non-text classifier, we decided not to explore more option for non-text-based classification. Instead, we evaluated two more options for text-based classification. The results in Section 6.2 were inconclusive, but indicated that the SSK approach was able to provide precision results above the RI-based classifier on some data sets. However, the computational requirements of SSK were prohibitive. As described in Section 5.4, mapping message words to their phrase grammar tokens would make message texts smaller, and therefore possibly computationally more tractable for the SSK approach, while preserving the grammatical structure of the texts.

Table 6 presents a set of results from a train-and-test evaluation of classifier performance. 90% of all messages, randomly selected with representations of all message categories, were used as training messages, and the remaining 10% as tests of classifier accuracy. The SSK classifier used in phrase structure classification continued to have a prohibitively

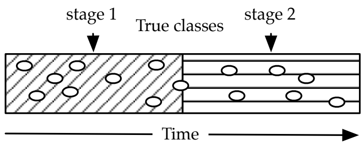


**Fig. 3** The simplest workflow consisting of two stages separated in time. Ovals represents messages along a timeline.

high computation time[2] which would effectively prevent it from being a viable option for classifying messages in an online workflow support system irrespective of precision. It was also the computation time of the SSK classifier that restricted the comparison method to train-and-test as compared to a stratified cross-validation. The combined classifier showed precision results that were within one standard deviation from those obtained in the comparison of the two text-based classifiers.

### 6.4 LKS classification

The LKS dataset consisted of two days' worth of e-mail messages divided in two workflow phases: one for day one and one for day two. 61 messages were sent the first day of the exercise and 52 the second, yielding a total of 113 messages. Our evaluation of classification on this dataset was performed to establish that classifier performance was dependent on the domain-significant "accuracy" of our own division of messages between the two phases, so that a cut between the decision classes *stage 1* and *stage 2* that was not consistent with the real division of messages would yield a comparatively worse predictive performance compared to a more accurate division. If the classification results were noticeably better when dividing the messages at the point in time when the LKS participants got a new task, the classifier would probably pick up on domain-significant features in the message set.

Figure 3 describes a transition in a workflow, where one stage (task) leads to another. Messages in each stage come from one or several actors and are supposed to be associated to only one stage per message. In this trivial workflow, a transition involves the activity of all those involved in the first stage. This representation makes it possible for us to investigate whether a workflow transition would be possible to identify in the simplest possible workflow with a transition: two sequential stages.

To evaluate classification for this purpose, we cut the set of messages in two parts to investigate whether there would be any single point in the message flow where the classifier could perform comparatively better. Specifically, the message flow was divided between *stage 1* and *stage 2* after 20, 30, . . . 90 messages which generated a set of conditions under which a classifier was evaluated (Figure 4) and for each of these conditions, standard cross-validation procedures were applied for evaluating classifier prediction accuracy in comparison to random classification. Since the performance of a classifier may depend on the number of messages in each decision class as noted above, we compared classification results in each condition to those expected from random classification. Our conjecture was that conditions similar to condition B in Figure 4 would yield the highest relative precision.

[2] Approximately 24 hours for finishing a single train-and-test evaluation procedure on the 849 messages in the ALFA -05 dataset.

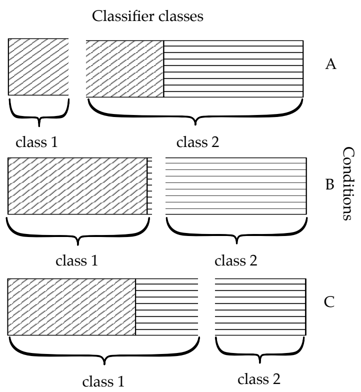


**Fig. 4** Examples of conditions for evaluating the performance of a classifier model with respect to the precision of classifying messages. The "true" workflow stage division is in the middle.

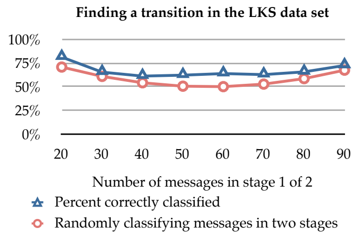


**Fig. 5** The difference in precision of a combined classifier when predicting the workflow stage of messages in the LKS dataset compared to a random classifier.

Condition B describes the case in which classes provided to the classifier for training are most similar to the true workflow stages (Figure 3). If a learning algorithm were only to achieve precision on par with random classification, then the classification would not be a function of the contents of the messages but merely a reflection of the proportions of messages in each decision class.

Figure 5 shows the results of using a combined classifier for determining which of the two stages in the workflow a message belongs to compared to random classification. The diagram shows the precision of classifying messages in the LKS dataset in two categories as a function of the number of messages in the first category (*stage 1*, intelligence). The results in Figure 5 indicate the strongest relative classifier performance compared to random prediction at about 60 messages. 61 messages were sent during the first stage (the first day) and 52 the second, which gave us the indication that the text-based classifier did achieve the best relative performance at the expected point in the workflow.

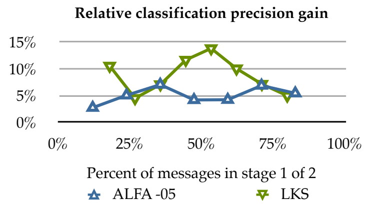


**Fig. 6** The precision of a combined classifier when predicting the workflow stages of messages in the ALFA -05 and LKS datasets minus the results of random classification.

## 6.5 Summary of results

Our classification of command and control text messages was conducted to establish whether man-made classification used attributes that would be significant for automated classification. Our approaches to classify messages yielded these results:

- When comparing non-text classifiers against one another on the dataset that was most rich in non-textual metadata, there was little difference between a rule-based classifier, a decision tree classifier, a Bayesian classifier and a decision-table classifier.
- When comparing two text-based classifiers on the two similar data sets ALFA -05 and C3Fire -05, the results were inconclusive regarding precision. However, a noticeable difference was that one of the approaches (the SSK-based classifier) displayed a prohibitively high computation time.
- A combined classifier which used both the message text as well as other attributes of messages for predicting a workflow transition would almost exclusively use the classifications predicted from the message text, not from other attributes.
- All classification approaches tested on the ALFA -05 dataset yielded similar classification precision results of approximately 50%.
- The LKS classification achieved the highest relative gain compared to random classification at the point in the LKS message flow when participants were expected to move from the first stage of the operation to the second (see Figure 6). This seemed to indicate that the RI-based text classifier would find relevant transitions.

In Figure 6 we summarize the results of predicting which class a message belongs to in a workflow stage. The graph shows precision gains over random classification as a function of the number of messages allocated to the first stage out of two in the respective datasets. For example, at 50 percent the graph shows the relative improvement over random classification when detecting a workflow transition if dividing the datasets evenly in two stages. In the case of the LKS dataset, the highest relative gain over a random classifier was attained at the most even split in two stages which also coincided with the manual, "true" classification. The difference in precision was approximately 15 percent. For comparison, we include the results of classifying the ALFA -05 dataset as if it contained two linearly separable workflow stages, which we had no reason to believe it did.

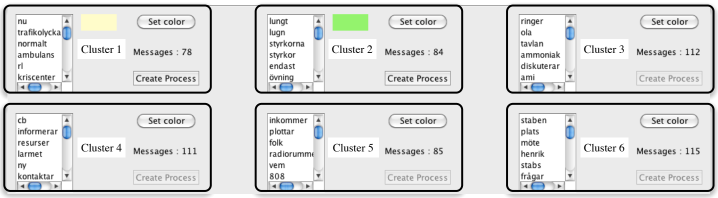


**Fig. 7** The Workflow Visualizer tool uses a text-based classifier to support the analysis of messages in a workflow. Here, a number of possible clusters of messages are identified by the key terms occurring in them. Users can assign colors to messages in a cluster and plot them along a timeline for closer inspection.

## 7 Discussion of results

The LKS dataset consisted of two workflow stages that could successfully be identified using text-based classification.

In a support tool for analyzing command and control communications data using machine learning techniques, a text-based classifier would be able to tell two classes of messages apart in the same manner as a human observer (in the LKS case), but only be able to classify approximately 45-50% of messages similarly to human observers in the ALFA -05 and C3Fire -05 datasets, even though the C3Fire -05 dataset had been sanitized to only contain messages two researchers had categorized similarly.

Classification of the ALFA -05 dataset with respect to message categories yielded precision results of approximately 50 % compared to the expected random precision of 29 %, which indicate that the classification did extract useful information from the message flow with respect to the four categories. The String Kernel-based classification of phrase structure sequences yielded the best precision of all four approaches, but demonstrated a prohibitively high execution time for text classification of even moderately sized corpora, which had previously also been noted by Lodhi et al. (2002). The text classification results were better overall than the non-text results, and in particular, when combined using a meta-classifier, the text-based classification showed higher precision results than the non-text-based classifiers. Taken together, text-based classification would therefore seem to be an appropriate candidate for applying

### 7.1 Implications for support systems

The results from searching for a transition in the LKS dataset and classifying message categories in the ALFA -05 and C3Fire -05 datasets gave indications of the classifier precision for predicting which message classes. The precision was not sufficient for supporting command and control researchers with automation of message classification, but could yield insights on how a set of messages can be divided in clusters that are of importance in the domain. Having established that the precision possible with machine learning approaches is approximately 50% with these datasets, we have some limitations on the possibilities for automatically using classification in the ESDA workflow of command and control research.

However, precision results alone are insufficient for determining the utility of automatic classification as a basis for supporting researchers in the analysis of command and control communications. Although the precision results were low, they were compared against human evaluations, which may have been made according to domain knowledge not encoded properly for machine classification. In exploratory sequential data analysis, there are no predefined quality measures of classification to apply. In fact, categories for labeling messages may be developed as part of the analysis itself, and two independent analysts may categorize the same set of messages differently with the same classification scheme as demonstrated in the C3Fire -05 dataset (see Section 4.2). However, exposing the classifier model of a classifier which has been trained on one dataset to categorize messages from another could possibly provide a valuable support system for exploring possible patterns in command and control communications. When devising such a support system, the issue of how to make the classifier model usable is likely to be the most challenging.

