16th ICCRTS

"Collective C2 in Multinational Civil-Military Operations"

Towards Intelligent Operator Interfaces in Support of Autonomous UVS Operations

Primary Topic 8 - Architectures, Technologies, and Tools
Secondary (1) Topic 1 - Concepts, Theory, and Policy
(2) Topic 7 - Modeling and Simulation

Dr. Kevin Heffner
Pegasus Simulation Services Inc.
PO Box 47552, Plateau Mont Royal PS, Montreal, QC, Canada, H2H 2S8
Tel: (514) 600-0141
k.heffner@pegasim.com

Dr. Fawzi Hassaine
Defence R&D Canada – Ottawa
3701 Carling Ave. Ottawa, ON, Canada K1A 0Z4
fawzi.hassaine@drdc-rddc.gc.ca

Abstract

Experience in recent conflicts indicates the employment of Unmanned Vehicle Systems (UVS) will continue to grow in coming years. New UVS capabilities involve greater complexity of payloads and interactions within unmanned vehicle (UV) subsystems, among UVS and between UVS and other systems, including Command and Control (C2) systems. This introduces additional requirements for UV operators. In some situations UV operators easily can be faced with cognitive information overload, while increasing UVS complexity and future concepts of employment such as single-operator multiple-UV operation require increased operator attention.

In order to attain the required level of operator efficiency, it is necessary to introduce higher levels of autonomy within the UVS subsystems in conjunction with the use of intelligent operator interfaces. This will allow for greater flexibility and effectiveness in supporting future mission requirements wherein UVS operator interfaces are able to reduce the work load, and allow operators to function at higher levels of abstraction.

In this context, this study justifies the employment of intelligent systems to attain higher levels of autonomy for a specific family of UVS, which are the Unmanned Aerial Systems (UAS). The proposed approach is based on various automation management strategies, combined with the use of formal languages for effectively capturing information elements flowing between the Unmanned Aerial Vehicle (UAV) operator and the UAS subsystems. This paper also proposes a technical approach towards the experimentation of these UAS concepts in a simulation environment using the Coalition Battle Management Language (C-BML) as an enabling technology for the interoperation of the C2 systems with some of the UVS subsystems.

1. Introduction

As witnessed in recent conflicts, there has been a significant increase in the employment of Unmanned Systems (US) by military forces over the last decade, in particular the use of Unmanned Aerial Systems (UAS). Success in achieving mission objectives combined with increased technology capability has led to new operational requirements and the need to increase UAS effectiveness. However, one of the key limitations to increasing future UAS effectiveness lies in the human factors challenges associated with the UAV operators' workload [1].

Additionally, a recurring operational requirement across the military services is the need to increase the levels of autonomy of UAS in order to optimize workflows for tasking, monitoring and disseminating information from these highly valued C4ISR assets [2]. For example, with increasing levels of UAS autonomy, UAV operators are less solicited to exercise lower-level control tasks and are therefore able to focus on higher-level tasks – the so-called Human Supervisory Control (HSC) – more closely related to mission goals. Similarly, freed from lower-level tasks, a single UAV operator may be able to operate multiple platforms.

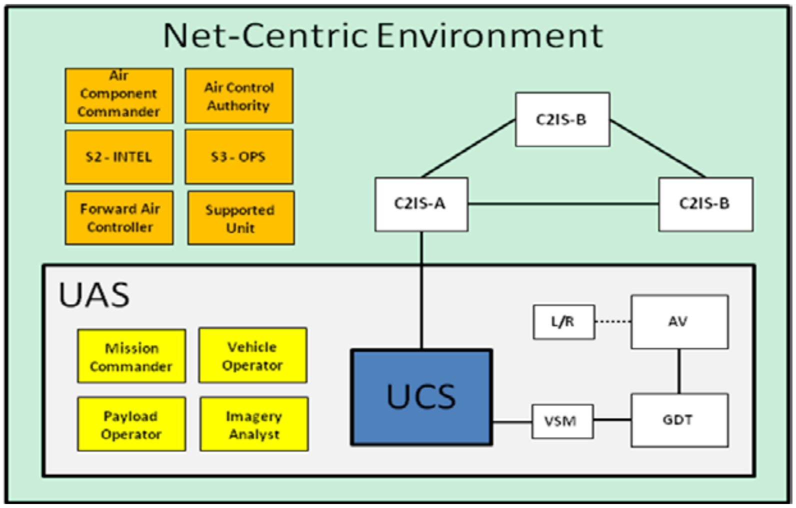


Figure 1-UAS Overview

1.1. Unmanned Aircraft Systems Overview

Figure 1 depicts a notional UAS in a net-centric environment. The UAS is generally comprised of the UV Control Station (UCS), the Vehicle Specific Module (VSM), the Ground Data Terminal (GDT), the Air Vehicle (AV) and the Launch and Recovery (L/R) element. Military personnel that are typically associated with the UAS are shown in yellow: the Mission Commander (MC), the Vehicle Operator (VO), the Payload Operator (PO) aka Mission Payload Operator (MPO) and the Imagery Analyst (IA).

The external stakeholders that interact with the UAS are shown in orange and include: the Air Component Commander (ACC), the Air Control Authority (ACA), the Intelligence Staff Officer (S2), the Operations Staff Officer (S3), the Forward Air Controller (FAC) and the Supported Unit; with the FAC only being present in the case of Close Air Support (CAS).

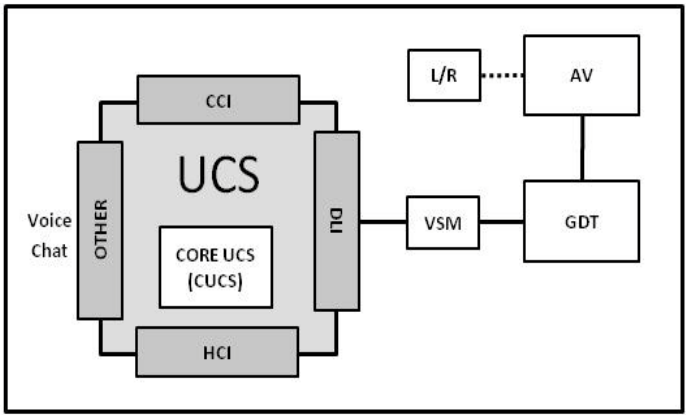


Figure 2- Notional UAV Control Station Architecture

1.2. UAV Control Station (UCS)

The UCS may be ground-based (i.e. Ground Control Station), transported during operations in another air vehicle or in a ground vehicle or may be remotely located. The NATO STANdardized AGreement (STANAG) 4586 [3] defines requirements for a standard set of UCS interfaces. It has been developed over the last decade to promote interoperability among UAS manufacturers and coalition partners. Consistent with the STANAG 4586 functional UAS Architecture, figure 2 illustrates the four primary sets of UCS interfaces: (1) Data Link Interface (DLI); (2) Command and Control Interface (CCI); (3) Human-Computer Interface (HCI); and (4) a set of alternate/complementary communication interfaces providing capabilities such as radio communications and Internet Relay Chat (IRC).

STANAG 4586 specifies that the CCI shall support a subset of standardized tactical messages formats used by participating nations: US Message Text Format (USMTF), NATO Allied Data Publication 3 (ADatP-3) and Over-The-Horizon-GOLD (OTH-GOLD).

The HCI, which is the primary focus of this paper, allows VO and MPO to exercise low-level and high-level control of the Air Vehicle (AV) and the payloads.

