10TH INTERNATIONAL COMMAND AND CONTROL RESEARCH AND TECHNOLOGY SYMPOSIUM
THE FUTURE OF C2

ENABLING BATTLEFIELD VISUALIZATION: AN AGENT-BASED INFORMATION MANAGEMENT APPROACH

TOPIC: HUMAN FACTORS ENGINEERING

AUTHORS: GLENN TAYLOR (GLENN@SOARTECH.COM)
DR. SCOTT WOOD (SWOOD@SOARTECH.COM)
KEITH KNUDSEN (KNUDSEN@SOARTECH.COM)

SOAR TECHNOLOGY, INC.
3600 GREEN COURT SUITE 600
ANN ARBOR, MI 48105
P: 734-327-8000
F: 734-913-8537

ABSTRACT

A key task of a commander is that of Battlefield Visualization – understanding the situation in order to make decisions to achieve operational goals. Central to this process is managing the information needed to make those decisions. As the battlefield becomes more complex, and the stresses on commanders more apparent, the need for automated tools to reduce the burden only increases. In this paper, we identify the requirements of a system for enabling battlefield visualization through automating the information management process. We describe an architecture for information management using intelligent interface agents to assist a commander with battlefield visualization. Our approach focuses on a knowledge-driven process of information management, in which the commander's information requirements (CCIRs) are understood within the current context by automatically decomposing them into specific, sensor-relevant collection needs, tasking available collection assets to gather the data to answer the information requirements, then fusing that data into decision-relevant knowledge to be presented to the commander. We describe the results of our effort and a feasibility prototype to illustrate the central ideas of our approach.

Enabling Battlefield Visualization: An Agent-based Information Management Approach

Glenn Taylor, Dr. Scott Wood, Keith Knudsen
Soar Technology, Inc.
3600 Green Court Suite 600
Ann Arbor, MI 48105
P: 734-327-8000
F: 734-913-8537
(glenn,swood,knudsen)@soartech.com

ABSTRACT

A key task of a commander is that of Battlefield Visualization – understanding the situation in order to make decisions to achieve operational goals. Central to this process is managing the information needed to make those decisions. As the battlefield becomes more complex, and the stresses on commanders more apparent, the need for automated tools to reduce the burden only increases. In this paper, we identify the requirements of a system for enabling battlefield visualization through automating the information management process. We describe an architecture for information management using intelligent interface agents to assist a commander with battlefield visualization. Our approach focuses on a knowledge-driven process of information management, in which the commander's information requirements (CCIRs) are understood within the current context by automatically decomposing them into specific, sensor-relevant collection needs, tasking available collection assets to gather the data to answer the information requirements, then fusing that data into decision-relevant knowledge to be presented to the commander. We describe the results of our effort and a feasibility prototype to illustrate the central ideas of our approach.

1 INTRODUCTION

1.1 IDENTIFICATION AND SIGNIFICANCE OF THE PROBLEM

As the battlefield becomes less predictable, and the amount of data available to warfighters grows exponentially with the employment of new sensor platforms, the commander and staff's ability to manage this information can dramatically. This is particularly true in the Unit of Action, which is the first operating unit deliberately built around the idea of Network Centric Warfare (NCW) technologies, such as the Global Information Grid (GIG). New information sources can increase the information burden on C2 elements, which points to the need for automated tools to help manage that information in order to improve decision-making.

This effort addresses the challenge of improving the decision-making process by 1) developing techniques to allow the commander to specify and track information requirements, 2) developing intelligent-agent technology to support information management, and 3) determining techniques for displaying and presenting required information to support decision-making. The aim of our effort is to reduce information- and cognitive-overload, improve the speed and quality of the decisions that are made, and enable warfighters to better focus on the information that is most important to mission success.

1.2 BACKGROUND: BATTLEFIELD VISUALIZATION, MILITARY DECISION-MAKING, AND INFORMATION MANAGEMENT

There are three major doctrinal processes related to decision-making: Battlefield Visualization, the Military Decision-Making Process, and Information Management (FM 6-0). Battlefield Visualization is a three-step command process whereby the commander develops a clear understanding of the current situation, envisions a desired end state, and visualizes the sequences of activity that will move his force from its current situation to the desired end state. As shown in Figure 1, the foundation of Battlefield Visualization is the information loop, which consists of defining the commander's critical information requirements (CCIR), collecting data related to those requirements, and transforming collected data into situational understanding. This final transformation process has traditionally been associated with data or information fusion (especially fusion levels 2-4; (Waltz and Llinas 1990)). The process of breaking high-level decisions into fine-grained information collection needs could be called a "decision fission" process. This fission process is related to the question decomposition processes in advanced question-answering systems – see, for example, (Harabagiu and Lacatusu 2004).

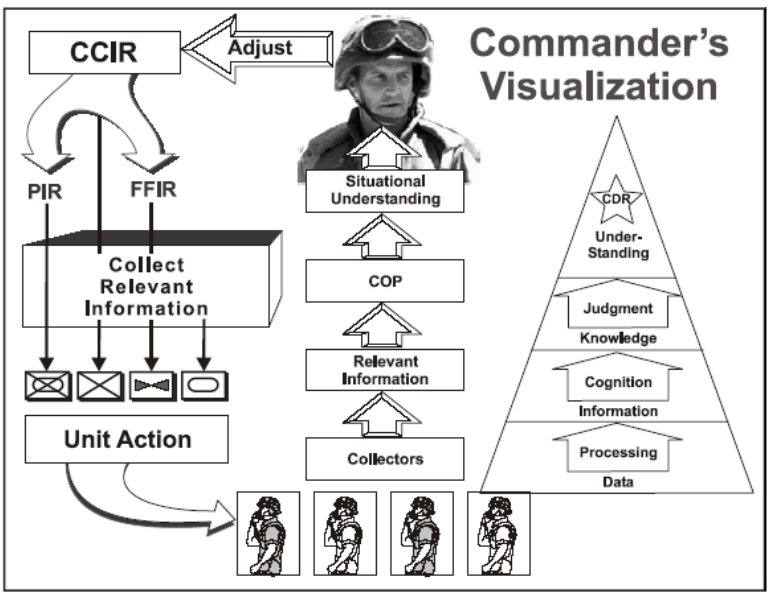

Figure 1: Commander's Battlefield Visualization (from FM 6-0, p4-5)