### 7.2 Support system requirements

In the analysis process of ESDA, the requirements for transparency and traceability (Albinsson et al. 2004; Thorstensson et al. 2001) present a challenge for automatic message classification since the classifier must maintain a clear trace between individual messages and the model created of how they relate to one another. To support workflow analysis through classification therefore require us to use the classifier model to primarily *highlight possible relations* among messages during exploration, and require that there are *several options* for how to classify (with respect to message categories, transitions or other workflow-related features). We name these two requirements *transparency* and *graceful regulation* (Leifler 2008).

*Transparency* represents the degree to which a computer-generated model of a dataset, such as a vector-based model of the texts in a communication flow, can be related directly to the underlying data sources the model is based on. Transparency can be achieved by exposing the defining features of the model through a graphical interface where the connection between raw data and computer model is as simple as possible to understand. In the case of a vector-space model of words and texts, this translates into making the terms extracted by text classification part of the interface for selecting and inspecting messages, and part of the description of message clusters. Rules or decision trees extracted by a non-text classifier could be made part of a selection interface in a similar manner.

*Graceful regulation* can be understood the ability to choose different uses of a computer-based model depending on how the user trusts the model and what the user needs. In the case of communication analysis, this translates into using message classification for two distinct purposes: selecting and inspecting parts of the communication flow based on specific key terms or selecting and inspecting clusters of messages according to the model. The former method only requires the user to rely on the computer model to present the most frequent terms, whereas the latter requires the user to trust the vector-based model to produce contextually significant clusters of messages. We have implemented these two requirements in a prototype support tool for exploring relations in communications.

### 7.3 Workflow Visualizer

Our implementation of a support tool for exploring communication patterns is called Workflow Visualizer and consists of components for selecting and visualizing messages that are part of a structured workflow in command and control. Figure 7 shows a view of Workflow

Visualizer, in which an RI-based model has been constructed from a series of crisis management scenarios. The evaluations reported in this paper indicate the contextual validity of the RI approach. Based on these evaluations, we used the vector-based RI representation of message texts for creating message clusters to present *possible* patterns in data. Such patterns can be used by command and control researchers who study $C^2$ teams and trace communication trails for information on how certain concepts have been communicated and understood and how the sensemaking process of the team has worked.

The message clusters were extracted from written communications and text logs in a command and control scenario and are represented in the graphical interface by the most significant terms in each cluster. The clustering shown in Figure 7 comes from a set of 10 similar command and control scenarios studied previously by researchers[3]. During analysis, the researchers were interested in finding if there had been deviations from standard communication patterns in any of the scenarios which could indicate stress or fatigue. One way of studying this is to explore the automatic clusters generated by the Workflow Visualizer. For example, cluster 2 identifies messages that relate to low workload (key terms being "lugn" (calm) and "lugnt" (calmly)). Those messages are not evenly distributed across all 10 scenarios and could give insights into whether the staff had had different experiences during the exercises. In future work, we will evaluate the how command and control researchers will make use of the Workflow Visualizer when used with authentic scenarios to help answer realistic research questions.

## 8 Conclusions

We have established that, given a known transition in a multi-actor workflow manifested in written communication, Random Indexing-based text classification is able to successfully detect the transition through a series of classification trials. We have also established a baseline for the precision attainable when using both text-based and non-text-based classification for identifying classes of messages that are relevant for helping researchers identify transitions in a command and control team workflow. Based on the precision results and a discussion of how to support command and control researchers, we have described two general requirements, transparency and graceful regulation, for tool support in command and control research and presented a prototype tool for supporting $C^2$ researchers find workflow-related patterns in communications.

The most time-consuming work in the analysis of command team behavior is the selection and filtering of data from scenarios and in particular communication data. In the study of structured team work environments such as command and control, we argue that automatic text clustering offers a viable technological basis for interactive exploration and analysis that offers concrete advantages for understanding of how groups of people work.

## 9 Acknowledgments

This work was supported by the Swedish National Defense College. The LKS dataset was contributed by the Swedish Defense Research Agency (FOI) and the ALFA -05 dataset was generously shared by Jiri Trnka, now at FOI.

[3] These scenarios contained uncategorized data and were therefore not part of the evaluations reported in this paper

## References


Albinsson, P.-A., Morin, M., and Thorstensson, M. (2004). Managing metadata in collaborative command and control analysis. In *Proceedings of the 48th Annual Meeting of the Human Factors and Ergonomics Society*.

Argyle, M. (1972). *The Social Psychology of Work*. The Penguin Press, London, UK.

Brehmer, B. (2005). The Dynamic OODA Loop: Amalgamating Boyd's OODA Loop and the Cybernetic Approach to Command and Control. In *Proceedings of the 2005 Command and Control Research and Technology Symposium*.

Brehmer, B. (2007). Understanding the functions of $C^2$ is key to progress. *The International $C^2$ Journal*, 1(1):211–232.

Chalamalla, A., Negi, S., Subramaniam, L. V., and Ramakrishnan, G. (2008). Identification of class specific discourse patterns. In *CIKM '08: Proceeding of the 17th ACM conference on Information and knowledge management*, pages 1193–1202, New York, NY, USA. ACM.

Eriksson, H. (2007). The semantic document approach to combining documents and ontologies. *International Journal of Human–Computer Studies*, 65(7):624–639.

Fayyad, U. M. and Irani, K. B. (1992). On the handling of continuous-valued attributes in decision tree generation. *Machine Learning*, 8:87–102.

Frank, E. and Witten, I. H. (1998). Generating accurate rule sets without global optimization. In Shavlik, J., editor, *Fifteenth International Conference on Machine Learning*, pages 144–151. Morgan Kaufmann.

Franz, T., Staab, S., and Arndt, R. (2007). The X-COSIM integration framework for a seamless semantic desktop. In *K-CAP '07: Proceedings of the 4th International Conference on Knowledge Capture*, pages 143–150, New York, NY, USA. ACM.

Geng, L., Buffett, S., Hamilton, B., Wang, X., Korba, L., Liu, H., and Wang, Y. (2009). Discovering structured event logs from unstructured audit trails for workflow mining. In Rauch, J., Ras, Z., Berka, P., and Elomaa, T., editors, *Foundations of Intelligent Systems*, volume 5722 of *Lecture Notes in Computer Science*, pages 442–452. Springer Berlin / Heidelberg.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. H. (2009). The weka data mining software: An update. *SIGKDD Explorations*, 11(1).

Jensen, E. (2007). Sensemaking in military planning: a methodological study of command teams. *Cognition, Technology & Work*.

Johansson, B., Persson, M., Granlund, R., and Mattsson, P. (2003). C3fire in command and control research. *Cognition, Technology & Work*, 5(3):191–196.

John, G. H. and Langley, P. (1995). Estimating continuous distributions in bayesian classifiers. In *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence*, pages 338–345. Morgan Kaufmann.

Kanerva, P., Kristoferson, J., and Holst, A. (2000). Random indexing of text samples for latent semantic analysis. In *Proceedings of the 22nd Annual Conference of the Cognitive Science Society*.

Klein, G. A., Orasanu, J., Calderwood, R., and Zsambok, C. E., editors (1993). *Decision Making in Action: Models and Methods*. Ablex Publishing corporation.

Kohavi, R. (1995). The power of decision tables. In *Proceedings of the 8th European Conference on Machine Learning*, pages 174–189. Springer.

Kushmerick, N. and Lau, T. (2005). Automated email activity management: An unsupervised learning approach. In *Proceedings of the Conference on Intelligent User Interfaces*.

Larsson, P. and Jönsson, A. (2009). Automatic handling of frequently asked questions using latent semantic analysis. In *Proceedings of the IJCAI Workshop on Knowledge and Reasoning in Practical Dialogue Systems*.

Leifler, O. (2008). Combining Technical and Human-Centered Strategies for Decision Support in Command and Control — The ComPlan Approach. In *Proceedings of the 5th International Conference on Information Systems for Crisis Response and Management*.

Leifler, O. and Eriksson, H. (2009). Domain-specific knowledge management in a semantic desktop. In *Proceedings of I-KNOW '09, The International Conference on Knowledge Management*.

Lodhi, H., Saunders, C., Shawe-Taylor, J., Cristianini, N., and Watkins, C. J. C. H. (2002). Text classification using string kernels. *Journal of Machine Learning Research*, 2:419–444.

McDowell, L. K. and Cafarella, M. (2006). Ontology-driven information extraction with ontosyphon. In *Proceedings of the 5th Internation Semantic Web Conference*.

Medina-Mora, R., Winograd, T., Flores, R., and Flores, F. (1992). The action workflow approach to workflow management technology. In *CSCW '92: Proceedings of the 1992 ACM conference on Computer-supported cooperative work*, pages 281–288, New York, NY, USA. ACM.

Quinlan, R. (1993). *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers, San Mateo, CA.

Sahami, M., Dumais, S., Heckerman, D., and Horvitz, E. (1998). A bayesian approach to filtering junk e-mail. In *Proceedings of the AAAI-98 Workshop on Learning for Text Categorization*.

Salas, E., Cooke, N. J., and Rosen, M. A. (2008). On teams, teamwork, and team performance: Discoveries and developments. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 50(3):540–547.

Sanderson, P. and Fisher, C. (1994). Exploratory sequential data analysis: Foundations. *Human-Computer Interaction*, 9:251–317.

Sanderson, P., Scott, J., Johnston, T., Mainzer, J., Watanabe, L., and James, J. (1994). MacSHAPA and the enterprise of exploratory sequential data analysis (ESDA). *International Journal of Human-Computer Studies*, 41(5):633–681.

Scerri, S., Handschuh, S., and Decker, S. (2008). Semantic email as a communication medium for the social semantic desktop. In *The Semantic Web: Research and Applications*. Springer Berlin/Heidelberg.

Seewald, A. K. (2003). Towards a theoretical framework for ensemble classification. In *Proceedings of the 18th International Joint Conference on Artificial Intelligence*.

Silverman, D. (2006). *Interpreting qualitative data: Methods for analyzing talk, text, and interaction*. SAGE Publications Ltd.

Thorstensson, M., Axelsson, M., Morin, M., and Jenvald, J. (2001). Monitoring and analysis of command post communication in rescue operations. *Safety Science*, 39:51–60.

Trnka, J. and Jenvald, J. (2006). Role-playing exercise – a real-time approach to study collaborative command and control. *The International Journal of Intelligent Control and Systems*, 11(4):218–228.