This paper assumes that introducing intelligence into the operator interfaces will most likely involve the use of *intelligent agents.* It is also assumed that the proper and efficient use of agent-based technologies requires well-defined protocols, i.e. standard machine interfaces and message structures. The present study discusses the benefits associated with the use of formal languages for the communication of military information to standardized automation elements based on intelligent agents so that they can be introduced in the HCI to improve operator effectiveness. In this regard, the following concluding statement from reference [4] provides the basis for this paper:

> *"...The design of an autonomous UAS depends not only on the addition of "smart" technologies but equally on the HCI and the nature, timeliness and relevance of the information presented to the operator together with the level of control afforded over the capability."*

This statement also is supported by the mission-centric philosophy of current design efforts for UAS operator interfaces that increasingly require greater levels of UAV autonomy as perceived by the VO and MPO [5].

1.3. C-BML as a Formal Language in support of Intelligent Operator Interfaces

This paper proposes a technical approach to support the experimentation of new UAS concepts of employment in a simulation environment using the Coalition Battle Management Language (C-BML), a formal language, as an enabling technology for the interoperation of simulation systems, C2 systems and UVS. This approach describes an experimentation capability that could be used to explore concepts for research, design, and rapid prototyping of next-generation UAV operator interfaces and involves the development of a simulation environment where real world C2 systems can interoperate with some of the simulated UAS subsystems using C-BML. The *intelligence* is introduced into the operator interfaces by applying automation management strategies, combined with the use of a formal language for effectively supporting automated information exchange between the Unmanned Aerial Vehicle (UAV) operator and the UAS subsystems.

In the remainder of this paper we discuss some of the identified gaps and requirements related to future UAS capabilities in Section 2, and then introduce the notions of Autonomy and Automation with the various automation management strategies in Section 3. A discussion follows in Section 4 on the employment of intelligent systems in order to increase UAS autonomy. Thereafter, Section 5 is dedicated to C-BML and to a discussion as to its relevance for UAS operations. Finally, we conclude this paper in Section 6 and discuss potential future work.

2. UAS Capability Gaps and Current Issues

This section presents UAS requirements based on capabilities for future UAS employment and specifically highlights areas of interest that might benefit from the introduction of additional automation in UAS, and specifically in the HCI utilized by UAV operators.

2.1. Greater Autonomy of UAS

Increasing UAV-platform and UAS autonomy is an underlying and cross-cutting theme that touches upon many of the aspects of current and future UAS operations [2]. Mission requirements call for AV to be able to accomplish missions even in the case of a momentary or permanent communications failure between the UCS and the AV, or during a transfer of control from one UCS to another. This capability is already available in some UAS, such as the Fire Scout, manufactured by Northrup Grumman, which has the capability to receive and automatically execute a flight plan that is uploaded prior to take-off, without subsequent operator intervention.

2.2. UAS Operations Agility

Agility is essentially the ability of friendly forces to act faster than enemy forces. This means that commanders may need to act without the luxury of waiting for complete information and that tasking and re-tasking may be performed in a dynamic context. Joint, Inter-Agency, Inter-governmental, Multi-national (JIIM) operations also impose time constraints associated with the coordination and synchronization of activities with other forces and agencies. From a commander's perspective, the ability to task and dynamically re-task complex systems, such as UAS to meet the changing mission objectives of a dynamic battlespace provides flexibility required to achieve mission goals in a timely manner.

2.2.1. Dynamic Command and Control

In a dynamic battlespace, concepts, such as *Integrated Dynamic Command and Control* (IDC2) call for coordination of tactical elements at all levels, be it within a given service, across services or in a multinational context. One of the keys challenges to achieving the required coordination is the synchronization of C2 activities in a way that minimises time delays within and between command levels [6]. This may include the ability to update and communicate information such as Rules of Engagement (ROE) and commander's intent at rates faster than occurring in traditional operations and which conceivably, could evolve during mission execution [7]. This study assumes that future UAS operations likely will utilize digital, machine-consumable representations of information, such as ROE, as inputs to decision-making UAS subsystems in support of concepts such as IDC2.

In fact, the Joint Consultation Command and Control Information Exchange Data Model (JC3IEDM) [8], discussed more in detail below, defines information elements for the ROEs but they are specified in free-text format and thus currently are not machine-consumable. The issue then becomes how to represent this information in a form that can be processed by machines for activities such as decision-support.

2.2.2. UAV Dynamic Re-tasking

Dynamic re-tasking occurs following changes in mission objectives, timings or mission routes during mission execution. Often involving vehicle re-routing through controlled airspace, mission planners must consider parameters such as current vehicle operating limits and weather and terrain conditions, while contending with a potentially hostile and changing environment, often under time constraints. In many instances, one of the most significant challenges associated with dynamic re-tasking of UAVs is airspace deconfliction.

2.2.3. Airspace Deconfliction

As unmanned AV become more numerous, airspace deconfliction will require increasing resources. From a VO perspective, disposing of 3-D graphical views of the controlled airspace has been shown to facilitate the task of re-routing [1]. The Joint Air Space Management And Deconfliction (JASMAD) project [9] aims to optimize the use of airspace through the introduction of dynamic airspace reallocation involving an increased situational awareness with enhanced graphical displays. This capability calls for a real-time position, course and speed of all aircraft. JASMAD also addresses airspace deconfliction requirements associated with Time-Sensitive Targeting (TST) involving UAS.

2.3. Operator Workload Reduction

UAV information overload is becoming a problem for many humans and machines in the UAS information loop [10]. In particular, UAV operator cognitive overload comes from several sources, including information from the AV (e.g. navigation, health system management) and sensors [4]. Moreover, the required level of detail of the VO situational awareness increases with the operator requirement to execute lower levels tasks. Therefore, higher levels of AV autonomy translate into a decrease in operator workload through the introduction of automation that allows the operator to execute primarily higher level control (i.e. human supervisory control).

Paramsuraman et al. [11] have developed an automation model for Human Interaction based on decision-making functional areas: *acquisition*, *analysis*, *decision-making* and *action implementation*. Each of these functional areas can be supported through automation and are used in the discussion below.

Increasing the level of control that operators exercise requires decision-making intelligence to be built into either: (1) the AV, (2) the UCS, or (3) both the UCS and the AV. Advances in AV platform autonomy have sparked interest in extended message sets for communication between the UCS and the AV, which allows the AV to complete critical tasks in the context of unplanned mission-critical events, such as: critical fault management, collision avoidance and sudden changes in weather (e.g. adverse winds, temperatures beyond operating range, etc.). In the case that the UAV platform only executes low-level control messages, it still is possible to expose higher-level control functionality at the operator interface through the introduction of intelligence in this interface – thus forming the basis for this study. Nonetheless, this greatly limits the operational capability during a communication disturbance between the AV and the GCS.

2.3.1. AV Status Monitoring

As per [12], UAV operator monitoring functions include monitoring: payload status, network communications, system health status, and sensor activity. Effective monitoring requires mechanisms for prioritizing, notification and communication to the operator through aural and visual cueing.

2.3.2. Communication with Stakeholders

Communication with stakeholders can take place using formatted text messages (FTM), voice communications or chat. In addition to standard reporting using FTM (e.g. status reports, situation reports, intelligence reports and battle damage assessment (BDA), etc.), UAV operators also are required to use voice and chat to coordinate with stakeholders that are external to the UAS for activities such as: authorization of requests (e.g. fires support, airspace coordination), notification to ACA of airspace use (or non-use) and coordination with ground forces (e.g. Close Air Support (CAS)). Two areas of particular interest with respect to communication with stakeholders are: (1) the extensive use of chat in UAS operations and, (2) the benefits of automatic reporting.

