2008

Coalition Battle Management Language (C-BML) Phase 1 Specification Development / Paper 08S-SIW-004

Blais, Curtis

Coalition Battle Management Language (C-BML) Phase 1 Specification Development

Curtis Blais
MOVES Institute
Naval Postgraduate School
Monterey, CA 93943
clblais@nps.edu

Saikou Diallo
Virginia Modeling, Analysis, and Simulation Center
Old Dominion University
Suffolk, VA
sdiallo@odu.edu

Gary Farmer
Applied Research Laboratory
University of Texas
Austin, TX
farmer@arlut.utexas.edu

Stan Levine
C4I Center
George Mason University
Fairfax, VA
stan.levine@us.army.mil

Chuck Turnitsa
Virginia Modeling, Analysis, and Simulation Center
Old Dominion University
Suffolk, VA
cturnits@odu.edu

Keywords:

Coalition, Battle Management Language, Interoperability, Simulation, Command and Control

ABSTRACT: *The Coalition Battle Management Language (C-BML) is intended to be an unambiguous language for expressing and exchanging plans, orders, and reports across command and control (C2) systems, modeling and simulation (M&S) systems, and robotics systems. In March 2006, the Simulation Interoperability Standards Organization (SISO) approved initiation of a Product Development Group (PDG) to generate a standard and guidance document for C-BML. The PDG laid out a 3-phase development effort to (1) specify a sufficient data model to unambiguously define a set of military orders using the Joint Command, Control, and Consultation Information Exchange Data Model (JC3IEDM) as a starting point; (2) develop a formal grammar (lexicon and production rules) to formalize the definition of orders, requests, and reports; and (3) develop a formal battle management ontology to enable conceptual interoperability across software systems. This paper reports on the technical approach and status of development of the C-BML Phase 1 Specification.*

1. Introduction

The Coalition Battle Management Language (C-BML) is an emerging standard for expressing and exchanging plans, orders, and reports across command and control (C2) systems, live, virtual and constructive modeling and simulation (M&S) systems, and robotic systems participating in Coalition operations. During the Spring 2004 Simulation Interoperability Workshop (SIW), a meeting of subject matter experts decided that it would be beneficial to the international M&S community to merge US Army Battle Management Language (BML) initiatives with other countries' BML interests to create a Coalition BML (C-BML) standard. As a result, a statement of work was drafted and submitted to the Simulation Interoperability Standards Organization (SISO) Standards Activity Committee (SAC). In September 2004, the SISO SAC approved the establishment of a Study Group (SG) on the C-BML to describe requirements and determine international interest in a standardization effort. The C-BML SG was formed under the following premise [1]:

> *In order to improve simulation interoperability and better support the military user with M&S-based capabilities an open standards-based framework is needed that establishes operational and technical coherence among C2 and M&S systems. The objective capability will enable automatic and rapid unambiguous initialization and control of one by the other.*

The C-BML SG formally began work at the Fall 2004 SIW under sponsorship of the SISO Command, Control, Communication, Computers, and Intelligence (C4I) Forum. In addition to its SISO membership, the SG collaborated with other organizations with potential interest in this work; in particular, the North Atlantic Treaty Organization (NATO) Modeling and Simulation Group (MSG) and the Command and Control Research and Technology Symposium (CCRTS). The SG completed work with submission of a final report [2] to the SISO Executive Committee (EXCOM), SAC, and Conference Committee (CC) at the Fall 2005 SIW. That report recommended initiation of a Product Development Group to proceed with development of a specification for C-BML for standardization through SISO, and provided a Product Nomination to that end. The SAC approved the Product Nomination, resulting in establishment of a Product Development Group and Drafting Group for development of the C-BML specification.

In accordance with SG recommendations, the C-BML specification is being produced in the following three phases providing incremental versions with increasing capability:

- Phase 1, Data Model: Phase 1 of the C-BML standardization effort (described in this paper) is defining the basic data model underlying the construction of C-BML expressions (plans, orders, and reports). The data model identifies a sufficient data set, using the Joint Command, Control, and Consultation Information Exchange Data Model (JC3IEDM) [3] as a starting point, for expressing portions of basic Orders information so that they can be unambiguously interpreted by C2, M&S and Robotic systems. The Phase 1 Specification also describes a standard information exchange content and structure specification in the form of an Extensible Markup Language (XML) schema, as well as an information exchange mechanism specification expressed as a Web Services Description Language (WSDL) document. The initial version of the C-BML XML schema will be evaluated by the parallel NATO MSG-048 effort. Discussion of the data model as a basis for C-BML can be found in [4].
- Phase 2, Formal Structure (Grammar): Phase 2 of the C-BML standardization effort will extend the Phase 1 products to more completely create unambiguous expression of Plans and Orders through a formalized grammar (syntax, semantics, and vocabulary). The objective is to formalize the definition of tasks such that they are rigorous, well documented, and parse-able. The grammar will be extended to accommodate reports after the tasking grammar is defined. Early work on the grammar can be found in [5, 6, 7].
- Phase 3, Formal Semantics (Ontology): Phase 3 will involve specification of a battle management ontology to enable conceptual interoperability across systems. Preliminary discussion of C-BML ontology issues can be found in [8].