To help accomplish the planning aspect of Battlefield Visualization, the Military Decision-Making Process (MDMP) provides a detailed, regimented sequence of steps for developing an operational plan to accomplish mission goals. The MDMP starts with the receipt of a mission from higher command, steps through the process of analyzing the mission and developing a course of action (COA) and concludes with the production and dissemination of an OPORD to the unit. The MDMP lays the foundation for decision-making during execution of the OPORD, by specifying points in the execution plan at which decisions (i.e., selections of pre-planned "branches") must be made.

One output of the MDMP is the commander's critical information requirements (CCIR), which are inextricably linked to commander decisions. Doctrinally, CCIR is defined as including Priority Intelligence Requirements (PIR), Friendly Force Information Requirements (FFIR), and (somewhat adjacently) Essential Elements of Friendly Information (EEFI). This categorization helps define how information is reported by elements in the field, deliberately collected, or deliberately concealed. Table 1 defines these categories.

Table 1: Critical Information Requirements (from FM 101-5)

PIR – Those intelligence requirements for which a commander has an anticipated and stated priority in his task planning and decision-making
FFIR – Information the commander and staff need about the friendly forces available for the operation.
EEFI – Critical aspects of a friendly operation that, if known by the threat, would subsequently compromise, lead to failure, or limit success of the operation, and therefore must be protected from threat detection.

CCIR are essentially a collection of PIR, FFIR, and EEFI that have been elevated in importance by the commander because they support decision-making, and so are given higher priority in resources allocated to find or protect them. Where collection is required, PIR must be further refined into very concrete requirements that support collection – these are called Specific Information Requirements (SIR), which are drawn up and used to task particular assets for collection via specific orders or requests (SORs). As these assets return reports, the information is combined to determine if a CCIR has been met.

Well-defined Information Requirements capture the five W's: who (subject of IR), what (activity of subject), where (area of activity), when (time window in which information is required), and why (what decision the IR relates to). Additional data is required for tracking IR through the entire IM process, from conception, through collection, and reporting to the commander. A commander's initial specification of a CCIR might only specify a portion of this information – a commander's staff would provide the rest.

The Information Management (IM) process is defined by doctrine to manage these CCIRs and related information. IM spans planning and execution (FM 101-5, Appendix I) and focuses on getting the commander the information required to make decisions. The IM process includes determining the indicators that would answer the CCIR defined in the MDMP, and so indicate which decisions should be made. During mission execution, the information requirements are constantly changing to account for changes in the mission or the situation. To accommodate this, the information manager must adjust the priorities of information requirements, and manage limited assets to address new requirements. As the battlefield is dynamic, so must the IM process be.

2 Objective Paradigm: Knowledge-Rich Intelligent Agents for Information Management Automation

This effort facilitates a commander's battlefield visualization by addressing the information requirements that the commander deems critical to decision-making. Our operating paradigm is the automated assistance of routine tasks using software agents. Such agents are becoming commonplace in assisting with tasks such as scheduling meetings, purchasing products, and searching for information (Wooldridge 2000).

One category of agents that refines the automated assistance paradigm is the interface agent (Laurel 1991). Interface agents are designed to reduce the complexity of human-system interaction, and can take the form of relatively simple agents for performing such tasks as mail filtering, or can be more complex for tasks such as web-based information seeking (Lieberman 1997). Essentially, interface agents provide a simplifying abstraction between a human and a computer. Some interface agents may operate entirely in the background, such as in email filtering. Others may more directly interact with the human user in performing a task, such as in simplifying the specification of a complex command to decrease task execution time and error rates. Interface agents may also help with information display, filtering irrelevant or extraneous information to provide critical information for the task at hand.

In the broadest sense, "interface" means a functional layer between two operating elements. In these terms, we view the commander's staff as the interface between the commander and the maneuver elements in the field. In the same way, an automated commander's support tool provides software agents that perform the interface functions of the commander's staff. Specifically, one identified role from doctrine is that of the Information Manager:

> The information manager…outlines and monitors the staff's performance and responsibilities in processing information to support he operation and flow that feeds the commander's requirements. He collects, tasks, analyzes, and presents the CCIR in a timely and accurate manner. (FM 101-5, Appendix I)

Recent work with commanders has shown the efficacy of some forms of automation, and has pointed to the need for automating portions of IM (Zaientz et al. 2005). Our effort focused on the feasibility of creating a suite of interface agents to assist in the Information Management process, analogous to the interface a command staff provides. What separates the interface agents here from those that perform simple, routine tasks is in the amount of knowledge these agents must have to perform their tasks. Given the types of information relevant to a commander, the agent must have a deep understanding of information requirements, military operations, the current mission, operational planning, tasking, and monitoring, information fusion, and information display. These agents must understand not only how general information relates to a given mission, but also understand the dynamic situation enough to recognize what is normal in an operation and when something is anomalous.