2.3.3. On the use of Chat in UAS Operations

The use of chat as a mission essential C2 tool to support real-time multi-user collaborative communication for military operations has been confirmed during recent conflicts in Iraq and Afghanistan [13]. Chat is equally used in both military and civil applications, and chat technologies have also played an important role in antiterrorism, homeland defence and disaster relief efforts. However, the extensive use of chat systems, such as multi-user Internet Relay Chat (mIRC) has unveiled chat-specific interoperability issues, such as the use of incompatible systems by partners who could not communicate in the context of coalition military operations or multinational disaster relief efforts [13].

The use of chat for UAS operations has provided for an invaluable, direct communication link between the supported unit (e.g. Close Air Support, Direct Support) and vehicle and payload operators. Targeting officers, Forward Air Controllers, Air Component Commanders can communicate in parallel with UAV operators for missions requiring real-time collaboration, such as close air support involving time-sensitive targeting (TST). Chat has also been utilized for CAS and Joint Fires Support (JFS) deconfliction, to task UAVs directly, to allow UAV operators to coordinate with the ACA, for monitoring purposes, during Medical Evacuation (MEDEVAC), and for communicating Meteorological and Oceanographic (METOC) forecasting support.

Perhaps the most significant negative aspect of chat is that it is not integrated into current C2 infrastructures and therefore represents a "parallel" channel. This creates an interoperability gap, as witnessed by the presence of a separate interface for UAV operators. This has led to situations where an over-reliance on chat interfaces resulted in: (1) operators heavily focused on chat had a tendency to miss important cues from their primary interface and (2) units not equipped with chat capabilities did not receive important tactical information that was communicated solely through chat.

In terms of autonomous UAS operations, if automation is to be leveraged as a means to achieve greater operations agility by streamlining military business processes and workflows associated with the command and control of unmanned assets, then information that is currently flowing through chat channels will need to be made available to machines, in addition to and, in some instances, in the place of humans. As suggested by Eovito [13], of primary importance is to clearly identify and analyze the requirements that are currently being satisfied by chat in a top-down approach. Only afterwards will it be possible to determine, in the context of intelligent systems and future concepts of employment, how these requirements can best be met.

2.3.4. Automatic Reporting

The ability for UAV operators and imagery analysts to generate and communicate reports effectively is obviously critical to mission success. The ability to partially or fully automate report generation and subsequent dissemination is consistent with the general vision for net-centric operations. The fully automated generation and dissemination of certain reports, such as task status reports, will undoubtedly be easier to achieve than those requiring more complex workflows such as enemy situation reports that require additional analysis. Nonetheless, virtually all reporting workflows can benefit from the introduction of automated processes.

2.3.5. Multi-UAV, Single-Operator Control

UAV are increasingly replacing fixed or rotary wing piloted aircraft, and are being used simultaneously in various roles and mission types. Human and machine resource limitations are driving the requirement for developing operator interfaces that would allow a single operator to control several AV. Cummings *et al* [14] propose an architecture to support human supervisory control of multiple UAV by a single operator. A pre-requisite to multiple UAV single-operator control is, of course, the ability for the operator to exercise HSC without having to address lower-level tasks.

2.4. The case for Intelligent UAV Operator Interfaces

While long-term requirements for future autonomous UAS may involve operations with limited or even no UAV operators in-the-loop [2], technical, legal, social and other considerations confirm that UAV operators will be required for quite some time to come. Furthermore, in light of the requirements and issues highlighted above, these operators will require enhanced interfaces with built-in information management and decision-making capabilities.

Intelligent operator interfaces are in a sense a disruptive technology and will impact not only the operator procedures, but will also impact procedures of external UAS stakeholders and possibly even the doctrine for autonomous UAS operations.

The design of these interfaces will require collaboration and input from areas such as: human factors, behavioural psychology, control theory, military and civil law and others. As a consequence, the development of next-generation systems likely will be iterative and will benefit from experimentation platforms that leverage simulation technologies and that can assist in validating design approaches and verifying critical assumptions.

The current study originates from preliminary work involving experimentation performed using actual C2IS and a simulated UAS [30] [32]. This work leveraged the emerging C-BML standard in conjunction with the use of intelligent UAV operator software agents for the automated command and control of the UAV asset. Based on this work, this paper considers how similar experimentation capabilities can be useful in the design of intelligent operator interfaces. In addition to helping address challenges associated with the design process itself, experimentation capabilities also may prove useful in the development of future revisions of the governing standards, namely STANAG 4586.

The remainder of this paper considers the issues associated with designing intelligent operator interfaces and the impact on the interoperability standards. Before considering UAV operator interface design issues, the following sections provide a short description of terms in the area of automation, autonomy and intelligent systems.

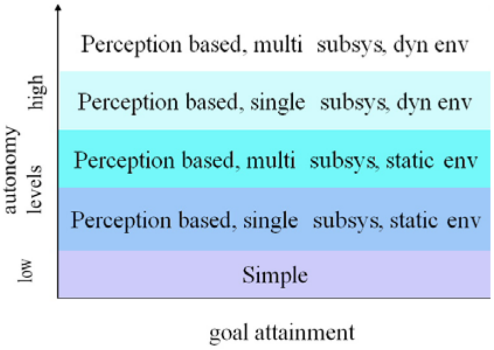


Figure 3 – Levels of Autonomy (taken from [16])

3. Automation & Autonomy

Before addressing automation requirements for intelligent operator interfaces, the following section provides a brief summary of relevant definitions and references for automation, autonomy and intelligent systems.

A system exhibits *autonomy* when it is capable of making - and is entrusted to make - substantial real-time decisions, without human involvement or supervision [1]. Autonomy implies the ability to act independently. However, a system’s levels of autonomy can only be defined with respect to a specific set of goals or functions. Shown in figure 3 and taken from reference [16], the *Autonomy Levels For Unmanned Systems* (ALFUS) framework defines levels of autonomy based on factors related to the system’s ability to: (1) achieve a set of prescribed objectives, (2) adapt to major changes, and (3) to develop its own objectives (i.e. the ability to learn and store/use knowledge). An important aspect of autonomy associated with this framework is the ability of subsystems to collaborate in the context of a changing environment.

Automation has many definitions, but for the intents and purposes of this study, it refers to the use of machines to execute functions that would otherwise be performed by human operators. Automation enables autonomy. Reference [11] proposes a model for representing different levels of human interaction with automation that is helpful to characterize different types of human-machine interactions with varying degrees of responsibility entrusted to the machine. Although this scale, shown in table 1, does not apply to all automation scenarios, it is particularly useful for the analysis of the implications of introducing varying levels of automation into workflows independent of the domain of application.

As part of this model, four classes of functions are defined for areas corresponding to the areas of human information: (1) information acquisition; (2) information analysis; (3) decision-making; (4) action implementation. Figure 4 illustrates a means for capturing the levels of automation applied to these functional areas, where the numbered circles correspond to the levels of automation described in Table 1.

Table 1- Levels of Automation [11]

Level	Automation Description
1	The computer offers no assistance: human must take all decision and actions.
2	The computer offers a complete set of decision/action alternatives, or
3	narrows the selection down to a few, or
4	suggests one alternative, and
5	executes that suggestion if the human approves, or
6	allows the human a restricted time to veto before automatic execution, or
7	executes automatically, then necessarily informs humans, and
8	informs the human only if asked, or
9	informs the human only if it, the computer, decides to.
10	The computer decides everything and acts autonomously, ignoring the human.