Each phase of the C-BML specification development will describe:

- A data model (the C-BML SG recommended JC3IEDM as a starting point for all phases of the effort);
- An information exchange content and structure specification defining valid form and content of C-BML expressions;
- An information exchange mechanism specification enabling a common approach to implementation of applications that can process C-BML information;
- Guidelines for adoption and application of the standard which explain C-BML use and provide practical examples.

This paper describes work to date on the draft C-BML Phase 1 Specification to inform the community on the development approach, status of the effort, issues that have arisen during the development and either resolved or yet to be resolved, and remaining effort needed to prepare the Phase 1 Specification for balloting.

2. Coalition Battle Management Language Concept

Fundamentally, when two systems need to exchange information, one system sends the information to the other through some communications medium, as depicted below.

Figure 1. Generic System-to-System Interaction

Several configurations are possible. System A could be a C2 system passing an order to a simulation system (System B) to be executed in the simulation environment. Or, System A could be a constructive simulation system passing synthetic target data to a virtual simulation (System B). Or, System A could be a robotics system providing situation report data to a C2 system (System B). Many other such combinations apply, but they all share the same fundamental notion. Currently, there are many formats for the information being transferred. Some of the formats are standardized and used by many systems; some are specialized and used by a small number of systems. In the worst case, two systems interact using

unique point-to-point (for that pair of systems) information formats.
In the case of the transfer of plans, orders, and reports, the C-BML concept, very simply, is all about standardization of the structure, content, and mechanism for this information exchange, as shown in Figure 2.

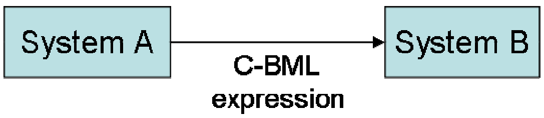

Figure 2. System-to-System Interaction using C-BML

In military operations, the nature (format and content) of the information to be exchanged is determined by the doctrine governing the exchange. Certain forces conducting certain operations are required by doctrine to provide certain information. The "C-BML expression" in the diagram above essentially encapsulates the doctrinal exchange. Said another way, for a system to employ C-BML, its doctrinal expressions, in whatever form (e.g., military message formats) must be transformed into C-BML expressions, either directly or through some adapter. If all systems adopt the C-BML standard, then only C-BML expressions will be transmitted between and among systems when transferring plans, orders, and reports.

3. C-BML Specification

In terms of the above C-BML concept, the purpose of the C-BML specification is to define standards for the generation and transfer of C-BML expressions. *Generation* of C-BML expressions depends upon two parts of the C-BML Specification: (1) the C-BML information exchange structure and content; and (2) the C-BML data model (see Figure 3). Together these describe what needs to be expressed and how it needs to be expressed. *Transfer* of the C-BML expressions across systems employs the third part of this specification; namely, the information exchange mechanism. This concept for employment of the elements of the C-BML Specification is summarized in Figure 3.

The C-BML Specification is intended for use by software developers (specification, design, implementation, and test) and standards developers in the C2 and M&S domains. The document will describe the structure and purpose of the language in sufficient detail to support early adoption while follow-on phases in the specification development continue. A separate Guidelines document will provide explanation and examples of application of the standard as an aid to adoption of the standard across M&S and C4I communities.

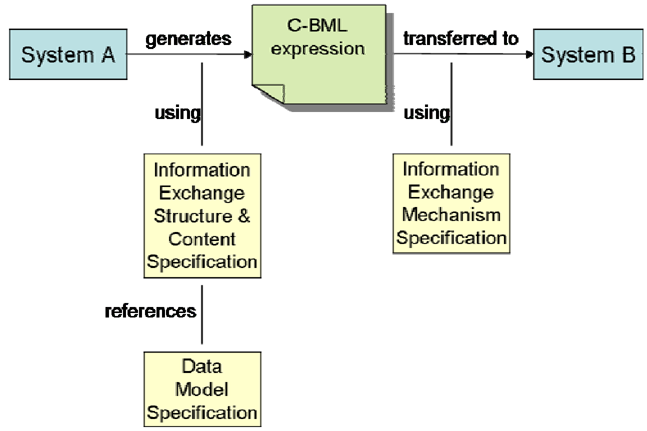

Figure 3. Concept of Employment of the C-BML Specification

3.1 C-BML Phase 1 Specification Scope

The C-BML Phase 1 Specification will provide a foundation and starting point for further development of the standard in the later phases. The current draft Phase 1 Specification defines the following portions of the C-BML standard: (1) a standard data model and procedures for extending the data model (the data model specification in Figure 3); (2) a description of basic information components of the language using XML as the specification language (initial elements of the information exchange structure and content specification shown in Figure 3); and (3) a standard approach for exchanging elements of the language specified using WSDL (initial information exchange mechanism specification shown in Figure 3).