2.1 Capability Requirements

The basis of our work is an understanding of the current doctrinal processes surrounding Information Management and how it supports Battlefield Visualization. In analyzing these processes, we have identified several key capabilities, in addition to those dictated by the doctrine, which would be required within an automated system to help support the commander's visualization process. We have concluded the need for an agent system that fully embodies the Information Management process; namely, tasks of collecting, processing, and relating acquired information to the commander in a timely and relevant manner. These capabilities presume that either the user (commander) explicitly defines CCIRs for the agent system (via a graphical interface), or the agent can distill CCIR from other MDMP/wargaming work products, such as the Decision Support Template and Decision Point criteria.

2.1.1 Knowledge Representation and Reasoning Requirements

A central task of an automated IM system is understanding the user's information requirements, what they mean in terms of collection, and their relationships with decision points. There are several types of knowledge required in the system to fulfill the capabilities listed above. Table 2 offers a brief breakdown of the different types of knowledge required for Information Management.

Table 2: Knowledge Required for an Information Management System

Knowledge Type	Definition
Situation Knowledge	Information about the current situation, including mission (OPORD, decision points), enemy, own troops (especially assets for tasking), terrain, time, and civilian aspects of the battlefield (METT-TC).
Doctrinal Knowledge	The meanings of common operational terminology in terms of concepts and relationships between those concepts, as defined in FM 101-5-1
IR Knowledge	The knowledge required to transform information requirements between their different forms (PIR to SIR, indicators to PIR) and relate them to the mission
Planning Knowledge	Includes the ability to generate reconnaissance and surveillance plans, and fit them into existing OPORDs.
User Knowledge	A constantly updated understanding of the user, including his overall objectives, his current goals and tasks, workload, etc., used to manage when and how information is delivered.
Display Knowledge	Knowledge about how to present information to the user, including user preferences for information, and information about particular display devices available.

Consider a hypothetical example that illustrates the information loop from Battlefield Visualization. Suppose a CCIR from a cordon and search mission: “Report if searches are being canalized toward or away from Objective Mike.” The first thing that must be understood about this is the language. Terms like *report, search, canalize*, and *area* all have very specific meanings within military operations, within a cordon and search generally, and within the context of the current mission. The meaning of the CCIR can be understood only if its piece-parts are understood across all these levels of information. Once the system has an understanding of what is critical to the decision, and is requested in the CCIR, it can then begin the decision fission process to determine what indicators or activity should be looked for to determine if the CCIR is met. In part, this requires a basic understanding of what it means to *canalize* (*to restrict operations to a narrow zone…by use of obstacles* – FM 101-5-1). In this example, the system must know what might indicate canalizing, such as a certain configuration of IEDs, light enemy contact from a consistent direction, or obstacles. The system would need some understanding of the current enemy’s tactics (reliance on IEDs and small infantry units). Also, since obstacles can be used to canalize, the system must understand what is meant by *obstacle* (*any obstruction designed or employed to disrupt, fix, turn, or block* – FM 101-5-1) and what might serve as obstacles in the current environment. A partial ontology that encodes the meaning of *obstacle* is given in Figure 2.