The inputs to the workflow are part of the information acquisition functional area where as the outputs are the implemented actions resulting from an action selection or decision-making process. In the case of UAS operations, the action selection could represent navigation or mission payload commands or the generation and communication of a report. A fully manual workflow is represented by a point in the center of the chart while a fully automated workflow would be represented by the blue line passing through the outer perimeter, as shown in figure 5. Although the latter case implies no human involvement, it is useful to consider some of the implications of such a workflow. The areas A, B and C can be considered as specific areas of interest wherein: (A) processing of inputs to support analysis; (B) transformation of analyses results into possible actions; and (C) generation of outputs based on action selection.

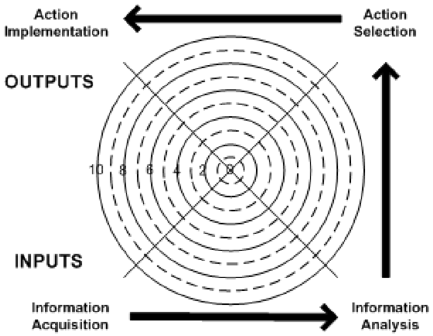


Figure 4 – Automation-enabled processes

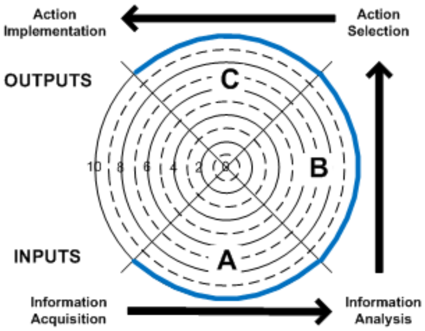


Figure 5 – Fully-automated workflow

Graphical representation issues and operator information overload issues are dealt with in area A. Concerns for area B include determining the validity of information such as the predictions and other analysis results while considering contextual information as well as information based on previous experience. Area C determines to what extent systems can decide and act independently. Of special significance in area C are the legal, safety and social implications of allowing machines to operate in this area at high levels of automation. For instance, currently, there is still much resistance to the concept of machines automatically detecting and engaging targets [17]. Moreover, the legal considerations of such automated tasking raise a number of questions that likely will require considerable reworking of the modern Law of War.

3.1. Automation Management Strategies

Higher levels of autonomy require proper automation management strategies in order to effectively lessen the operator load while avoiding automation-related side effects such as: *automation bias*, *mode confusion* and *reduced situation awareness* [4]. Consistent with [4][18], the following categories of Automation

Management Strategies (AMS), shown in table 2, can be defined: human-based, management-by-consent (MBC), management-by-exception (BME) and machine-based.

Table 2- Automation Management Strategies

	Automation Mgt Strategy		LOA
A	Human-based	operator must perform actions and tasks	Level 1
B	Management-by-consent	requires operator approval for task execution	Level 5
C	Management-by-exception	requires operator override or task will be executed automatically	Level 6
D	Machine-based	tasks are executed automatically	Levels 7 to 10

These strategies are useful as guidelines in the analysis of military enterprise processes and workflows and are used below, in the example use-case considered in this study.

4. Increasing Autonomy and the use of Intelligent Systems

Increasing the levels of autonomy of complex systems such as UAS requires automation which must be introduced with great care. For example, automation management strategies must be developed and refined such that the advantages associated with the utilization of machine-based *intelligent* systems, systems capable of making decisions, are not outweighed by potential negative side effects, such as *unintentional workload increase*, *reduced situational awareness*, *automation bias* and *skill degradation* [11]. In other situations, such as in the case of operator intervention associated with a change in system automation mode, there is also a risk of *mode confusion* [4] that has led in the past to the loss of aircraft.

Intelligent system design generally involves the use of autonomous software components know as *software agents*. The use of Agent-Based Modeling (ABM) also known as Multi-Agent Systems (MAS) relies on the availability of information in a machine computable form and therefore these areas are closely tied to the field of knowledge representation, which is central to intelligent systems, as discussed below.

4.1. Intelligent Adaptive Systems

Intelligent Adaptive Systems (IAS) and Intelligent Adaptive Interfaces (IAI) are able to configure themselves automatically based on contextual information in the form of internal or external triggers allowing them to operate in an optimal manner as part of a system of systems or in conjunction with a human-in-the-loop [19]. Intelligent *adaptive* systems are able to modify their automation mechanisms based on context-dependent information, such as system health status, threat-levels and operator fatigue. Another important aspect of intelligent systems is their ability to learn, store and re-use knowledge based on previous execution.

The present paper does not consider the internal design of agents, (see for example reference [19][20][21]). However, common to all agent-based design approaches (whether intelligent systems are adaptive or not) are the primary requirements for establishing the appropriate and necessary languages and protocols for representing domain knowledge in a form suitable for use by agents and provides the necessary support for communication among the agents.

4.2. Formal Knowledge Representation of Military Information

Fortunately, over the last decade, much progress has been made in the area of the formal knowledge representation in the military domain in support of such information exchange requirements to enable interoperability among system of system architectures, and more recently to support more efficient information sharing in the context of net-enabled and net-centric capabilities. Of primary importance becomes the ability to capture and share relevant, useful and current data and information in a standardized machine-consumable format so that it can be made readily available for use by other systems.

It is possible to identify an evolution in the format of electronic formats for military information such as *orders* and *reports* over the past 30 years or so that is consistent with the parallel evolution of devices used by the armed forces to communicate this information. For instance, standards developed during the 1980s and 1990s employing military Message Text Formats (MTF), like the Allied Data Publication 3 (ADatP-3), often were developed with the teletypewriter as the intended terminal device [22]. Over the last decade, the transition to XML formats has become commonplace, be it as a means to support Web Service Definition Language (WSDL) compliant payloads or as part of an overall standardization strategy [23]. More recently, many efforts have been working toward the development of ontology-based knowledge representations that will support requirements for military net-centric information sharing [24][25][26][27].

The Joint Consultation Command and Control Information Exchange Data Model (JC3IEDM) that has been developed over the last decade by the Multilateral Interoperability Programme (MIP) – initially formed by 18 nations, currently counting 28 nations, and is one of the most extensive and widely employed military IEDM. Nearly all of the abovementioned efforts working on ontology representations for the military domain utilize the JC3IEDM as the underlying model.

Future C2IS likely will utilize tactical messages based on a formal knowledge representation and consequently, the UCS interface for the exchange of messages between the UAS and C2IS, the CCI, will need to evolve to support these message sets.

4.3. Intelligent Agent Communication

In addition to a formal knowledge representation, the successful use of agent-based approaches also requires satisfying interface requirements for communication among agents. For example, simple Web Service status codes are not sufficient to capture the possible outcomes of an agent-to-agent interaction. Specifications such as the IEEE Agent Communications Language (ACL) developed by the Foundation for Intelligent Physical Agents (FIPA) proposes a standard set of protocols for communication with agent-based systems [28]. This standard specifies twenty-two *communicative acts* based on various interactions, including: accept/refuse, confirm/disconfirm, subscribe/inform, call for proposal, accept proposal, reject proposal, propose, propagate, failure, etc.

This richer set communication interactions would be useful, for example, in addressing shortcomings in the failure codes between AV and the UCS where, as pointed out by [4], operators can receive failure messages that only indicate that a failure has occurred and do not provide enough information to take corrective measures.