Trnka, J., Johansson, B., and Granlund, R. (2006). Information support in collaborative command and control work – empirical research using a role-playing exercise approach. In *Proceedings of the 11th International Command and Control Research and Technology Symposium (ICCRTS)*.

van der Aalst, W., van Dongen, B., Herbst, J., Maruster, L., Schimm, G., and Weijters, A. (2003). Workflow mining: A survey of issues and approaches. *Data and Knowledge Engineering*, 42(2):237–267.

van der Aalst, W. and van Hee, K. M. (2002). *Workflow management: models, methods, and systems*. MIT Press, Cambridge, MA, USA.

Weick, K. E. (1995). *Sensemaking in organizations*. SAGE Publications Ltd.

Wen, L., Wang, J., and van der Aalst, W. M. P. (2009). A novel approach for process mining based on event types. *Journal of Intelligent Information Systems*, 32:163–190.

Witten, I. H. and Frank, E. (2005). *Data mining : Practical Machine Learning Tools & Techniques, Second Edition*. Morgan Kaufmann Series in Data Management Systems. Morgan Kaufmann Publishers Inc.

PAPER V

Ola Leifler and Henrik Eriksson. Communication analysis tools in command and control - open-endedness and transparency as design criteria. Submitted for publication, 2010

**Cognition, Technology, & Work manuscript No.**
(will be inserted by the editor)

# Communication Analysis Tools for Command and Control Researchers

## Open-Endedness and Transparency as Design Criteria

**Ola Leifler · Henrik Eriksson**

September 23, 2010

**Abstract** Our current understanding of command and control ($C^2$) team behavior and performance comes in part from studies in which researchers gather and process information about the communications and actions of teams. In many cases, the data sets available for analysis are large, unwieldy and require methods for exploratory and dynamic management of data. To support $C^2$ research concerning the behavior of command teams, we present the results from a study in which we investigate how to successfully deploy systems for text-based pattern extraction from $C^2$ scenarios.

First, we interviewed $C^2$ researchers to gain an understanding of their workflow when studying $C^2$ teams. Based on design criteria elicited from the interviews, we constructed an analysis tool for $C^2$ researchers who study electronic textual communication. The analysis tool used text clustering as an underlying pattern extraction technique. We evaluated the tool together with $C^2$ researchers in a workshop to establish whether our design criteria were valid and the approach taken with the analysis tool was sound.

The design criteria (open-endedness and transparency) elicited from our interviews with researchers were highly consistent with the results from the workshop. Specifically, evaluation results indicate that successful deployment of advanced analysis tools require that tools can treat multiple data sources and offer rich opportunities for manipulation and interaction (open-endedness) and careful design of visual presentations and explanations of the techniques used (transparency). Finally, the results point to the high relevance and promise of using text clustering as a support for analysis of $C^2$ data.

**Keywords** command and control, communication analysis, text clustering, exploratory sequential data analysis

O. Leifler · H. Eriksson
Dept. of Computer and Information Science
Linköping University
SE-581 83 Linköping, Sweden
Tel. +46 13 281000
E-mail: {olale,her}@ida.liu.se

### 1 Introduction

Understanding and assessing the performance of command and control ($C^2$) teams is central to $C^2$ research. For training purposes, it is critical to understand how command teams work in order to help them improve their performance in crisis management. For researchers in decision making and team cognition, the context of $C^2$ offers fertile ground for studying the fundamental processes involved, but also technical challenges when doing so.

The technical challenges when studying $C^2$ concern setting up the proper instrumentation and subsequently collecting sufficient data to enable the study of the selected aspect of the command team (Andriole, 1989). To study the psycho-social aspects of team work, for example, researchers need detailed logs of what team members say, how they interact with one another and an assessment of their backgrounds. To study the workflow of a group of commanders, researchers need to gather information about both the patterns in electronic communications of the group, how computer systems are used as well as how team members interact with one another in person. Due to the amounts of data generated in these settings, researchers need flexible methods for studying data sets that can accomodate different approaches to visual representation and reasoning.

Researchers study $C^2$ with different goals and perspectives and different methods. Common to most studies of $C^2$, however, is the issue of how to efficiently extract useful information from large assemblages of research data. For designers of support tools for $C^2$ analysis, this presents a great challenge. The objective of this study is to provide a better understanding of the conditions for successfully deploying automated pattern extraction systems in $C^2$ research.

### 1.1 Research method

The study in this paper has been conducted in three parts in which we have interviewed researchers, elicited critical issues in their workflow to justify the design of a support tool, and evaluated the resulting tool in a workshop.

1. In the first part of our study, we conducted a series of semi-structured interviews with command and control researchers regarding their work process and the tools they use for support. We also investigated data from three different exercises they had been involved in and the current tools used to process these data. The researchers were from different backgrounds but were all involved in $C^2$ studies to some degree. The results of the first part were of a set of critical issues in research that could be related to the use (current or future) of support tools for selecting important features of $C^2$ data to study.
2. The second part consisted of creating a specific support tool for the task of selecting information by using automatic pattern extraction techniques. Most communication used in the study of command and control comes as text directly or through transcription of speech. Thus, using text-based classification of messages was considered an viable option for filtering the datasets used in analysis. Based on earlier work on the feasibility of using automatic pattern extraction in texts for supporting data exploration (Rosell and Velupillai, 2008; Leifler and Eriksson, 2010), we designed a prototype tool to create patterns from communications.
3. The third part consisted of a workshop during which we presented a working prototype of our text clustering-based support tool to participants in the study and had discussions concerning the affordances of such a tool in their work. This discussion resulted in conclusions regarding the conditions for successfully deploying pattern extraction techniques in $C^2$ exploration tools.

### 1.2 Outline

The remainder of this paper is organized as follows: Section 2 provides a background to $C^2$, data analysis and tools for generating and verifying hypotheses about $C^2$ team performance. In Section 3, we present the results of interviewing researchers about their work process during analysis. Section 4 describes the design of the *Workflow Visualizer* for exploring relationships in communications and similar data sets. Section 5 presents the results from a workshop where the *Workflow Visualizer* was evaluated with respect to the possible applications of the tool and the technique it embodies, Section 6 puts the design and the results of the evaluation in context and Section 7 concludes this paper with conditions for successful deployment of automated clustering.

## 2 Background

$C^2$ is used as a term for describing what people commanding others do: directing the work of subordinate units and coordinating their efforts toward a common goal (command), making sure that orders are carried out and monitoring the outcome of all actions (control).

### 2.1 Command and Control

Until fifteen years ago, the term $C^2$ was almost synonymous to problem solving and decision making; that is, identifying clearly defined problems, determining possible options for how to act and evaluating those options according to established criteria (see e.g. Lawson Jr., 1981). During the last decade, however, researchers in $C^2$ have come to recognize new ways of describing this process and the situated characteristics of reasoning about concepts such as problems, options and criteria, which has been stimulated by visions of the affordances of new technology (Alberts et al., 2000) and new results on what it means when people command others (Klein, 1998; Schmitt and Klein, 1999). In contrast to the view of $C^2$ as a highly structured process of identifying problems, generating and evaluating options for solving them according to rational methods, Ross et al. (2004) presented an alternative model of decision making. In his model, commanders rarely set up and evaluate multiple plan options as mandated by military doctrine, but are more involved with building an understanding of a complete situation (not just a problem), communicating this understanding to others to form an intent and monitoring their environment for relevant changes (Jensen, 2007). At a very abstract level, the joint process of $C^2$ has been described as consisting of *making sense of a situation* and *communicating the commander's intent* (Shattuck and Woods, 2000).

### 2.2 $C^2$ Research

These new appreciations of what characterizes $C^2$ has resulted in an increased interest in methods for studying what it means to successfully make sense of the environment and communicate intent. To measure how well commanders perform their key functions, and consequently measure the effect of $C^2$, researchers have adopted methods from cognitive psychology for training staff and reasoning about staff work.

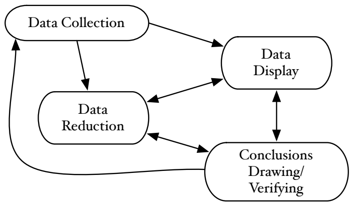


**Fig. 1** Four stages in qualitative data analysis.

As a compromise between field studies and laboratory experiments, researchers have begun to use role-playing simulations. In role-playing simulations, the roles and responsibilities of those studied are similar to naturalistic settings which facilitate the study of group dynamics between staff (Rubel, 2001).

As part of a role-playing simulation, a group of staff members assemble to form a team assigned with a task that they are likely to encounter in their profession. When operating in a role-playing simulation environment, participants play a scenario with much the same tools they would be expected to use in real situations which means there are many data sources from the simulated environment available for analysis afterwards. However, important parts of a team's work may be performed as part of discussions between members of staff, and thus, aspects of their work may not be well articulated in terms of interactions with the computer systems at all. To capture these important aspects of staff work, researchers typically complement simulation logs with human observers that periodically give accounts of what the staff are doing and evaluate their performance with respect to predefined categories (Thorstensson et al., 2001; Jenvald and Eriksson, 2009). Different human observers tend to tag events differently, however, reducing the reliability of human observations. After the exercise has concluded, all scenario data are collected and made available through one or several tools for exploratory sequential data analysis (Sanderson and Fisher, 1994). When conducting such analyses, researchers devise indicators of team performance given the outcome of the scenario at hand. These indicators are used the next time there is a similar exercise to focus the goals of the exercise and direct the data collection. This process leads to successively improved understanding of team performance, methods for analysis, and tool use.

## 2.3 Data analysis methods

The process of analyzing data after an exercise can be characterized in several ways. One way is to describe it is by using Miles' description of the four stages of qualitative data analysis (Figure 1, adapted from Miles and Huberman (1994)). The description presents four stages and the interplay between them. How data is selected affects the display of data, and the conclusions drawn guide further explorations (reduction/collection/display). In the case of exploratory sequential data analysis, Sanderson and Fisher (1994) described the method of data collection and analysis during a single experiment as one in which the researcher is engaged in either the *Observe* sequence, setting up data collection instruments and gathering information from them, or the *Analyze* sequence, in which transformed products (transcribed speech, annotated events) guide the search for patterns and answers that form conclusions in the work. Figure 2 shows how these two processes are guided by a set of formal concepts or questions.