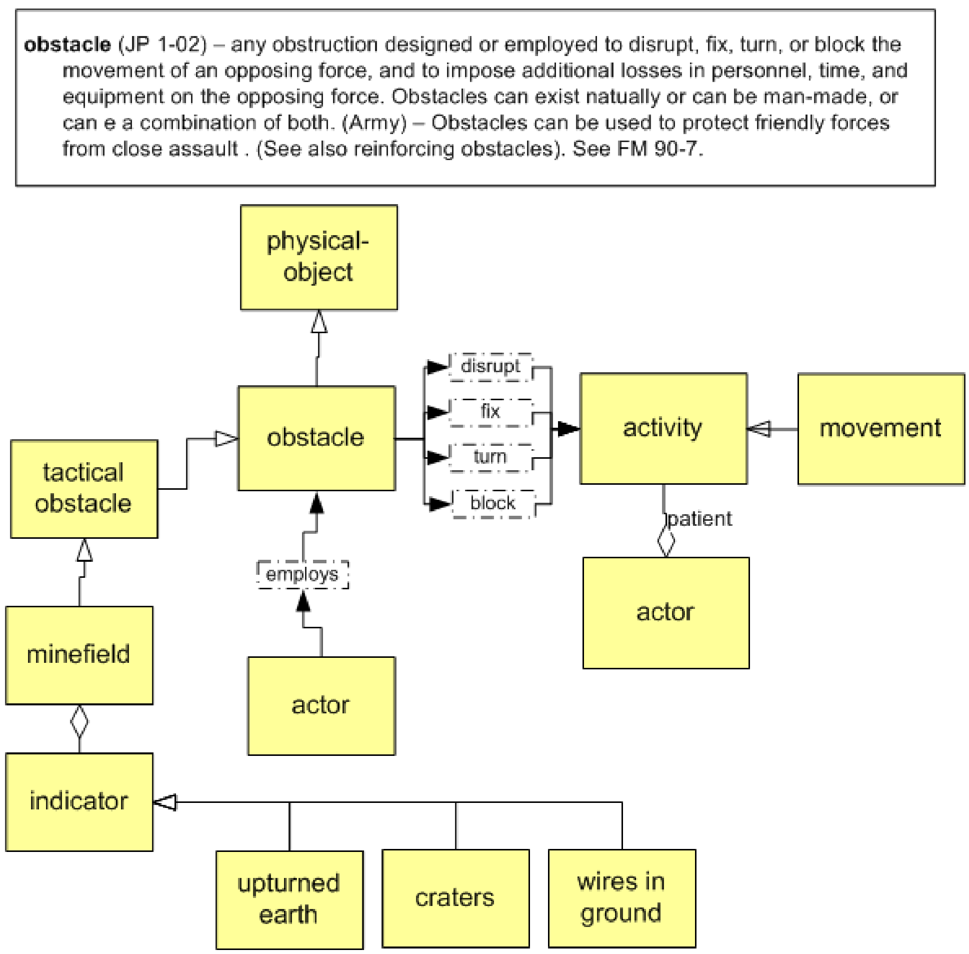

Figure 2: A partial ontology for the term 'obstacle' derived from FM 101-5-1

From this understanding of indicators, the system must construct specific information requirements (SIRs) that specify precise areas and times to look for these indicators. This requires an understanding of the current operation, where it is taking place, what the scheme of maneuver is, and when the indicators would be relevant. An example SIR derived from the original CCIR is: “*Report all evidence of IEDs in* *and around OBJ Mike – upturned earth, suspicious packages, loose wires sticking out of ground -- LTIOV 1700 25 Jul 04"* Multiple such SIRs can be derived from the initial CCIR. From these SIRs, the system must understand what assets are capable of sensing the indicators – UA infantry companies engaged in the search might be the ones to find the IEDs; UAV assets might be the ones to find obstacles in advance of a maneuvering unit. (Though in an NCW environment, it may be possible to send a request on the network looking for collection assets that meet certain criteria, such as the ability to sense the named indicators.) Those specific assets are then tasked via SORs. When reports come back from these assets, the system must be able to make sense of the report, both its format and its content. The reports give evidence for an indicator, and multiple reports from all the tasked assets need to be fused together to obtain an overall picture to know if the CCIR has been confirmed or denied. Supporting these dual processes of *decision fission* and *information fusion* is a core aspect of an architecture for IM.

The information required for this kind of reasoning includes both dynamic information from the environment that could be stored in databases, and static information (such as doctrinal information) that can be stored in ontologies, as in Figure 2. Such representations will eventually need to be developed for an automated system to understand and reason over common military terms.

2.1.2 HUMAN-COMPUTER INTERACTION REQUIREMENTS

The primary purpose of the AIM-TRU system is to assist the commander in decision-making. Presenting the user with a natural interface is critical to his use of the system. Based on doctrinal sources (FM 101-5, etc) and other studies (Ingram 1996), we identified the activities of the commander within the MDMP and the Information Management processes, such as his interactions with the Information Manager and other S2 staff. The results of these interactions are, at various times in the process, an initial approved set of CCIRs, commander's modifications to the CCIRs as the mission progresses, and reports regarding the status of those CCIR. Ideally, the typical commander-staff interactions would be preserved in an automated system. Table 3 maps these commander-staff interactions to commander-system interactions at different points in the process.

Table 3: HCI Requirements Analysis

Commander's Need	HCI Requirements
Task 1: Generating CCIR (MDMP)	
A list of the CCIRs that emerge from the war game process.	• Allow the commander to add or remove CCIR from the list as well as modify any CCIR in the list.
ISR collection plan	• Present the recommended list of CCIRs and derived indicators and SIRs to the commander for review and modification, if necessary. • If assets are not available to gather information for a CCIR and IMA is unable to resolve the situation, the GUI must be able to alert the commander. • Present the resource allocation plan to the commander for review and possible modification.
Task 2: Tracking CCIR	
Assistance tracking assets' execution (i.e., are the tasked assets able to perform their assignments).	• Upon request, display the relationship between CCIR and assets to the commander for review and possible modification. • If assets become unable to gather information for a CCIR and IMA is unable to resolve the situation, the GUI must be able to alert the commander.
Assistance tracking all indicators and CCIR status.	• Display the relationship between CCIR, SIR and indicators as indicators are being reported.
Task 3: Responding to CCIR	
Be alerted when CCIR is triggered.	• Alert the commander to CCIR that are triggered in a manner cannot be ignored, but that does not prevent the completion of whatever task the

	commander is involved with.
Assistance Responding to a CCIR	• Provide the commander easy, intuitive access to CCIR context (all indicators, history, wargaming and all other related information). • Provide a simple mechanism for the commander to request additional information from assets regarding the CCIR, SIR and indicators. • Present suggested strategies for responding to the CCIR.

An automated system must have an interface that is as natural to deal with as command staff. For example, throughout the doctrinal manuals CCIRs are typically specified in natural language expressions grounded in the standard military terminology. Other natural human-human interactions might include mixed speech with gesture and artifacts such as maps.

3 Technical Approach

Our approach to assisting the battlefield visualization process centers on developing an agent-based service framework to embody the information management process performed by the command staff. We adopt the Intelligent User Interface (IUI) metaphor (Maybury 1998), wherein the interactions with the user are driven by intelligent interface agents that, themselves, interact with other services to perform aspects of information management. The agents may elicit information requirements from the user, develop and execute collection plans, and present the information to the user in helpful ways. The agents in this approach must be *knowledge-rich*; that is, they have a deep understanding of their task, and can bring to bear large amounts of knowledge to perform the kinds of problem-solving necessary in these complex tasks. Such activities include planning, interaction with other agents, and managing the roles and responsibilities within a larger system of agents and humans.