Some Agent-based software development frameworks such as the open-source Java Agent DEvelopment (JADE) framework comply with the FIPA specification and therefore allow for FIPA-compliant agents to communicate with each other.

4.4. Intelligent Agents and Net-Centricity Requirements for STANAG 4586

By many measures, the STANAG 4586 Standard for the interoperability of UCS interfaces can be considered a success in promoting re-use of system hardware and software components and fostering collaboration among coalition partners. Looking toward the future, the STANAG 4586 Custodial Support Team (CST) also has identified several focus areas to be addressed in future blocks, including: (1) the need for the UCS to be able to exercise higher-level control over AV exhibiting greater autonomy and (2) the capability to integrate the UCS as one system in a system of net-centric systems.

Concerning the first goal, although the initial intent of STANAG 4586 was to provide both lower-level control and higher-level control (aka HSC) of UV platforms by operators, the focus thus far primarily has been on lower-level control [4]. However, initiatives are planned for defining extensions to the DLI to provide for the communication of additional information between the UCS and the VSM, as required to support higher-level control.

Toward this second goal, the STANAG 4586 CST has formed the STANAG 4586 NNEC/SOA Working Group to address how requirements for the use of Web technologies might best be integrated into future blocks of this standard. This working group is currently addressing net-centricity requirements through the specification of a set of Web Services that would be exposed by the UCS CCI [29]. These services include:

- Track (AV Status and Position);
- Asset Registration;
- Sensor Observation;
- Sensor Planning
- AV Route Planning;
- Motion Imagery;
- Still Imagery;
- GMTI Data; and
- ADatP-3 Messaging.

The current authors suggest that although the information exchange requirements for many of the above services are satisfied by existing standards and have already been defined in sufficient detail, some of the services warrant further analysis to determine if a more formal representation is required. For example, the ADatP-3 message and AV Route planning services are excellent candidates for intelligent agent-based processing. This also has great implications on the C2 systems that are communicating ADatP-3 and similar messages to the UAS. This is not addressed in this paper. Also, the question arises as to whether Web Services technologies such as UDDI, WSDL and SOAP technologies for discovery, binding and messaging, respectively, provide sufficient expressiveness for describing services for subsequent processing by software agents [27].

Exposing services as Semantic Web Services is one means of addressing expressiveness gaps such that messages can be formulated using a representation that can be interpreted by machines. This implies potentially developing more formal representations of existing standards. The validity of this approach is perhaps confirmed by parallel efforts to generate ontologies for the JC3IEDM, (see for example [24][25]).

4.5. On the Use of Formal Languages to Support Intelligent Operator Interface Requirements

As per reference [4], future UAV operator interfaces must incorporate increased intelligence to support operator needs. In addition, the authors of this study suggest that in order to support automation requirements, the UCS information exchange requirements may need to be extended to include the use of formal language to ensure that *intelligent* capabilities are, in fact, usable and useful.

Currently there are several initiatives to create formal language based representations of military information such as orders, reports and requests by the operational C2 community. This study considers how the concepts and/or actual components of one such language developed by the Modelling and Simulation community, the Coalition Battle Management Language (C-BML), initiated nearly a decade ago [37], can be leveraged for the purposes of exploring how the use of formal languages will contribute to satisfying requirements for enhanced automation support associated with the development of intelligent operator interfaces.

4.6. Intelligent System Summary

This section has provided a brief overview concerning the use of intelligent systems for use as part of enhanced UAV operator interfaces. The argument has been made that it will be necessary to introduce higher levels of automation into the UCS HCI in order to support agile UAS operations while addressing operator cognitive overload issues and possible additional operator tasks such as those required for multi-UAV control. It also has been suggested that additional automation likely will be in the form of agent-based intelligent systems.

Furthermore, it has been suggested that the successful integration of intelligent systems requires both a formal knowledge representation and specific languages and protocols to support the collaboration and communication among agents. C-BML is one such language that meets some of the requirements and may be used, in part or in whole, as input into future UAS standardization initiatives aimed at supporting automation requirements for autonomous UAS operations.

5. Using C-BML to Support UAS Automation Requirements

C-BML is currently being developed by the Simulation Interoperability Standards Organization (SISO) as an unambiguous, machine-computable language for the communication of tactical military information such as *orders*, *reports* and *requests* among C2, simulation and autonomous systems. Early experimentation using preliminary versions of C-BML has shown encouraging results concerning the use of C-BML for concept exploring involving the tasking of UAS by C2 systems and also for automatic reporting from the UAS to the C2 system [30][31][32].

Introducing automation into the UCS can help to resolve many information management issues, including operator overload. However, while automation is certainly a part of the solution, there is still a need for operators in loop for some time to come. The key is to assist operators through the elaboration of intelligent operator interfaces. These augmented interfaces implement various automation management strategies that are required to automatically perform some of tasks for operators while simplifying other tasks. Decreasing the operator cognitive load and freeing up operators to perform high priority tasks while resulting in less human induced-errors.

As an ontology-based formal language, C-BML can link C2IS, simulation systems and autonomous systems and may prove useful in the development of future revisions of UAS interoperability standards, such as STANAG 4586.

5.1. C-BML Overview

C-BML is an XML-based formal language for exchanging military orders, reports and requests among C2, simulation and autonomous systems. Reference [33] presents C-BML in terms of the following characteristics which are summarized as follows:

- *Expressive and precise*: a set of unambiguous valid expressions (i.e. based on a formal grammar or production rules),
- *Computable*: military information that can be parsed, validated and processed in an unique manner based on a common reference model (i.e. semantic interoperability),
- *Understandable*: expressions that can be interpreted by the consumer as intended by the producer (i.e. pragmatic interoperability [34],
- *Multi-doctrine*: is not tied to any specific doctrine (i.e. doctrine-agnostic), but supports NATO and national doctrines,
- *Multi-domain*: BML should support air, maritime, land and joint operations,
- Information Exchange Mechanism (IEM) independent: should not be tied to any one IEM and
- *Standardized*: should be an international standard to promote interoperability within and across national systems.

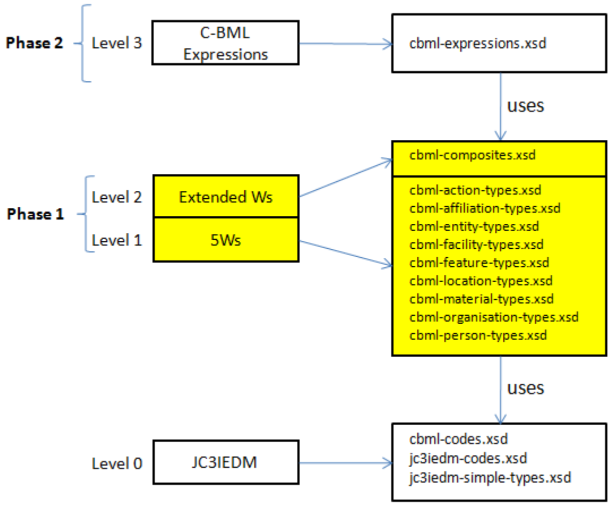


Figure 6- C-BML Layers

Many of these characteristics collectively can be found in message formats and protocols that were discussed in the previous sections. C-BML is being developed in three phases that are divided as follows: (1) Data Model; (2) Grammar; and (3) Ontology. Currently phase 2 efforts are considering ontology representations to capture the grammar or *production rules* that allow for the construction of valid C-BML expressions. The JC3IEDM is the underlying model upon which the phase 1 C-BML model has been developed, as shown in figure 6.