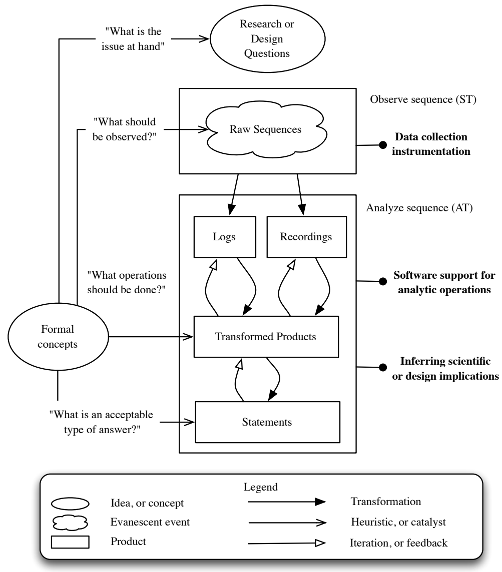


**Fig. 2** Exploratory sequential data analysis according to Sanderson. Adapted from (Sanderson and Fisher, 1994).

Another way of characterizing hypothesis generation comes from the community of data mining, where Cutting has characterized the process of formulating hypotheses about possible relationships in datasets as a process of iterating two activities: scattering and gathering (Cutting et al., 1992). In general, this process defines *scattering* as the act of creating a set of distinct objects of study by using some metric for comparison of objects, and *gathering* as the act of treating some objects as similar according to some criteria. This process may be iterated so that objects considered similar may again be scattered according to some new metric and so on.

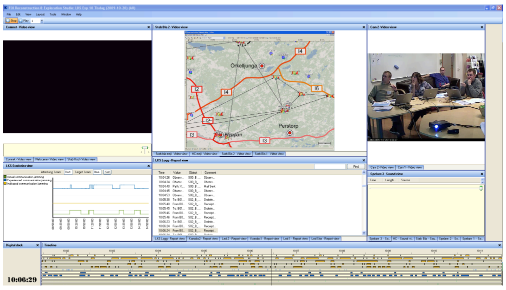

**Fig. 3** The F-REX Scenario Reconstruction Toolkit used for analysis of $C^2$ scenarios.

In much the same terms as Cutting, one recent text-mining approach by Rosell and Velupillai (2008) explains the process of mining for patterns in text as:

1. cluster the text
2. identify interesting clusters
3. explore cluster contents
4. formulate potential hypotheses

The four steps described by Rosell and Velupillai (2008) encapsulate the scatter/gather paradigm well, but could also be reconciled with the data analysis description of Miles and Huberman (1994) for suitable interpretations of what it means to identify and explore. Also, these descriptions could possibly be part of the iterated work process between *Logs* and *Recordings* and *Transformed Products*. When analyzing audio logs, researchers transcribe speech into text, this text may be annotated according to a schema with categories that are decided to be interesting, which in turn may cluster data as a set of episodes, which in turn direct further analysis by making video logs or observer reports at specific points in time relevant to study.

### 2.4 Data analysis tools

In qualitative content analysis, researchers seek to categorize data from interviews and other sources in a common framework. The framework can either be taken from previous literature from the research field the study concerns (a priori coding), or be developed as part of the analysis (emergent coding) (Lazar et al., 2010). Especially when an emergent coding is needed for analyzing data, it can be very labor-intensive. To support the coding and analysis of communication data from command teams, exploratory sequential data analysis (ESDA) is often used as a method for understanding patterns in the sequence of messages exchanged between the members of a group.

In ESDA the interplay between collection and analysis of data is central and researchers collect, merge and view data sources by using ESDA tools such as the F-REX exploration tool in Figure 3 (Thorstensson et al., 2001) or MacSHAPA (Sanderson et al., 1994). In exploration tools for ESDA, all data sources are available as a series of events along a scenario timeline. Figure 3 shows a screenshot where screen-captured video, radio communications, text messages and other data sources are available through a graphical interface that contains a scenario timeline at the bottom. A central aspect of the exploration phase during the analysis of a command scenario is communication analysis. With multiple actors involved in a scenario and several parallel courses of events unfolding, it is important that the sequence of communication events can be managed efficiently. It has proved to be especially useful to have tools for annotating, searching and visualizing the flow of information (Albinsson et al., 2004; Morin and Albinsson, 2005).

The use of reconstruction and exploration tools have opened up new possibilities for researchers in formulating hypotheses about team performance since the amounts of data that can be treated has increased greatly, but this has accentuated the problem of data reduction or, by using the terminology of Sanderson (Figure 2), the interplay between sequences of data and the transformed products one can create from them. To facilitate this process of reducing data sources to manageable and comprehensible chunks, researchers have devised tools for visual exploration of patterns (Albinsson and Morin, 2002) to find critical incidents by using explicitly available attributes of communications to elicit patterns.

### 2.5 Pattern extraction

Techniques for extracting patterns from data collected during a $C^2$ exercise require an intimate understanding of the requirements for using them, the possible outcomes and how to interpret the models constructed. When confronted with the data sources most often available from a role-playing simulation, however, the process of validating data sources and making sure that they are valid as a basis for statistical analyses can be a serious impediment. Some data sources may be textual notes from observations made by human observers that have categorized their observations. Such manual categorizations may differ among observers and thus have poor reliability as a basis for statistical analyses. Other condensed metrics such as communication density are unreliable and incomprehensible predictors of team performance in real situations (Gorman et al., 2003).

There are other methods for using pattern extraction techniques though, that do not involve making automatic predictions of team performance. Many techniques for building patterns in data are in fact not used to completely automate the process of building a team performance model,

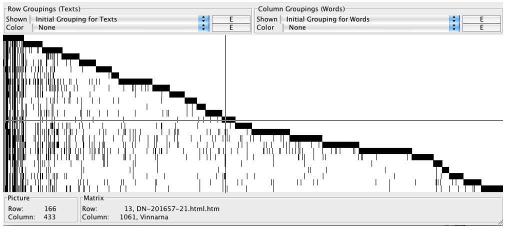


**Fig. 4** Infomat, an Information Visualization GUI by Rosell and Velupillai (2008)

**Table 1** The roles of people interviewed regarding communication analysis in $C^2$.

| Person | Role |
|---|---|
| Jane | Communications analyst |
| Charlotte | Communications analyst |
| John | Communications analyst, research project leader |
| James | Researcher |
| Charlie | Senior researcher, $C^2$ training expert |
| Sebastian | Communications analyst |
| H-G | Tools specialist |
| Freddy | Tools specialist |

but instead assist researchers in their work through methods for reducing and displaying the dataset.

For example, text clustering can be performed to relate texts to one another based on distance metrics. One technique for text clustering is Random Indexing, which is a vector-based clustering method that can be used to cluster texts and extract patterns in large data sets (Kanerva et al., 2000) such as communication data between members of staff during a simulation. Random Indexing has been demonstrated as a basis for a tool for presenting visual and direct representation of the relations between terms and texts in the Infomat information exploration tool by Rosell and Velupillai (2008). Figure 4 presents the Infomat tool in which users have a matrix-based view of a dataset where they have direct control when exploring possible patterns in the dataset: clusters, important terms and word cooccurrences.

## 3 Interviews with communication analysts

To establish how communication analysts perform their work and how they use their tools to support their analyses, we performed a series of 8 interviews with people involved in $C^2$ analysis. The participants had experience from conducting research on $C^2$ and training staff. They were interviewed to establish how they perform data analysis with a particular focus on how they find patterns in communication data. All interviews were semi-structured and conducted using critical-incident interview techniques (Flanagan, 1954) where we probed for situations in which the participants had engaged in critical and specific activities typical to the analysis of large data sets.

The participants were chosen based on their expertise in the analysis of communications and development of technical systems for the support of such analyses. Their work was studied through interviews because the work they carry out is infrequent and distributed over longer periods of time which, on a practical level, makes it difficult to investigate the context of their work over a given period in a more situated manner. Semi-structured interviews were therefore considered a valuable tool for extracting information about their methods when analyzing communication and their relationships to tools used for such analyses.

The people interviewed are listed in Table 1, where their names have been anonymized. Each interview lasted approximately 60 minutes and followed a script in which three main themes were discussed:

- *What is the purpose of conducting analysis of team communications?* The participants were given a chance to elaborate on the purpose and nature of exercises or experiments they had been involved in and how those purposes directed the analysis of scenario data.
- *How exactly is communication analysis performed?* The participants were asked to answer this question by relating to their own personal experience from one or more scenarios in which they had conducted analysis. Depending on the role of each participant, they had been involved in various parts of communication analysis or possibly the construction of after-action review support tools.
- *What are the most time-consuming, challenging or precarious stages of communication analysis?* In describing their work with analyzing data, all participants were asked to elaborate specifically on challenges with communication analysis, both with respect to tool usage but also with respect to other factors.

### 3.1 Interview analysis

All interviews were recorded and transcribed. The interviews were in Swedish and the quotes that follow have therefore been translated to English. They were divided in dialogues for clarity and annotated using the main categories of *purpose*, *method*, *tools* and *challenges*, which related to the themes of the questions during the interviews. The annotated materials were later tagged with sub-categories for each category. Later, the sub-categories with special relevance for the interplay between tools and methodological issues were selected for further analysis. We selected the critical incidents mentioned by the participants in which these sub-categories appeared and grouped these together as three *issues of special concern* related to the use of technical systems in research.

The first issue concerned the activity of *focusing* the research question, restricting the overall research question to a specific question or selecting a subset of the available research material for further study. The second issue concerned how researchers *draw conclusions* from data and how different representations of the material help them in this work, and how the representations and tools they use affect conclusions they draw. The third issue concerned their *understanding of limitations* in tools and data used.

### 3.2 Focusing

The participants stated that the goals in their studies were concerned with either understanding how people work when they solve a particular task or evaluate their performance. One of the participants, Jane, described a research project on the behavior of a command team in the absence of specific competencies. Her task was primarily to study the general behavior of the team under the specific conditions of the study. However, *the amount of data* recorded soon restricted what could be studied.