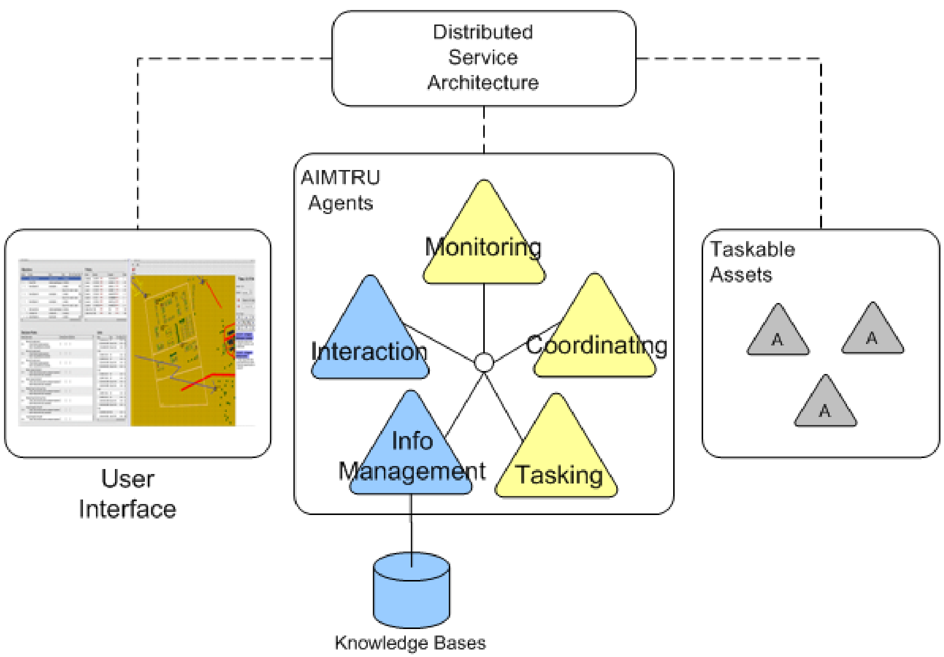

Figure 3: Overall AIM-TRU System Design Concept

For this entire system, we presume a distributed service-based architecture, where components can be producers or consumers of information. Each component registers with a central server with its capabilities. Interaction between components is done entirely through message-based communication that is moderated through the message server. The overall system design concept is given in Figure 3.

While these individual pieces may be replaced with others in a final implementation, we believe the capabilities provided by these components are required in an automated IM system. This is especially true when looking forward to a "smart information pull" approach within the future Network-Centric Warfare vision of a Global Information Grid (Alberts et al. 1999).

The foundation for our design for this work has been the CIANC3 framework. CIANC3 defines a multi-agent system whose agents perform the roles of tasking (high level task assignments from the mission plan), monitoring (mission execution status monitoring), and coordination (assigning tasks to individual assets). See (Wood 2003; Wood et al. 2004) for more details about these agents. Additionally, we draw on the BINAH architecture (Zaientz 2003), which extends the CIANC3 architecture to include data fusion and information display capabilities. We extend CIANC3 and BINAH by introducing two new agents to manage the IM process and interactions with the user:

Information Management Agent. The Information Management Agent (IMA) is responsible for working with the commander and the other agents in order to collect and provide to the commander the information required for timely decision-making in the planning and execution of a mission. Specifically:

- Eliciting the commander's critical information requirements (CCIR)
- Analyzing CCIRs in the current context to derive detailed, actionable SIRs
- Augmenting the mission plan with the ISR annex and assigning assets for collection
- Analyzing intelligence reports from the battlefield to relate low-level indicators to higher-level information requirements
- Reporting priority information to the commander for decision-making

Interaction Agent. The Interaction Agent (IA) is responsible for managing interactions with the user, using the Intelligent User Interface metaphor. Specifically, the IA is responsible for:

- Managing the direct user interaction for CCIR specification
- Presenting reports to the user in a form amenable to decision-making
- Managing knowledge about the user, including preferences, workload, and task context
- Managing the amount and forms of information presented to the user
- Tailoring input and output modalities to user preferences

In addition to these agents and the CIANC-based agents, the system definition includes the user, the user interface, and collection assets in the environment available for tasking.

4 FEASIBILITY PROTOTYPE

To illustrate the feasibility of the approach, we developed a representative version of the overall AIM-TRU system just described. The prototype includes a simpler multi-agent system working within a distributed service architecture, a user interface for CCIR management and status updates, and the illustrative FCS scenario described previously. The AIM-TRU prototype system integrates agents into a combined simulation and operational environment for information management. The reduced architecture is illustrated in Figure 4, and its components are described below.

In this prototype, the Information Management Agent (IMA) and Interaction Agent (IA) are essentially as defined above in, with a few simplifications. Namely, CCIRs are pre-specified as part of the scenario, rather than being elicited from the commander or derived from the MDMP. Also, the amount of situational reasoning knowledge in the prototype is limited to simple ontological knowledge about entity types (vehicles), and some spatial reasoning. The point is to illustrate the types of information transformation and sense-making these agents will need to perform.