5.2. JC3IEDM and C-BML

The JC3IEDM is specifically designed to quantify information related to the conduct of war. The JC3IEDM is a successor of a long line of military data models that have been developed by the MIP for over twelve years and is now released as NATO STANAG 5525.

The fundamental building blocks of C-BML, often referred to as the 5 Ws (*Who*, *What*, *When*, *Where*, and *Why*) are defined in the foundational work on the Command and Control Lexical Grammar (C2LG) [35], and are relatively well represented in JC3IEDM. For instance, the *Who* can be represented by an *ObjectItem* and can be attributed a unique *ObjectIdentifierDigit*. The *What* can be represented as an *ActivityCode*, *EventCode* or *EffectCode*, and the *When* can be represented as a *date-time* group or as a *TemporalAssociation*. The *Where* can be equated to a detailed *Location* and the *Why* can be expressed as a *FunctionalAssociation* to another task or as a desired effect. However, the JC3IEDM covers a broader set of requirements than C-BML and a large portion of JC3IEDM is not required in order to convey an order, a report or a request.

However, the JC3IEDM was not intended to be utilized as a formal language and it cannot be assumed that JC3IEDM information elements are adequate or sufficient for machine to machine communication. Thus, C-BML aims to leverage the richness of the JC3IEDM within the expressiveness and capacity for automation of a formal language.

5.3. Grammar

While the C-BML data model essentially provides the vocabulary, the C-BML grammar is comprised of the production rules that constitute the set of valid C-BML expressions. Composites are logical groupings of basic information elements, based largely on the 5 Ws that form the basis for constructing expressions such as reports or orders.

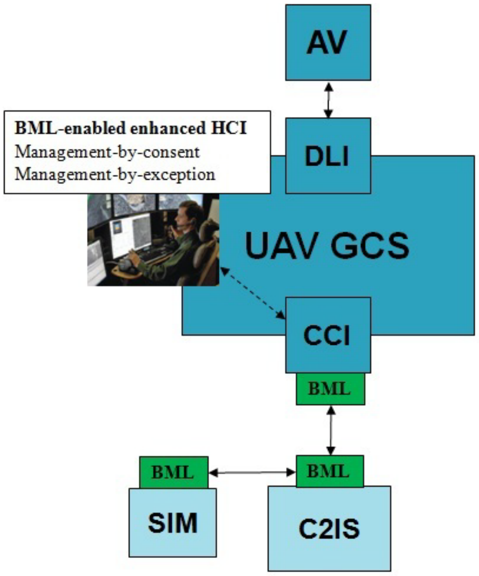


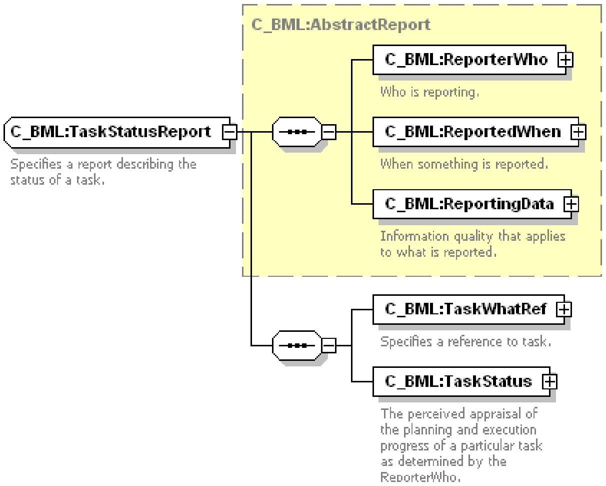
Figure 7 – UAS Experimentation Capability

5.4. BML-Enabled UAS Experimentation

Figure 7 depicts a BML-enabled UAS experimentation capability similar to the one described in [30][31][32]. A BML interface acts as the common communication link between C2 and simulation systems and between C2IS and the UCS. The output of existing C2IS tasking (e.g. ADatP-3) readily can be translated into BML messages while maintaining the possibility to add additional information, such as rules of engagement and command intent, for potential use by intelligent agents within the UCS. Similarly, the enemy and friendly force situations and tasking can be communicated to the simulation for execution. Not shown in the diagram is the possibility to link sensor emulations within the UCS to the simulation such that the operator interacts with the virtual battlespace – thus closing the loop on the experimentation. In this manner, various algorithms for automation management strategies can be validated with the operator in the loop while maintaining the possibility to revert to traditional operations wherein the BML messages contain no additional information.

5.5. C-BML Example

The C-BML model is currently expressed as an XML Schema Description (XSD) document, and consequently, for simplicity the example below is based on schema screenshots. Figure 8 depicts a relatively simple C-BML expression: a task status report. Obviously a necessary element of a task status report is the task status shown in figure 9. This example illustrates the basis structure of a C-BML expression. A task status report is comprised of three mandatory elements: (1) a reporting who; (2) a reported when; (3) reporting data; (4) a task reference and (5) a task status. Note that the reporting data, as per the JC3IEDM, represents the pedigree of the information and can include information such as the accuracy, credibility and reliability of the information contained in the report. The task status is comprised of four mandatory information elements: (1) an identifier; (2) a category code; (3) a completion ratio; and (4) a planning indicator code. There are also four optional elements: (1) a progress code; (2) an amend timing code; (3) an approval indicator code; and (4) a feint indicator code. In addition to the readily evident mapping between the C-BML OID and a JC3IEDM OID, the task status category code can also be mapped to a JC3IEDM Action-Category-Code.



C_BML:TaskStatusReport
Specifies a report describing the status of a task.

C_BML:ReporterWho
Who is reporting.

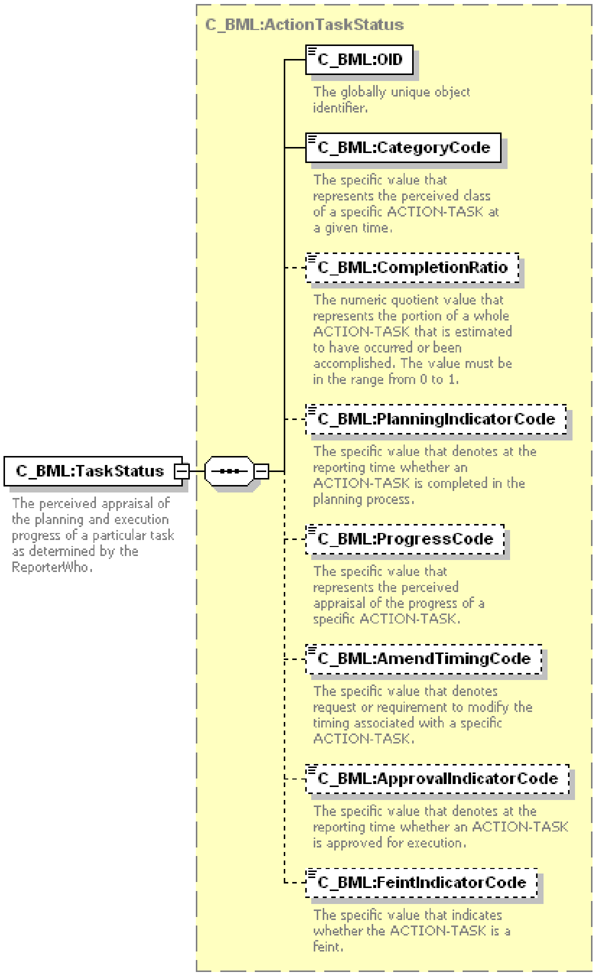
C_BML:ReportedWhen
When something is reported.