> The reason why I chose not to [study other aspects] was that there was such a huge amount of material. [. . . ] it was an issue of time as well. *(Jane, line 565–567)*

The type of data to be recorded during the exercise studied by Jane was known in advance, but it was only during the initial analysis that they saw how much data had been collected and decided to restrict what was to be studied. They focused the study by removing all communication trails that did not have to do with two specific staff members who were assigned duties they were not trained for, with no regard to what those communication trails could include. Sebastian described a similar situation in which he was tasked with analyzing pilot communications from a fighter pilot training session. As in Jane's case, the amount of data made it necessary to reduce the scope of the analysis to narrow time frames prior to when weapons were launched:

> sometimes [the course of events] go quite fast in these situations and sometimes things are rather slow [. . . ] so then we chose a minute before the shot as a good compromise. *(Sebastian, line 371–375)*

The one minute time frame was chosen by Sebastian and his supervisors based on experience, though he could not remember any more details regarding the rationale, why one minute before would be better than two minutes, or one minute before and one after. The reason why a portion of the communication logs would be relevant for analysis was because of a hypothesized relation between the communication, the situation awareness in the team and the joint performance of the team, which was assessed by both performance metrics in the simulation as well as subjective evaluations by a senior officer.

> [The participants] have to be synchronized and you have to coordinate every little step like this and be sure that others have understood. I mean, before you move on with the next part of the procedure. *(Sebastian, line 408–413)*

All efforts at focusing a study by selecting subsets of the communication data to analyze were conducted with the intention that the data studied should be an optimal subset for correlating team performance with communications. In Charlie's interview, he explained that when distilling a set of reports into a few key observations to highlight during team debriefing after training, he narrowed down the set of available observations by including only those that had enough reflective remarks in them and selected those reflections that matched his own subjective evaluations. So, the agreement between his own impressions and the data collected by others was vital for his conclusions.

Focusing the communications analysis effort towards a particular question was described in the interviews as a process guided either by experts who had their own hypotheses regarding the parts of team communications that were relevant, or as a result of manual work transcribing communications. In focusing their work, the participants used mostly spreadsheets for collecting communication trails and categorizing communications according to a number of categories. They mentioned the use of ESDA tools mostly in the context of clarifying what someone was talking about in cases where a spoken conversation related to objects visible on a computer screen or on a table. When asked if there were specific theories guiding the hypothesis generation, the participants answered by relating to doctrine (Jane), personal evaluations and experience (Charlie) or subject expert evaluations (Sebastian). The tool support available enabled them to create tables with communications and classify those communications according to categories, but there was no automatic support for extracting statistical patterns from communication features such as message length, direction of messages or co-related terms.

### 3.3 Drawing conclusions

The second issue of special concern during the interviews was how they drew conclusions from the data they had selected for analysis. Jane explained specifically how and when she could draw conclusions regarding the effect of competence loss on staff performance. She had manually annotated episodes (conversations, threads) in the communication flow when the staff members talked about a specific topic which they had no prior experience in dealing with and then began to search for how they had managed that lack of competence.

> [My assessment of their performance] probably started with these somewhat obvious errors [. . . ] when he informed [his colleagues] incorrectly given the directions from the scenario management team. Then I could see more things that had gone wrong. [How I reached my conclusions] is difficult to say, it was a process and difficult to remember exactly. *(Jane, line 732–737)*

Another participant noted that the process of understanding the communication structure of a team became obvious given annotated communication and the amount of communication sent in the different annotation categories. No further help was needed to understand the structure of a team's communication given figures extracted from the amount of messages in each of the categories used.

The participants described the process of drawing conclusions as one in which researchers narratively describe what has happened in a scenario which has transpired: who communicated with whom, what actions were taken and so forth. A generalization of such a narrative may generate a model of people's responsibilities, key communicators, pivotal events and typical responses to those events, maybe in the form of a team workflow. Relating a model of team communications to team performance was described as rather difficult. A descriptive model can be written in many different ways, depending on the aspects the researcher is interested in describing. A prescriptive model, that is, one directly related to team performance, needs to model the aspects of teamwork that most concretely influence some type of performance. Establishing a metric for performance is maybe the most challenging task in creating a model for prescribing team performance.

However, all participants also noted that assessing performance in a joint team is very challenging due to the difficulty in describing the nature of what a team does, and that performance assessments therefore tend to be concerned with metrics that are constructed to be simple to measure, and that those measurements can be used together with others to triangulate some understanding of the concept of performance. In one case, John described how they studied the effects of command styles on communication and performance:

> We have looked at the frequency of direct orders [in this simulated environment] versus communication of intent [and] by looking at that you get an idea of what kind of style the commanders in this exercise have. If you could do statistical analyses and establish a connection between a particular style and performance that would be very exciting. We have not been able to do that. *(John, line 69–78)*

Correlating that which can be analyzed to the more elusive concept of team performance is challenging. Written communication is an accessible form for analysis, in contrast to video and other media. It is therefore natural that the researchers look for patterns in such data to guide their analyses. However, because performance measures are difficult to specify clearly enough to be measured unambiguously, researchers do one of two things. Either, they measure success by a proxy variable (communication style), with an hypothesized but unverified connection between the proxy variable and the outcome of the scenario. Alternatively, performance is defined subjectively by experts, which can lead to difficulties if the reasoning conducted by the expert is not well understood. In a study Sebastian participated in, experts helped him construct a classification scheme which would identify problems believed to be associated with low team performance. The subject experts were also involved in establishing the performance gradings for the teams.

> There is a positive correlation between the number of [communication] problems per minute and grading, that is, the more problems the higher grade. There is another [communication] category which is higher for the best team and that is "unclear information". *(Sebastian, lines 1179–1181, 1188–1189)*

The relationship between communication issues and scenario outcome could not be determined as the expert had suggested. During the interview, Sebastian discussed the possibility that increased communication of problems could indicate a willingness to discuss issues instead of avoiding them, a willingness which might be positive to the joint situation awareness of the team.

The descriptions all interview participants gave of the process of drawing conclusions from data centered on the representations used. John noted that the tabular representations of *messages and categorizations of messages* directly led to conclusions regarding communication style. Jane could not pinpoint exactly when she could draw conclusions regarding how the team had managed performances, but she reasoned in terms of how the communication already selected for analysis was *color-coded in episodes* (dialogues, communication threads) and how those episodes formed a direct basis for drawing conclusions. All representations cited in the interviews were the direct result of using spreadsheets and some paper calculations. In the process of drawing conclusions from data, existing ESDA tools were only cited as useful in training, where Charlie explained how they used their ESDA tool similar to F-REX for after-action debriefing of staffs they trained in emergency management.

The visual, integrated presentation was the central aspect of using an ESDA tool in their case and a source for reasoning about joint staff behavior.

### 3.4 Understanding tools and data

Several researchers had used F-REX or similar ESDA tools for data exploration that allowed multiple data sources to be shown simultaneously. ESDA tools can offer significant advantages for analysts when they search for key events in a scenario with many data sources, but several of those interviewed described that the tools are difficult to use for communicating results.

> Even if you as an analyst [understand the tool], it is difficult to demonstrate [your results] in a good way to someone else without access to the tool itself. *(John, line 181–183)*

Having data available as tables in a spreadsheet was considered much easier when communicating results to others. Especially for categorizing messages, sorting, selecting them and presenting simple summations of results, spreadsheets fill many roles in research.

> You work a lot with software that makes it easier to sort and mark things, so it has become a lot of Excel and then I can have a column next to [the messages] where I enter their category codes. [. . . ] When it comes to visualization you often use Excel because then I can create my tables right there and show them. *(John, line 494–496)*

The visualization and direct representation of synthesized results inspired one of the researchers involved in creating support tools to develop an annotation component to be used when reasoning about key events:

> You are looking at some kind of data source in a window and wonder how it is related to, ah that, and then you have a map there [. . . ] Then you want to like save this, just as it is right so that the next person doesn't have to do that all over again. *(H-G, line 730–747)*

The most important use of the annotation component was to communicate important events to other researchers. Several participants described the process of communicating results through the tools they used to manage data sources. The difficulty when using tools like F-REX to communicate results could be caused by the fact that using audio and video sources are difficult in themselves and could be the real reason why people resort to formats that are easier to analyze such as text. John described how they used mostly text because of the amounts of audio and video generated during multi-day scenarios. Those amounts could simply not be managed within the timeframes commonly available for analysis. Jane described how she had only selected a small subset of the scenario episodes related to lack of competence in a team out of all the telephone logs for transcription. She decided on a certain subset of episodes to study before she transcribed any audio simply because of the time required to analyze all data.

Understanding the limitations of the data available and the tools was considered critical by all participants, and especially two areas of concern were highlighted: the reliability of human observers and the transparency of statistical modeling tools. Regarding inter-observer reliability, that is, the degree to which independent observers describe the same course of events using the same categories, John went as far as suggesting that a computer system for annotating events that was at least consistent between multiple scenarios would be preferable to human observers. It would not have to annotate using the same categories as a human, but if it could at least behave in a consistent manner, that would make analyses possible, as opposed to when annotations could not be used due to the differences in how people evaluate the same situation in the same scenario.

When using statistical tools, Charlotte described how she used LISREL (Jöreskog, 1973) modeling which can reveal several statistically valid equation models with the variables measured. Only some of the models constructed would actually be contextually reasonable though, which made the work of interpreting them difficult without knowledge of the work context, the mathematical properties of the underlying variables and the distribution of possible solutions. However, the ability to explore several possible relations in data was considered very valuable in her research.

### 3.5 Summary of interviews

Taken together, the interviews illustrated two observations of tool use that we considered relevant for the construction of support systems for communication analysis:

*Open-endedness* Several research projects described during the interviews started with open questions on how to characterize teamwork, irrespective of performance metrics or the relation between team performance and manifested behavior. Therefore, tools to support analysis of communications must not make or require any specific assumptions about team communications. The interview participants mentioned that an automated approach for annotating communications would possibly be useful even if it did not use the same tags for annotation as a human observer. Also, they noted that the exploration of possible patterns in data, both when using spreadsheets and statistical modeling tools, was very useful for their understanding. The utility of the tools they used was described not so much in the level of automation provided but the freedom to choose how to operate on data.