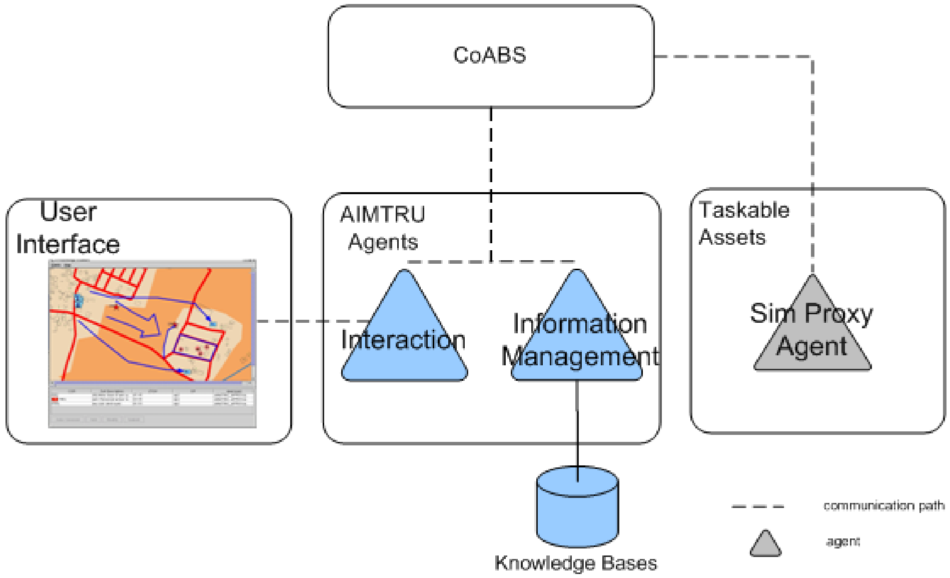

Figure 4: The Reduced Prototype Architecture

The demonstration system includes an agent that is a stand-in proxy for a simulation environment. That is, due to limited resources, we did not connect to a simulation engine such as OTB. Instead, we have scripted the movement and discovery of entities as if they were running within a simulation. The general reporting capabilities of these entities are required in a fully functional system. However, in the prototype, the report information is generated from the script rather than by actual entities in the environment. Note, though, the formats of the message are the same, and use the same interfaces and information channels, that would be used in the final system.

CoABS (DARPA-IPTO 2001) provides basic distributed service capabilities, such as discovery, capabilities registration, lookup, and a message transport layer. Being Java-based, CoABS is platform independent, and supports generic message content. It was selected for this prototype primarily due to our experience with it, and the ability to quickly build a demonstration around it. In an operational system, we would want to look at other service architectures that are being considered as the baseline technologies for the GIG, such as the Publish and Subscribe System (PASS), as well as other digital transport mechanisms used in military applications.

An agent communication language (ACL) provides a common way for agents to communicate. An effective ACL must enable interface agents to communicate between multiple echelon hierarchies of both robotic and human forces. The Foundation for Intelligent Physical Agents (FIPA) (Huhns and Singh 1997) has defined a flexible agent communication language called FIPA-ACL that provides a standardized message format for communicating between agents, which includes domain-independent features, as well as the ability to extend the message content to domain-specific applications.

The basis for the agents in this prototype is the Soar cognitive architecture (Laird et al. 1987). Soar is well-suited to applications in which a great deal of knowledge must be brought to bear to understand and act in complex environments. Because AIM-TRU is a service-based architecture, we could potentially use different agent types for different roles, as long as they all could communicate using the same ACL, and understood their responsibilities within the overall architecture.

We developed a prototype user interface for AIM-TRU, shown in Figure 5 below. This interface is meant only to illustrate some of the user-system interactions required within the AIM-TRU vision, and is not intended to be the final interface for a commander. The interface is based on VISTA, a generic, Java-based toolkit for creating visualization tools for agent systems (Taylor et al. 2002). VISTA was originally designed to serve as the basis for tools that help display information about an agent's behavior, its knowledge, and its decision-making processes. However, VISTA's generality and extensibility has made it suitable as the basis for intelligent user interfaces.