C_BML:ReportingData
Information quality that applies to what is reported.

C_BML:TaskWhatRef
Specifies a reference to task.

C_BML:TaskStatus
The perceived appraisal of the planning and execution progress of a particular task as determined by the ReporterWho.

Figure 9- C-BML Example, Task Status Report



C_BML:TaskStatus
The perceived appraisal of the planning and execution progress of a particular task as determined by the ReporterWho.

C_BML:OID
The globally unique object identifier.

C_BML:CategoryCode
The specific value that represents the perceived class of a specific ACTION-TASK at a given time.

C_BML:CompletionRatio
The numeric quotient value that represents the portion of a whole ACTION-TASK that is estimated to have occurred or been accomplished. The value must be in the range from 0 to 1.

C_BML:PlanningIndicatorCode
The specific value that denotes at the reporting time whether an ACTION-TASK is completed in the planning process.

C_BML:ProgressCode
The specific value that represents the perceived appraisal of the progress of a specific ACTION-TASK.

C_BML:AmendTimingCode
The specific value that denotes request or requirement to modify the timing associated with a specific ACTION-TASK.

C_BML:ApprovalIndicatorCode
The specific value that denotes at the reporting time whether an ACTION-TASK is approved for execution.

C_BML:FeintIndicatorCode
The specific value that indicates whether the ACTION-TASK is a feint.

Figure 8 – C-BML Example Task Status

5.6. Example Use Case: Dynamic Re-Tasking

In the case of dynamic re-tasking, operators are confronted with significant challenges in determining alternate routes based on a changing battlespace, including: enemy situation on ground, weather and terrain factors, airspace restrictions and changing mission objectives. Re-crafting the ATO in coordination with the ACA during the UAV mission is therefore a high-pressure, time-constrained activity for which UAV operators would benefit from automation aids such as Path Planning Algorithms (PPA) and enhanced 3-D visual displays [1].

Reference [1] proposes a taxonomy of re-routing event triggers that includes:

- New/change to target tracking requirements;

- Change in airspace availability;
- Aircraft avoidance;
- Counter-detection;
- Weather avoidance; and
- Terrain.

Figure 10 suggests how a UAV dynamic re-tasking workflow can be expressed in terms of the workflow functional areas and automation management strategies described in section 3.1.

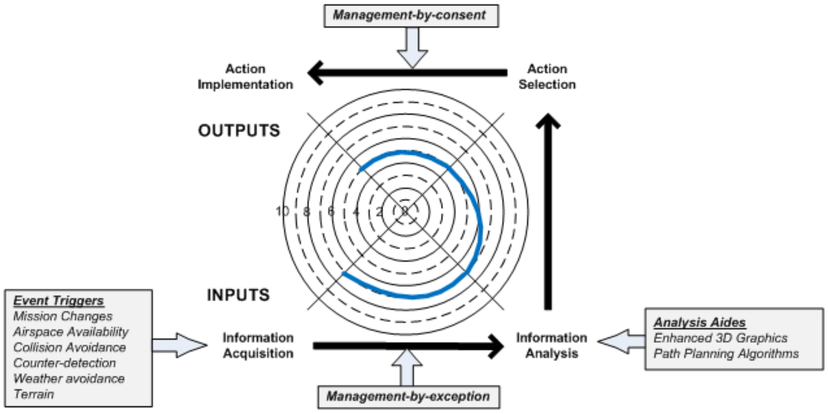


Figure 10 – Notional Dynamic Re-tasking Automation Strategy

Handling of event triggers can be performed automatically while ensuring that operators maintain awareness of these events using visual and aural cueing adapted to parameters such as mission-specific contextual information and operator-specific information (e.g. experience, fatigue level etc.). As this information is further processed and analyzed, results may be presented to the operator for further consideration (e.g. using enhanced 3-D graphics) and communicated to decision-support components for activities such as automatic path planning. Employing the management-by-exception AMS, the flow of information toward the information analysis process is uninterrupted, unless the operator decides to intervene. The results of decision-making activities are then made available for *action selection* and subsequent *action implementation* functions. However, in the case of dynamic re-tasking, human approval is required, corresponding to level 5 automation and a management-by-consent AMS that is appropriate when action implementation has legal and/or safety implications.

5.7. Additional Information Exchange Requirements for Automated Operator Workflows

5.7.1. Real-time collaboration

For at least the short-term, chat undoubtedly will be utilized as the primary means for real-time collaboration among UAS stakeholders. However, it is necessary to study how chat might be integrated in a manner that would not present air gaps.

5.7.2. Notification

The information exchange requirements to support the semi-automated workflow example use-case illustrated require that the information associated with the event triggers be acquired and processed automatically. Notification mechanisms such as the *smart-push* have been advocated by efforts such as the Valued Information at the Right Time (VIRT) approach [36], wherein conditions of interest are expressed by interested systems. The ability to specify and verify conditions of interest also supports requirements for a formal language representation consistent with those used to satisfy Semantic Web Service requirements.

6. Conclusions and Future Work

In light of the increasing employment of UAS and UAS-related capabilities, there is a clear requirement for additional autonomy in unmanned platforms and unmanned systems. At the same time, UAV operators are being exposed to greater workloads and are placed in situations of cognitive overload. In parallel, the need for faster response time to changing mission requirements such as dynamic re-tasking results in even greater demands on operators.

This study has illustrated how introducing automation into the UCS can help resolve many information management issues, including operator overload. However, while automation is certainly a part of the solution, there is still a need for operators-in-the-loop for some time to come, due to technical challenges but also in light of political, legal and ethical considerations. The suggested approach is to assist the operators with intelligent interfaces. These augmented interfaces implement various automation management strategies that are required to automatically perform some of the tasks while simplifying others, hence decreasing the operator cognitive load and freeing up operators to perform high priority tasks while resulting in less human induced-errors.

At the same time, increasing the net-centricity of UAS also has been identified as a key requirement. Toward that goal, standardized, machine-computable communication mechanisms must be put into place. Chat is a proven and extremely useful and highly valued means of communications in support of UAS operations, yet presents several interoperability barriers. But, how will the highly utilized chat capability be transformed into readily usable, net-centric, alternative? A challenge that arises is formalizing the requirements behind chat's success and then utilizing them as inputs in the development of future interoperability standards for C2IS and UCS. Furthermore, this approach may provide the basis for a transition from *legacy* chat to next-generation *net-centric* chat, with the latter possibly being integrated within C2IS and UAV operator interfaces.

The introduction of intelligent operator interfaces for autonomous UAS could be considered as a disruptive technology and will have far-reaching implications in terms of both technical challenges and the evolution of operational procedures. Research, analysis and experimentation are required to assist in the development of: (1) new concepts of operation, (2) prototypes of next-generation intelligent operator interfaces and (3) new and revised interoperability standards. We have demonstrated how the BML technology is well-suited for use in the UAS experimentation capabilities that could support such development efforts. Furthermore, a simulation-based, BML-enabled experimentation capability involving C2IS, UCS and UAV operators that communicate with relevant stakeholders has already proven useful in the understanding and demonstration of these new concepts, and will undoubtedly be utilized as the simulation testbed for the development of future operational capabilities. This same experimentation capability also could very well support the development, verification and validation of requirements and approaches for future revisions of standards such as STANAG 4586.