*Transparency* Many of the participants described that they used tools for analysis that allowed a direct and visible relationship between synthesis and data. H-G had constructed a component for one of the ESDA tools used to make the association between reasoning and data transparent by adding annotations directly to the timeline of events, and one of the main arguments for the use of simpler tools such as Excel was that the connection between data and statistics was much easier to make. The use of specialized tools was described as dependent on the ability to use the tool for communicating results, and the primary risk with specialized tools was considered to be the risk of not being able to show the insights gained through them to other researchers or clients.

These two observations relate to our earlier observations regarding criteria for decision support systems in $C^2$, where we identified *graceful regulation* (allowing different uses of a tool in open-ended scenarios) and *transparency* as central conditions for success (Leifler, 2008) for tools that assist in military planning. Some participants described that using special purpose systems such as F-REX was problematic. In their descriptions of why, one could attribute the difficulties to the task performed by explorative multimedia management tools (managing large, heterogeneous data sources), a discrepancy between researchers' work and the specific requirements of the tool or the fact that any special-purpose program requires too much dedicated work to be used frequently enough.

To understand how these observations could be further concretized to guide the design of support tools, we developed a prototype system for navigating and finding patterns in data. All three main areas of concern elicited in the interviews revealed that the data sets were not reduced or classified using automated methods for pattern generation. Although the interview participants indicated that great care must be taken when implementing special-purpose systems that implement advanced analysis techniques, clustering systems have been deployed successfully in other settings (see Section 2.5). We therefore chose to implement a system for navigating messages that would contain several options for representing, navigating among and selecting sets of messages, in part based on the Random Indexing clustering technique.

## 4 A Prototype Communication Analysis Tool

In our design of the support tool, we first recognized that researchers spend much of their effort on narrowing research questions and looking for patterns in data (Section 2.4) with little use from automatic tools for pattern extraction such as text clustering techniques (Section 2.5). Following the interview series, we interpreted the implications of the resulting two observations in the following way with respect to the use of text clustering:

*Open-endedness* implies the ability to choose how to use computer-based models of a scenario depending on trust in the model and user needs. In our design, text clustering can be used for two distinct purposes, or not at all. Users can choose to inspect parts of the communication flow based on key terms occurring in the messages, or inspect clusters of messages based on their proximity to each other according to the clustering model. The former method requires users to only rely on the computer model for selecting the most relevant terms to select messages by, whereas the latter requires users to trust the vector-based model to produce contextually significant clusters of messages. The option not to use clustering at all (but still use the tool) means that users can select messages through directly available attributes and metadata instead, in a selection component where all explicitly available attributes of messages, such as the participants in the communication, the timeframe of the communication, are represented graphically.

*Transparency* implies that computer-generated models of data sets, such as vector-based models of the texts in a communication flow, should be made directly accessible to users by exposing the defining features through a graphical interface. If possible, the process used to create the computer-based representation should also be directly comprehensible. Transparency depends much on the conceptions users have of the underlying techniques that the computer uses. In our design, we hypothesized that *making key terms* extracted by the vector-space model into *part of the interface* for selecting groups of messages would help make the clustering process more transparent.

The intended use of the tool, which we named the *Workflow Visualizer*, was to support researchers in exploring relations and draw better conclusions from $C^2$ research data. Two use cases had been elicited from the interviews and discussions with researchers, and they were based on datasets from authentic $C^2$ research scenarios.

### 4.1 Use Cases

The two use cases came from $C^2$ scenarios that the interview participants were familiar with but where they had not used their own tools for eliciting patterns in data. The first scenario concerned performance analysis of staff engaged in information warfare and specifically their reactions to radio interferences in a scenario. The use case that builds on this scenario consisted of how to search for patterns in data with respect to specific events and extract text messages based on prominent terms in the communication flow that were related to those events. The second scenarios concerned a series of ten $C^2$ exercises for the rescue services where analysts wished to explore differences in the communication and performance of the teams between the exercises. In the second scenario, we presented a use case with the Workflow Visualizer for finding patterns between exercises by automatically clustering data according to distance metrics imposed by the text clustering engine. Both scenarios and the accompanying data sets were provided by researchers who participated in our study.

### 4.2 Performance analysis

The first scenario concerned information warfare, in which a group of commanders were responsible for securing transportation of VIP's via helicopter in a hostile, fictive area. They coordinated their efforts with their higher command through e-mail and were monitored by human observers who took notes of their actions during the scenario. They also logged their own perceptions of threats in their environment during the course of the experiment.

The data sources from the first scenario (text messages, observer logs and simulation logs) were imported in the Workflow Visualizer using an appropriate import component. Each import component creates a data model of each scenario that is based on the concepts of *message* and *event*. A *message* contains participants, a timestamp, text, possibly a manual classification and other scenario-specific metadata. An *event* has a timestamp, a description and possibly other metadata. Based on these messages and events, the Workflow Visualizer enables after-action reviewers with means of sifting through the information through direct manipulations for selecting, visually presenting and clustering data. Also, the importer can be configured to consider certain parts of the scenario data to be static (in the information warfare case, radio interferences they should react to) by which to select messages.

Users can choose to select a subset of messages based on keywords or other attributes. In Figure 5, the user has made a selection for messages that contain the keyword "strning" (interference). The number of matched messages is indicated before the user chooses to populate the timeline view (*Create Process*). In the first scenario, the simulation logs have information about all actual interferences during the scenario. When analyzing team performance, the user of Workflow Visualizer wants to inspect the team's reactions to interferences and therefore selects all messages sent among the staff mentioning interference.

Much written text from exercises has no explicit context structure such as threads that are available in e-mail conversations. Therefore, it can be difficult to identify which messages relate to one another as responses to earlier questions for instance. When creating a visual overview of the

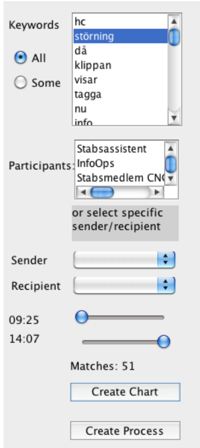


**Fig. 5** Selection of messages based on communication participants, the time frame for the communication and the keywords present in the messages.

communication between different participants in the exercise, we chose to implement a color scheme based on the hypothesized context of a message. Based on a selection of keywords in the keyword list, a set of messages is drawn as communication arrows (6). The color of each arrow depends on whether there has been any earlier conversation between the sender and recipient. If the last message received was from the intended recipient of the current message and contained the selected keyword(s), then the same color as that last received message is chosen. Otherwise, a color is chosen at random. This scheme was intended as guidance to testing whether a set of messages was to be considered a conversation on the same topic or not. All actions are not available as messages, however, which is why the user needs to triangulate messages with logs from human observers who monitor and log the team's behavior.

The observer reports available in the dataset from the first scenario consist of reports categorized according to a hierarchy of possible conditions for exerting command. When importing the observer reports, the import component provides these observer categories as a parameter on which to make selections. This parameter, along with all other parameters, is used to create a graphical selection component in the selection view. Figure 7 shows the selection components for observer categories that can filter reports depending on the category of the report. Each category in the scenario represents an enabling condition ("frutsttning") for $C^2$ that the report is concerned with. In the scenario, there are two different categories of reports related to interferences, labelled "communication interference/minor" and "communication interference/major". When selecting one of these in the tree,

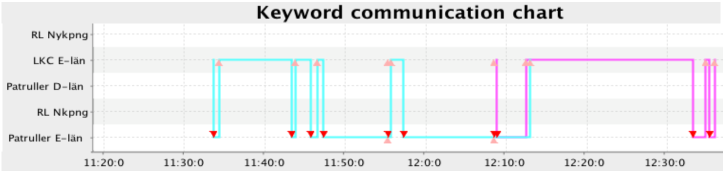


**Fig. 6** Hypothesized threads in the communication indicated by arrows of different colors.

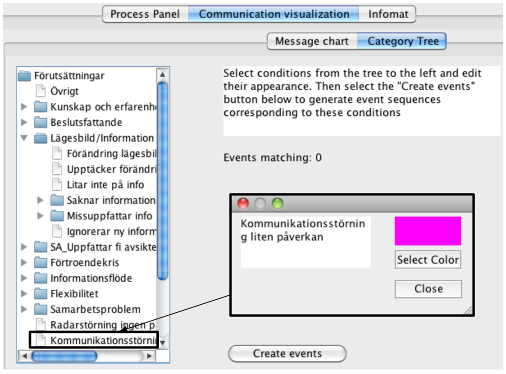


**Fig. 7** When users select one of the observation categories in the tree ("communication interference/minor" above), they can choose a color when plotting observations with that category along the timeline.

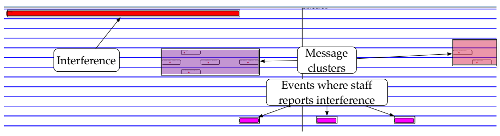


**Fig. 9** Clusters of messages and a set of observations regarding interference as displayed in the timeline view of the scenario at the time when there is simulated radio jamming.

the user gets to select a color that will be associated with the observer reports indicating interference. With a selection of messages and events that the researchers believe may capture situations during the scenario when the staff has reacted to interferences, the researcher can turn to the timeline view for better understanding how they have reacted to the interference.

In Figure 8, we see the timeline of events that is displayed when observers noted that the staff believed they were subjected to a major interference. At the top, two true interference episodes are listed and it is clear from this representation that the staff did not react in time on interferences they had been subjected to.

At another stage of the scenario (Figure 9), a selection of the messages sent by members of the staff indicate that they were consistently late at recognizing both the presence and absence of interferences. In the timeline, we have chosen to group messages in clusters according to when they were sent, and color them with a color gradient between two colors according to a metric (here, the same metric as used by the Random Indexing clustering). We can notice that the clusters appear with an offset in time from actual changes in interference. At the end of a period of interference, the first cluster indicates that the staff begins to talk about interference (as corroborated by human observations) and decide to act on the interference at about the time when the interference ceased. Their reaction to the absence of interference comes much later as indicated by the second cluster of messages.