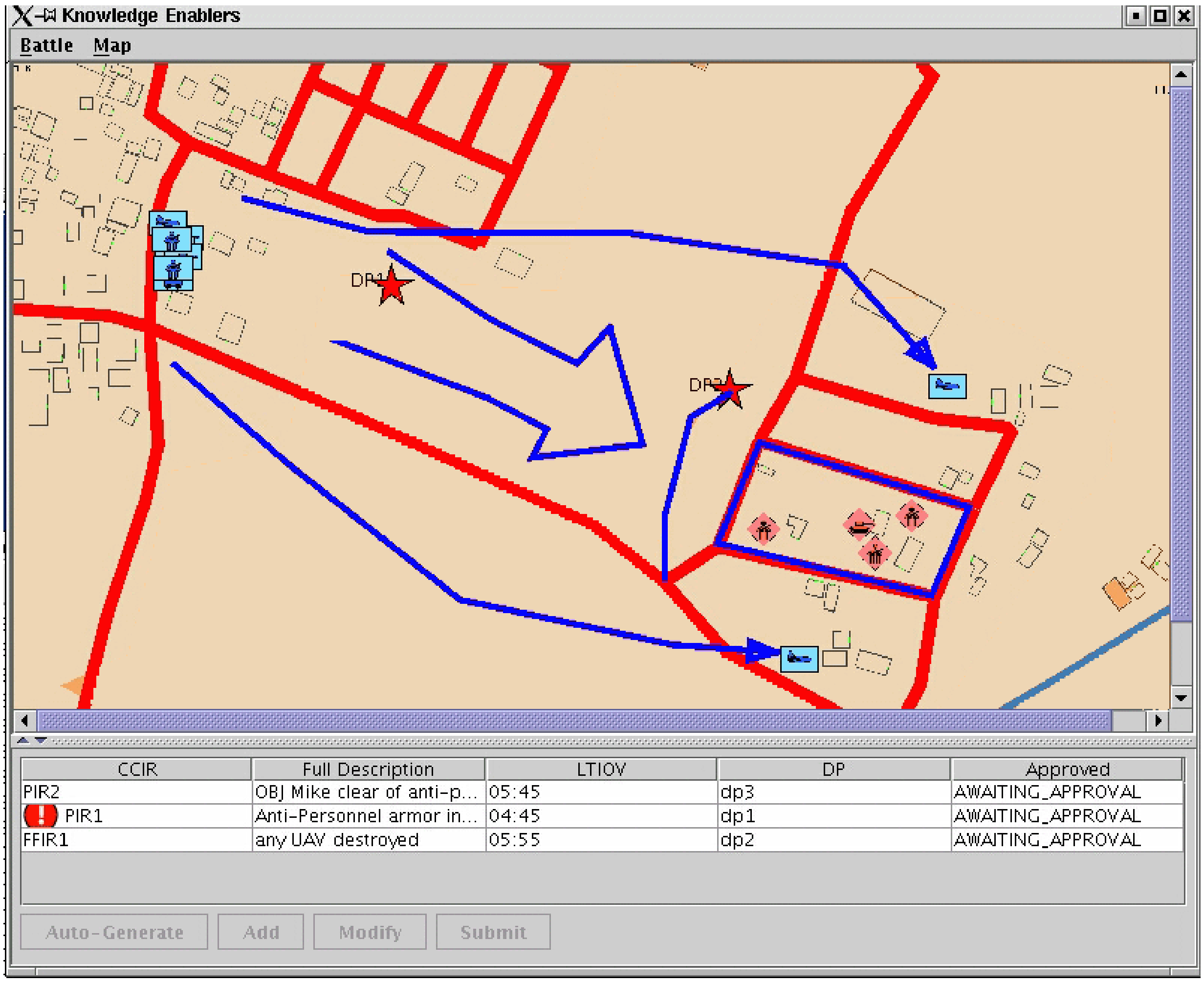

Figure 5. A Snapshot of the AIM-TRU Prototype User Interface

The user interface shows a compound breach scenario in progress. The upper pane shows graphical control measures that are defined to orient the user, and would ideally be part of the mission plan used by the system in reasoning. Icons are used for units known by the system – both friendly and enemy forces. Enemy force positions are populated based on sensors of the blue UAVs located north and south of OBJ Mike. The lower pane contains the CCIR for this mission, and the related decision points. Suggested CCIR are pre-defined as part of the scenario definition in the prototype. The Information Management Agent communicates the CCIR to the Interaction Agent, which then displays them in the GUI. The Commander's task is to select those IRs deemed to be critical for mission success. As the simulation runs, information in gathered by active blue assets, passed as SITREPs or SPOTREPs to the Information Management Agent, and passed again to the Interaction Agent, then updated on the screen, When their criteria have been met, CCIRs are "triggered." This triggering then causes the Interaction Agent to send a notification to the user that a CCIR has been met (denoted by the red exclamation point icon to the left), and that a decision point must be acted on. The system notifies the user that a decision is required, and the user must decide to execute or ignore the branch associated with the decision point. On engagement of a branch, the command is sent to the Interaction Agent, then to the Information Management agent to enable tasking collection assets.

While this interface is rudimentary in its presentation of information and the interactions with the user, it represents the kinds of interactions a user would have with the system during mission execution. We have so far left the design of more natural interfaces for CCIR specification to later work.

5 Relationship to Other Work

Other efforts have addressed various aspects of the CCIR process but none have specifically considered the relationship between declarative ontological representations, knowledge structures and goal-driven cognitive reasoning. The work of (Gerber et al. 1998), for example, addresses the issue of automatically extracting CCIR information from the output stream of the JANUS battlefield simulation system. In their solution a CCIR is an explicit mapping between a set of JANUS simulation output fields and either a USMTF S303 SALUTE or USMTF S507 Resource report. Here, the CCIRs can be defined so long as they fall into pre-defined categories of information that map directly to simulation artifacts. This rigid definition structure, and the interactions with the user, reflects neither the richness of CCIR, nor the naturalness of the commander-staff interactions in defining and reporting CCIR.

(Gratch et al. 1999) presents a CGF that derives its own CCIRs from an operational plan, and uses those CCIR during execution. The system has a few limitations, namely that the CCIR are not deliberately sought out through asset tasking, and the system has limited information fusion capabilities. However, some of the features of this system are in line with our work here.

Some work has been done in representing battlefield concepts in an ontological form, including (Rebbapragada et al. 2002; Matheus et al. 2003). Another effort attempts to describe a pseudo-natural language called the Battle Management Language (BML) based principally in the operational terms defined in FM 101-5-1 (Carey et al. 2001). The intent is that, if these terms could be defined in such a way that a computer could understand them, humans and computers could interact using this “natural” language within a NCW environment.

Information/data fusion has obviously received a great deal of attention in research and applications, though most of the successes have been at the lower levels of data fusion, rather than in the kind of conceptual information fusion that is required here. The “decision fission” process is analogous to question decomposition has received some attention in the question answering literature, though is still in its infancy. Both, as relative to this effort, are largely unsolved issues.

6 Conclusions and Future Work

We have identified the Information Management process as part of the commander’s Battlefield Visualization which can benefit from automation. We have developed a set of capability requirements and initial design for a knowledge-rich agent-based system to serve as a commander’s assistant for IM. We designed a multi-agent architecture (AIM-TRU) that extends the ongoing CIANC3 and BINAH work being developed by Soar Technology. These extensions include the addition of two additional agents, the Information Management Agent and the Interaction Agent. The overall system’s job is to assist the commander in this hypothesis generation and testing process. To the extent that the commander can communicate these hypotheses to the agent, the agent would be more capable of relating CCIR to decisions the commander has to make.

We have developed a simple prototype to demonstrate the feasibility of this approach, based on the example FCS scenario. The prototype integrates intelligent agents for IM and a graphical interface for user interaction, VISTA, within a distributed service architecture, CoABS. The agents interact using a well-defined communication protocol, FIPA-ACL. The agents employ a data-driven approach to problem-solving and draw upon formalized knowledge sources (ontologies) that help define their understanding of CCIRs, and also their roles, responsibilities, and relationships with others agents. The prototype development effort has already highlighted some of the organizational and integration issues that we will face in a more complete system.