7. References

[1] M. Cook, H. Smallman, "When Plans Change: Task Analysis with Navy UAV Operators, Display Requirements, and UAV Re-routing Taxonomy", Pacific Science & Engineering Group, Inc. Sept 2008.

[2] United States Army, "Eyes of the Army - US Army Unmanned Aircraft Systems Roadmap 2010-2035", U.S. Army UAS Center of Excellence, 2010.

[3] NATO STANAG 4586 Edition 2.5, Feb 2007.

[4] M.L. Cummings, A. Kirschbaum, A. Sulmistras and J.T. Platts, "STANAG 4586 Human Supervisory Control Implications", *Unmanned Vehicle Systems Canada Conference*, Montebello QC, 2006.

[5] Ian Ashdown et al., "Common HMI for UxVs: Design Philosophy and Design Concept", Human Factors Integration, Defence Technology Centre Report, *BAE Systems*, June 2010.

[6] Anders Josefsson, J. Marklund, "IDC2 - a new C2 concept within the framework of a Network Based Defence Concept", 13th ICCRTS, 2008.

[7] P. Gustavsson, M. Hieb, P. Moore, P. Eriksson and L. Niklasson, "Operation Intent and Effects Model", Journal of Defense Modeling and Simulation, Sept. 2010.

[8] JC3IEDM, Version 3.0.2, May 2009.

[9] F. DiLego, J. Hitchings, C. Salisbury and H. Simmons, "Joint Airspace Management and Deconfliction (JASMAD)", *US Air Force Research Laboratory* AFRL-RI-RS-TR-2009-13, Jan 2009.

[10] "Too Much Information, Taming the UAV Data Explosion", Defense Industry Daily, May 2010.

[11] R. Parasuraman, T. Sheridan, "A Model for Types and Levels of Human Interaction with Automation",Vol. 30, No. 3, May 2000.

[12] Carl Nehme, Jacob Crandall, Mary Cummings, "An Operator Function Taxonomy for Unmanned Aerial Vehicle Missions", 12th ICCRTS, 2007.

[13] B. Eovito, "The Impact of Synchronous Text-Based Chat on Military Command and Control", 11th ICCRTS, 2006.

[14] M.L. Cummings, S. Bruni, S. Mercier and P.J. Mitchell, "Automation Architecture for Single Operator, Multiple UAV Command and Control", *The International C2 Journal*, Vol 1, No 2, 2007.

[15] G. Halak, "Joint Capabilities Group UAV Study Request - Architecture/Standards for Multi-domain unmanned platform Control System, i.e. UxS Contorl System", April 30th 2010.

[16] H.-M. Huang, E. Messina and J. Albus, "Toward a Generic Model for Autonomy Levels for Unmanned Systems (ALFUS*)", National Institute of Standards and Technology*, 2003

[17] Letter from Boeing to STANAG 4586 CST. 2010.

[18] M.L. Cummings, J.T. Platts and A. Sulmistras, "Human Performance Considerations in the Development of Interoperability Standards for UAV Interfaces", *Moving Autonomy Forward Conference*, Lincoln UK 2006.

[19] S. Banbury et al. "Intelligent Adaptive Systems – Literature Research of Design Guidance for Intelligent Adaptive Automation and Interfaces", DRDC–Toronto Contract Report SP W7711-067983/001/TOR, May 2009.

[20] G. Putrich, "US Air Force formalises UAV pilot training", *Pakistan Defence*, http://www.defence.pk/forums/general-defence/62692-us-air-force-formalises-uav-pilot-training.html, Feb 2009.

[21] D.H. Pham, "Efficient Representation and Effective Reasoning for Multi-Agent Systems", Ph.D Thesis, University of Queensland, April 2010.

[22] M. Ucuncu, and M. Demirezen, "Formatted Message Exchange in a Multinational Environment" *Information Fusion for Command Support, (pp. 2-1 – 2-10).* 2006.

[23] K. Müller, "NATO and XML", *XML Europe Conference*, June 2000.

[24] H. Demers, J.-R. Duquet, "SATAC Knowledge Representation and Automated Reasoning with JC3IEDM", Defence R&D Canada – Valcartier, Contract Report 2008-61, Sep 2008.

[25] C. Matheus, B. Ulicny, "On the Automatic Generation of an OWL Ontology based on the Joint C3 Information Exchange Data Model", 11th ICCRTS, June 2006.

[26] B. Smith, K. Miettinen, W. Mandrick, "The Ontology of Command and Control (C2)", 14th ICCRTS, June 2009.

[27] A.-C. Boury-Brisset, “Concepts and technologies for a knowledge environment supporting situation awareness”, 13th ICCRTS, June 2008.

[28] Foundation for Intelligent Physical Agents, http://www.fipa.org/specs/fipa00037/SC00037J.html.

[29] R. Franklin, “STANAG 4586, NATO Network Enabled Capability Service Oriented Architecture Working Group – Status Overview Presentation”, Oct 2010 STANAG 4586 CST Meeting, Oct 2010.

[30] F. Hassaine, K. Heffner, M. St-Onge, S. Chartrand, “Using BML to Command and Control UAV Systems in a Coalition”, NATO Modelling and Simulation Group 079, C-BML Workshop, Farnborough UK, Feb 2010.

[31] K. Heffner, M. Pullen, L. Khimeche, “MSG-048 Technical Activity Experimentation to Evaluate Applicability of a Coalition Battle Management Language in NATO”, NATO MSG-076 Symposium on “Blending Live-Virtual-Constructive (LVC) Simulation to Better Support Training & Experimentation”, Utrecht, The Netherlands, Sept. 2010.

[32] K. Heffner and F. Hassaine, “Using BML for Command & Control of Autonomous Unmanned Air Systems”, 10F-SIW-054, Fall 2010 Simulation Interoperability Workshop, Orlando Fl, Sept 2010.

[33] NATO Modelling and Simulation Group 048 (C-BML) Technical Activity Final Report, 2010.

[34] A. Tolk, J.A. Muguira, “The Levels of Conceptual Interoperability Model (LCIM)”, *Fall Simulation Interoperability Workshop*, Orlando, FL, Sep 2003.

[35] Ulrich Schade and Michael Hieb, “Development of Formal Grammars to Support Coalition Command and Control: A Battle Management Language for Orders, Requests, and Reports”, 11th International Command and Control Research and Technology Symposium, Cambridge, UK, 2006.

[36] F. Hayes-Roth, “Valued Information at the Right Time (VIRT): Why *less* Volume is *more* Value in Hastily Formed Networks”, Naval Postgraduate School Cebrowski Institute, 2006.

[37] S. Carey, M. Kleiner, M. Hieb and R. Brown, “Standardizing Battle Management Language - Facilitating Coalition Interoperability”, *European Simulation Interoperability Workshop*, London UK, June 2002.

Author Biographies

DR. KEVIN HEFFNER holds a BS in Engineering from the State University of NY at Buffalo and a Ph.D from University of Paris VI. He has worked in the field of modeling and simulation for over 20 years. His current activities include standards development for C2-simulation-autonomous systems interoperability and research in the area of future employment of Unmanned Vehicle Systems.

DR. FAWZI HASSAINE is a defence scientist at the Defence Research and Development Canada (DRDC). He holds a Ph.D in Computer Science from Paris VI University, and has accumulated more than 20 years experience in the domains of parallel and distributed computing, integration of distributed systems, and for the past 8 years, military applications involving Synthetic Environments, Computer Generated Forces, Simulation Interoperability, Tactics Development Systems, Command and Control, and UAVs.