### 4.3 Exploration

The second scenario contained data from 10 runs of minor command and control exercises (lasting approximately 4 hours each) with the rescue services, where the researchers are interested in finding whether any one of the exercises had been different from the others based on the contents of the messages between members of staff. In this scenario, we demonstrated the exploratory use of Random Indexing for clustering messages. Our approach with finding patterns in large text sets using Random Indexing was inspired by the Infomat information exploration tool by Rosell and Velupillai (2008) (see Section 2.5). Infomat was created to support exploration patterns of large text corpora through the use of vector-based representations of terms and texts. The Infomat representation of the texts and terms was a sparse visual matrix representation, with dots indicating occurrences of terms in a particular text. The Infomat is a powerful, multipurpose vector-space exploration tool, but unfortunately with a very steep learning curve for analyzing specific content such as communications and an interface that is very far from what is commonly used in exploration and analysis when the raw data sources themselves are displayed (see Figure 3).

Our adaptation of the Infomat interface is shown in Figure 10 where a set of message cluster is shown using the key terms occurring in each, where stop words have been filtered out but no stemming or other preprocessing has been conducted. The user can select a certain number of clusters to create, which causes a clustering algorithm to generate the

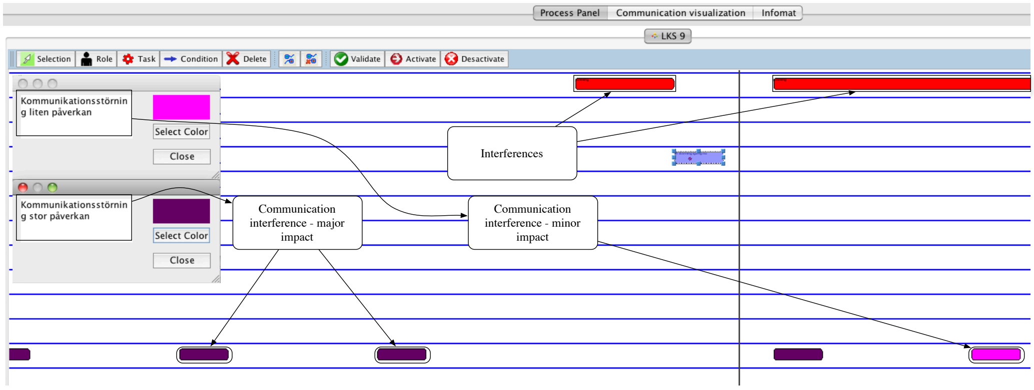


**Fig. 8** A set of message clusters and observations concerning the staff's beliefs about interferences.

best partitioning of the set of messages in the selected number of cluster. By selecting a color for a cluster, the user can differentiate clusters from each other when showing them in the scenario timeline. Also, the user can reason about their relevance to the scenario by inspecting the key terms deduced for each cluster of messages. A cluster that contains related terms the user is interested in can be selected for further inspection. For example, Cluster 5 in Figure 10 contains a reference to "incoming" ("inkommer"), "radioroom" ("radiorummet") and "plotting" ("plottar"), which are related terms that describe the process of managing new information arriving at the staff and entering the information on a common situation overview map. Cluster 2 is described by terms that denote calm ("lugn", "lugnt") periods. To explore the differences in perceived stress, the user hypothesizes that messages in cluster two are concerned with status reports regarding low workload. He chooses that cluster together with cluster one that is concerned with messages regarding a traffic accident ("trafikolycka").

Figure 11 displays the timeline of the second Sand scenario. Only a few messages from the selected clusters appear in the timeline. On the other hand, in the third scenario depicted in Figure 12, we see a larger number of messages. The higher frequency of messages could indicate better communication, increased workload or other differences between the scenarios, but by navigating through the timeline, and possibly by using other data sources such as video recordings, a researcher can probe the dataset and search for explanations. By selecting individual messages, he can see the message texts and additional information, and when moving along the timeline, there is a vertical time indicator moving with the cursor.

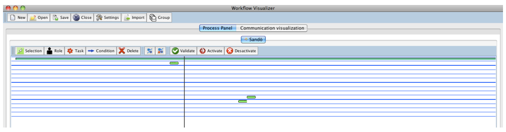

**Fig. 11** The two clusters of messages during scenario two of the Sand material.

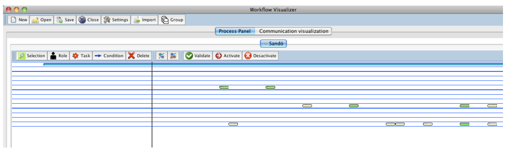

**Fig. 12** The same cluster of messages at scenario three in the Sand material. There is a clear difference compared to scenario two according to the clustering.

## 5 Prototype Evaluation Workshop

These two scenarios, highlighting the support for performance analysis and exploration, were used as the basis for a workshop during which we presented both scenarios to five participants: four from the original group of interview participants and one external communication analysis expert. The workshop was conducted over the course of half a day as a set of sessions in which the participants got to discuss our interpretations of the challenges in communication analysis as expressed in the scenarios presented, and the two use cases of the Workflow Visualizer tool in each scenario. Each session was conducted with a presentation of the scenario and use case followed by a five-minute individual reflection where each participant wrote down their own impressions on paper and then a 45 minute joint discussion.

Both scenarios were discussed according to four topics: whether the scenario or intended tool use seemed *ob-*

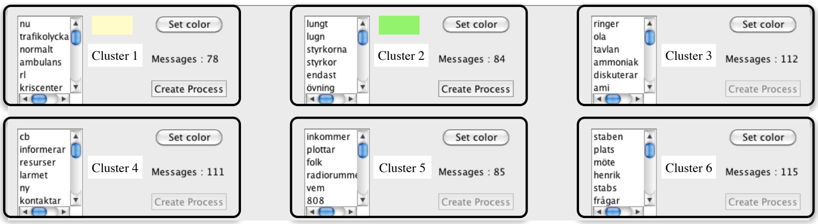


**Fig. 10** Representation of a set of message clusters, as hypothesized by the Random Indexing approach to cluster message texts.

*scure*, whether the task described was *similar* to scenarios that the participants had taken part in themselves, whether there were parts of the presentation of the tool that were *surprising or unfamiliar* and last, if they had *general remarks* on the described scenario and use of the Workflow Visualizer.

These four topics were chosen to provide options for the participants to organize both their critique and their reflections regarding current work practices (similar tasks) and the presented tool support. All participants were actively involved in discussing all four topics regarding both scenarios. They were directed to bring notes on all four topics to the joint discussion and share them with each other in turn.

### 5.1 Team performance scenario

The first use case concerned analyzing team performance when the performance metric was decided in advance as the time and character of the team's reactions to communication interferences. In the discussion of the proposed method for analyzing team performance using Workflow Visualizer, the participants stressed the importance of transparency, especially in the communication chart (Figure 6). They found that the most obscure part of the presentation concerned the chart and the basis for constructing it.

> These keyword-based threads are rather opaque for me as a communications expert so I get unnerved when a program does that for me.

The communication chart quickly became the focus of attention for the workshop participants when discussing this scenario. The graphical nature of the chart, along with an opaque reasoning for creating it, generated many questions regarding exactly how the threads were generated. The threading of sentences was originally intended for different use and was not presented as a central piece of the software for supporting communication exploration, but it immediately caught the attention of the participants. One of the participants, however, was not familiar with the material used in the scenario and the issues raised during the manual analysis of the material. He was therefore hesitant to comment:

> I feel that I have too little knowledge to say what it is exactly that is unclear. [ ... ] I can see a few screens but I have a hard time to get a feel for it.

Another participant noted that he felt the purpose of creating threads of communications in the first place was not clear to him, at least not in the scenario in which it was described:

> [The timeline view] seems much more important for the analysis you intended to do, whereas the previous [keyword-based communication chart], it seemed like a very good diagram [ ... ] but I cannot really see the point.

They continued to note that when specialized tools, that make specific assumptions about data in order to perform deeper analyses, are used in settings where conditions and data collection methods may change over time, researchers need to be careful. One participant gave an example from a series of experiments:

> An important aspect of the [X] dataset is that the method used may not be 100 percent logical. [ ... ] They have developed and changed [the method used in the trials] between trials which means that you mix many different kinds of data. [A specialized tool such as F-REX] becomes very sensitive to what you enter in it. If you are aware of what you are entering and are also aware of what you are looking at it may be very helpful and very good. If you have a vague understanding of the logic behind it, it can be very confusing.

Following an explanation that the keyword-based chart was to be considered as a hypothesis and support for exploration, the participants expressed positive views regarding the possibilities for using it for similar tasks that they were engaged in. One of the participants noted that, in particular:

> I think you could use this tool not just for communication analysis but for cause-and-effect analysis in general. I see reasonably large similarities with what we do in [tools such as F-REX]. You can see a thread in a course of events.

Also, when there are no prior categories to apply for categorizing textual data collected from a $C^2$ experiment, they thought it would be helpful to have access to a tool for efficient exploration:

> In our [microworld] study, we have huge quantities of data and we're not clear exactly what we are looking for. [ ... ] For the exploratory phase if we're using your terminology, I think this is most useful.

With understanding and transparency in focus, they went on to state that human analysts always have knowledge that can not be encoded in the tool and that the tool must allow close human control over how messages are grouped and how terms are treated:

> If I know that this word is really important, can I tag it [as such]? I think that would be a very important feature. I know that *Klippan* is [an important term], and when we evaluate team performance there are certain terms that we look for. That is central to evaluating the outcome of the scenario.

The participants recognized the specific features of the scenario in which the tool had been deployed and could reason concretely about how they would like to use clustering support.

### 5.2 Pattern exploration scenario

The second scenario, in which we demonstrated how to use clustering as a means to guide hypothesis generation and search for patterns in communication, sparked even more discussions. Here, the workshop participants engaged in discussions of what they would want the tool to do in the future and asked questions regarding how the work process would look like given the tool:

> can you use this to look for interesting sequences and identify points in time that you want to zoom in on, do a new clustering on and eventually arrive at something that is manageable for an analyst in the end?

Working with clusters as hypotheses for communication patterns was considered valuable for understanding team performance. The mathematical foundations of the clustering approach were considered difficult, though, and although the messages in each cluster could be plotted along a timeline and the key terms defining each cluster were given in the interface, they were hesitant to use a model that they did not have any prior understanding of. However, when discussing general remarks regarding clustering, the participants began constructing scenarios in which they would like to use the tool for exploratory purposes:

> It could say something about the development of a scenario–in the beginning, you talk a lot about "danger" or "risk" and as time goes by you talk more about [other issues]. You could extract a graph of that particular word or that [cluster in which the word is central] along a timeline.