Our initial work looked at the problem of CCIRs as somewhat isolated from the planning process. Future work will allow AIM-TRU to take advantage of the products of the MDMP, including the developed eCOAs, the decision support templates or the stated decision points and criteria. This process of understanding multiple CCIRs as trying to refute eCOA hypotheses would require another class of knowledge and reasoning to build an overall picture of the commander's problem-solving task.

We currently intend this tool to be used by the commander. A similar tool could just as well be used by the commander's staff in managing a mission. However, the system requirements, especially in the user interface, would likely be different for these two users

7 ACKNOWLEDGEMENTS

This work was performed under a Department of the Army Small Business Innovation Research Program for topic A04-081, contract # W911QX-04-C-0049. We gratefully acknowledge the sponsorship of the Army Research Laboratory, and thank ARL COTR Mr Tim Hanratty for his guidance and support. We would also like to thank LTC Scotty Abbott (USArmy--Retired) for his insight into Army operations and FCS, and the Soar Tech development team for their diligence and creativity.

8 REFERENCES

Alberts, D. S., Garstka, J. J. and Stein, F. P. (1999). Network Centric Warfare: Developing and Leveraging Information Superiority. Washington, DC, CCRP: DoD C4ISR Cooperative Research Program.

Carey, S., Kleiner, M., Heib, M. and Brown, R. (2001). Standardizing Battle Management Language -- A Vital Move Towards the Army Transformation. Fall Simulation Interoperability Workshop.

DARPA-IPTO (2001). Control of Agent-Based Systems. http://www.darpa.mil/ipto/programs/coabs/

Gerber, W., Henninger, A. and Stone, G. (1998). Intelligent Extraction, Filtering and Reporting of Critical Decision Information from a Simulation. 1998 Spring Simulation Interoperability Workshop, Orlando, FL.

Gratch, J., Marsella, S. and Hill, R. W. (1999). Deriving Priority Intelligence Requirements for Synthetic Command Entities. Computer Generated Forces and Behavioral Representation, Orlando, FL, SISO.

Harabagiu, S. and Lacatusu, F. (2004). Strategies for Advanced Question Answering. Workshop on Pragmatics of Question Answering (HLT/NAACL2004), Boston, MA.

Huhns, M. N. and Singh, M. P. (1997). Agents and Multiagent Systems: Themes, Approaches, and Challenges. Readings in Agents. San Francisco, CA, Morgan Kaufmann Publishers, Inc.**:** 1-23.

Ingram, M. C. (1996). Battlefield Visualization: Understanding Warfighter Priorities. Fort Leavenworth, KS, TRADOC Analysis Center.

Laird, J. E., Newell, A. and Rosenbloom, P. S. (1987). "SOAR: An Architecture for General Intelligence." Artificial Intelligence **33**: 1-64.

Laurel, B. (1991). Interface Agents: Metaphors with Character. The Art of Human-Computer Interface Design. B. Laurel. Reading, MA, Addison-Wesley Publishing Company, Inc.**:** 347-354.

Lieberman, H. (1997). Autonomous Interface Agents. The ACM Conference on Computers and Human Interaction [HCI-97], Atlanta, ACM Press.

Matheus, C. J., Baclawski, K. P. and Kokar, M. M. (2003). Derivation of Ontological Relations Using Formal Methods in a Situation Awareness Scenario. Multisensor, Multisource Information Fusion: Architectures, Algorithms, and Applications 2003, SPIE**:** pages 298–309.

Maybury, M. T., Wahlster, Wolfgang (1998). Intelligent User Interfaces: An Introduction. Readings in Intelligent User Interfaces. M. T. Maybury, Wahlster, Wolfgang. San Francisco, Morgan Kaufmann Publishers**:** 1-13.

Rebbapragada, L., Dietz, J. E. and Dawidowicz, E. (2002). Information Clustering for Commander Critical Information Requirements. Army Science Conference, Orlando, FL, US Army.

Taylor, G., Jones, R. M., Goldstein, M., Frederiksen, R. and Wray, R. E. (2002). VISTA: A Generic Toolkit for Visualizing Agent Behavior. Eleventh Conference on Computer Generated Forces and Behavior Representation, Orlando, FL, University of Central Florida.

Waltz, E. and Llinas, J. (1990). Multisensor Data Fusion. Norwood, MA, Artech House, Inc.

Wood, S. D. (2003). Cooperative Interface Agents for Networked Command, Control, and Communications (CIANC^3): Phase I Final Report, United States Army Research Institute for the Behavioral and Social Sciences.

Wood, S. D., Zaientz, J., Beard, J., Frederiksen, R., Lisse, S., Crossman, J. and Huber, M. J. (2004). An Intelligent Interface-Agent Framework for Supervisory Command and Control. Command and Control Research Technology Symposium (CCRTS), San Diego, CA, DoD Command and Control Research Program.

Wooldridge, M. (2000). Reasoning About Rational Agents. Cambridge, Massachusetts, The M.I.T. Press.

Zaientz, J. (2003). Information Delivery Configuration for Augmented Warrior Readiness: Phase I SBIR Final Report, US Air Force.

Zaientz, J., Wood, S. D. (2005). Integrating Decision-Centric Displays with Cooperative Intelligent Agents to Improve Semi-Autonomous Interpretation and Implementation of Commander's Intent. (In Process)

FM 3.0 Operations

FM 6.0 Mission Command: Command and Control of Army Forces

FM 34-8-2 Intelligence Officers Handbook

FM 34-8 Combat Commander's Handbook on Intelligence (ST 2-50.4)

FM 34-2 Collection Management and Synchronization Planning

FM 101-5 Staff Organization and Operations

FM 101-5-1 Operational Terms and Graphics