Other participants filled in and discussed how the color of each cluster could be used to separate them according to keywords in them. They also gave examples in this use case of how the tool could be used with better defined user control to achieve a better workflow during the exploration phase:

> I would like to edit these clusterings and remove certain keywords that are not relevant and merge certain keywords that were essentially the same like calm, calmly.

## 6 Summary and Discussion

The interviewed researchers noted that the act of identifying interesting objects of study from the data sets they had collected was not straightforward, despite the use of structured tools and methods. In fact, they described several cases in which the decision to restrict a study to a certain subset of data was based on tacit knowledge. They rarely used advanced pattern exploration techniques as part of the workflow for selecting possible hypotheses regarding patterns in data.

Data exploration and analysis tool such as F-REX do not make use of automated techniques for pattern generation in the data sets they manage but primarily support users with a unified interface to several data sources. Although some attempts have been made to augment these tools with metadata and annotation capabilities on raw data sequences, these capabilities have not been extended to include automated reasoning about data.

The construction of the Workflow Visualizer support tool was guided by two observations. First, the interviews resulted in two broad requirements for support tools in analysis; open-endedness and transparency. These requirements were consistent with observations from earlier work on support tools for $C^2$, and were interpreted in the context communication analysis. Second, the similarity between the processes of analyzing research data and performing data mining suggested that there could be a sufficient overlap between the affordances of data mining approaches and the requirements of the research analysis process to justify using a data mining approach in $C^2$ research. We therefore crafted a prototype tool for selecting and representing team communications based on text clustering (Rosell and Velupillai, 2008) and other features which we tailored to the purpose of selecting parts of team communications for analysis. Our development focused on providing a transparent representation of message clustering and several components for managing communications, simulation data and observations using a common set of manipulations, thereby permitting an open-ended use of the tool.

The workshop evaluation indicated that the approach taken by the Workflow Visualizer tool was highly relevant to the tasks performed in $C^2$ analysis. The workshop participants correctly understood the tasks presented and how the tool was intended to support their work. They could clearly articulate several possible applications and desirable features of a tool for selecting parts of a communication flow based on text clustering or other techniques for extracting communication patterns. They also mentioned how the tool could be used in more general settings where not only communication data is analyzed but possibly also video streams and voice communications. The relationship between the timeline-based representation and the selection of events and messages was considered straight-forward, and they could readily reason about how to extract information about messages given their appearance in the timeline.

However, the workshop evaluation also indicated that the speculative nature and opaque reasoning of both the communication chart (Figure 6) and the message clustering (Figure 10) were obstacles for the participants in evaluating the utility of the tool. Although the communication chart was simple to understand once we explained the details of how the colored threads were constructed, it was something that seemed obscure due to the lack of a direct representation of *how the threads and clusters had been created* from the messages in the communication flow. They also gave indications that the proposed approach to analyzing messages by using mathematical models of text similarity was difficult to relate to as a general method. They did not use vector-space models for other purposes and had difficulties reconciling the concepts introduced by them to their understanding of communication data. Related to these remarks, Charlotte had explained in her interview how it was difficult to grasp the mathematical underpinnings of the LISREL models she used and that several possible models could be inferred in situations where only one might make sense in the context of the scenario. Thus, even if all the requirements for conducting a mathematical analysis are presented in the graphical user interface and the researcher understands well how to use the tool, introducing an analysis tool with no firm knowledge of the theoretical foundations underlying it introduces the risk that conclusions drawn with it may be frail. In the case with Random Indexing, the conclusions possible with small datasets may not depend on the features of the vector representation but on domain-specific artefacts (Rosell, 2009).

One option for presenting the rationale behind the clusters would be to animate the clustering by showing words and messages in an Infomat representation and demonstrate how the co-occurrences of words represent the total similarity measure of messages. Although each word is represented as a vector with high dimensionality which defies representations in 3D, pairs of words could be presented visually with colors representing the scalar cosine distance between the word vectors. Such a distance could give indications of how the randomly assigned word vectors contribute to the distance measures of messages. Similarly, an animation of how the system believes messages can be organized in threads could use a timeline, along which the selected messages are plotted and compared to others in sequence to highlight the features (e.g. when messages are sent) that determine how threads are constructed.

## 7 Conclusions

Our objective in this paper was to study options for using automatic pattern extraction to support researchers analyze $C^2$ communications data. We elicited two design criteria (open-endedness and transparency) for $C^2$ analysis tools from interviews with researchers, and interpreted these for the construction of an analysis tool based on text clustering.

The workshop participants considered the tool highly relevant, with several potential applications in their work. In particular, they expressed appreciation of the *open-endedness* of the tool: it could be used for different kinds of data, with little assumptions about data and with multiple modes of selecting and manipulating data.

Concerns and issues raised during the workshop illustrated the relevance of *transparency* as a design criterion for support tools, but also the challenge of interpreting it in the context of tool design. The participants could clearly see how messages were grouped by threads or clusters, yet they felt that the models used for creating them were too opaque. The visual representations did not connect well enough to their own models of how messages relate to others in episodes or categories.

Taken together, the evaluation results could be directly related to the design criteria elicited from the interviews, which corroborate their importance as criteria in analysis tool design. The use cases suggested by the participants in which they would like to use tools for automatic pattern extraction point to the relevance and promise of text clustering for understanding $C^2$ data. We believe that, with the appropriate visual cues and representations of how an automatic approach groups messages, text clustering could become a valuable asset for command and control research.

## 8 Acknowledgments

This work was supported by the Swedish National Defense College. We would like to thank the participants at the Swedish Defense Research Agency and VSL Systems AB for participating in this study and generously providing material used for the scenarios in this article.

## References

Alberts, D. S., Gartska, J. J., and Stein, F. P. (2000). *Network Centric Warefare: Developing and Leveraging Information Superiority*. National Defense University Press, Washington, DC.

Albinsson, P.-A. and Morin, M. (2002). Visual exploration of communication in command and control. In *Proceedings of the Sixth International Conference on Information Visualisation*, London, UK.

Albinsson, P.-A., Morin, M., and Thorstensson, M. (2004). Managing metadata in collaborative command and control analysis. In *Proceedings of the 48th Annual Meeting of the Human Factors and Ergonomics Society*.

Andriole, S. J. (1989). *Handbook of Decision Support Systems*. TAB Books Inc.

Cutting, D. R., Karger, D. R., Pedersen, J. O., and Tukey, J. W. (1992). Scatter/gather: A cluster-based approach to browsing large document collections. In *Proceedings of the 15th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*.

Flanagan, J. C. (1954). The critical incident technique. *Psychological Bulletin*, 51(4).

Gorman, J. C., Foltz, P. W., Kiekel, P. A., and Martin, M. J. (2003). Evaluation of latent semantic analysis-based measures of team communication content. In *Proceedings of the Human Factors and Ergonomics Society 47th Annual Meeting*.

Jensen, E. (2007). Sensemaking in military planning: a methodological study of command teams. *Cognition, Technology & Work*.

Jenvald, J. and Eriksson, M. (2009). Structured reflective observation in continuing training. In *Proceedings of the 8th WANO Human Performance Meeting*.

Jöreskog, K. G. (1973). A general method for estimating linear structural equation systems. In Goldberger, A. S. and Duncan, O. D., editors, *Structural Equation Models in the Social Sciences*. Seminar Press.

Kanerva, P., Kristoferson, J., and Holst, A. (2000). Random indexing of text samples for latent semantic analysis. In *Proceedings of the 22nd Annual Conference of the Cognitive Science Society*.

Klein, G. (1998). *Sources of Power. How People Make Decisions*. MIT Press, Cambridge, Massachusetts.

Lawson Jr., J. S. (1981). Command control as a process. *IEEE Control Systems Magazine*.

Lazar, J., Feng, J. H., and Hochheiser, H. (2010). *Research Methods in Human-Computer Interaction*. Wiley.

Leifler, O. (2008). Combining Technical and Human-Centered Strategies for Decision Support in Command and Control — The ComPlan Approach. In *Proceedings of the 5th International Conference on Information Systems for Crisis Response and Management*.

Leifler, O. and Eriksson, H. (2010). Studying professional teamwork through message classification. Submitted for publication.

Miles, M. B. and Huberman, A. M. (1994). *Qualitative data analysis: an expanded sourcebook*. SAGE.

Morin, M. and Albinsson, P.-A. (2005). *Creating High-Tech Teams: Practical Guidance on Work Performance and Technology*, chapter Exploration and context in communication analysis, pages 89–112. APA Press.

Rosell, M. (2009). *Text Clustering Exploration - Swedish Text Representation and Clustering Results Unraveled*. PhD thesis, KTH School of Science and Communication.

Rosell, M. and Velupillai, S. (2008). Revealing relations between open and closed answers in questionnaires through text clustering evaluation. In *Proceedings of LREC 2008*, Marrakesh, Morocco.

Ross, K. G., Klein, G. A., Thunholm, P., Schmitt, J. F., and Baxter, H. C. (2004). The recognition-primed decision model. *Military Review*, pages 6–10.

Rubel, R. C. (2001). War-gaming network-centric warfare. *Naval War College Review*, 54(2):61–74.

Sanderson, P. and Fisher, C. (1994). Exploratory sequential data analysis: Foundations. *Human-Computer Interaction*, 9:251–317.

Sanderson, P., Scott, J., Johnston, T., Mainzer, J., Watanabe, L., and James, J. (1994). MacSHAPA and the enterprise of exploratory sequential data analysis (ESDA). *International Journal of Human-Computer Studies*, 41(5):633–681.

Schmitt, J. and Klein, G. (1999). A recognitional planning model. In *Proceedings of the 1999 Command and Control Research and Technology Symposium*, Naval War College, Newport, Rhode Island, USA.

Shattuck, L. G. and Woods, D. D. (2000). Communication of intent in military command and control systems. In McCann, C. and Pigeau, R., editors, *The Human in Command: Exploring the Modern Military Experience*, pages 279–292. Kluwer Academic/Plenum Publishers, 241 Borough High Street, London.

Thorstensson, M., Axelsson, M., Morin, M., and Jenvald, J. (2001). Monitoring and analysis of command post communication in rescue operations. *Safety Science*, 39:51–60.