Follow-on phases in the specification development process will augment and extend this initial specification with (1) description of a formal grammar for plans, orders, and reports (Phase 2) that will identify extensions to the data model, expansion and detailing of the information components, and expanded support for plans, orders, and reports; and (2) specification of formal semantics of the language (Phase 3). The Phase 1 Guidelines document has not yet been developed to explain application of the Phase 1 Specification pending resolution of key issues remaining in development of the Phase 1 specification.

3.2 C-BML Phase 1 Specification Objectives

The primary objective of the C-BML Phase 1 Specification is to provide sufficient information to enable early adopters of the C-BML standard to construct and exchange standard information elements in plans, orders, and reports.

3.3 Related Standards

The Military Scenario Definition Language (MSDL) is a common representation of scenario information that can be exchanged across multiple C2 and M&S systems. Intended for use in initializing various systems, MSDL describes the physical setting of a scenario (e.g., terrain, weather), forces and force structures, control measures, and other information. MSDL has also progressed to Product Development Group status in SISO and is in initial balloting at time of this writing (SISO 2007).

MSDL and C-BML are mutually dependent standards development efforts. This dependence is rooted in the identification in MSDL of forces and scenario settings that are used in BML expressions of plans, orders, and reports. Furthermore, as a common language for initializing C2 and M&S systems, MSDL needs to contain initial sets of plans and orders (e.g., air tasking orders, initial ground movement orders, and ship-to-shore landing plans) that are expressed in C-BML. Accordingly, all C-BML products and artifacts developed under the PDG are shared openly with the MSDL PDG. To facilitate cooperation and collaboration between the two PDGs, one member of each group serves on the leadership of the other. There are also a number of other individuals who are members of both PDGs and respective Drafting Groups (DGs), helping to ensure effective coordination and collaboration across the groups in development of their respective standards products.

In the event that a conflict is identified between the C-BML PDG and the MSDL PDG, the individual discovering the conflict will notify both PDGs. If the PDGs cannot resolve the conflict then the conflict will be elevated to the SAC. Notification will be made to the Technical Area Director (TAD) responsible for oversight of the C-BML products in accordance with the defined appeal process found in the SISO-ADM-002-2006 SISO Policies & Procedures document.

3.4 Phase 1 Specification Development Starting Point

The C-BML Phase 1 Specification development effort began with the following understanding of the prior work and PDG guidance:

(1) The Phase 1 C-BML data model is JC3IEDM, as directed by the C-BML PDG.

(2) Grammar will be addressed in Phase 2 of the C-BML specification development. To avoid issues of grammatical structure in Phase 1, the Drafting Group focused on defining the 5Ws (Who, What, When, Where, Why) as the basic information elements and building blocks of the language without regard to production rules for constructing valid "sentences" in the language using the 5Ws.[1] The 5Ws have long been recognized as important, although not exhaustive, information elements of real-world Operations Orders, Air Tasking Orders, reports, and other doctrinal expressions. The Phase 1 Specification defines these "terms" but does not specify any restricted set of "sentences" using the terms as will occur in the Phase 2 effort. In defining the formal grammar for C-BML in the Phase 2 Specification effort, necessary extensions and detailing of the 5Ws and addition of other needed information elements will be specified as well.

(3) Ontology will be addressed in Phase 3 of the C-BML specification development. To avoid issues of ontology, the DG did not address any formally defined semantics of the 5Ws or expressions using the 5Ws beyond association of the 5Ws with the underlying data model (JC3IEDM) and any explicit or implicit semantics resulting from association with that model.

(4) Numerous resources were available for review and consideration in the drafting effort, including papers, reports, studies, demonstrations, XML schemas, WSDL specifications, and code implementations. All of the prior implementations addressed portions of specific doctrines (e.g., US ground operations, US air operations, NATO ground operations, etc.). The DG tried to keep the Phase 1 Specification as generic as possible to avoid leaning toward any particular national or service doctrine, other than what may be implied by the intrinsic vocabulary and definitions in the identified data model (JC3IEDM). In the Phase 2 effort to define a formal grammar, it is expected that the grammar will enable creation of expressions that fully meet the information needs of some set of real-world doctrinal expressions.

These assumptions and starting positions have shaped the drafting of the Phase 1 Specification. Of course, the DG is open to re-direction if any of its understandings are incorrect or if the Phase 1 effort has been scoped improperly.

4. C-BML Phase 1 Data Model Specification

This section describes the data model selected for the C-BML Phase 1 Specification; namely, the JC3IEDM Baseline Edition 3.1a [3] managed by the Multilateral

[1] A grammar can be denoted by the set (V_N, V_T, P, S) where V_N, V_T, P, and S are, respectively, the non-terminal symbols, terminal symbols, productions, and start symbol [9]. In Phase 1, the 5Ws can be considered an initial set of non-terminals for future specification of the C-BML grammar.

Interoperability Programme (MIP). This data model provides a basic lexicon for C-BML.

4.1 Base Model: JC3IEDM

JC3IEDM is the central reference model for initial specification of C-BML. JC3IEDM is sufficiently robust to handle most, if not all, of the required data in command and control data exchanges across the systems that C-BML is intended to serve (C2, M&S, and robotic systems). Within the international M&S and C2 communities, there is widespread acceptance of the JC3IEDM as the standard construct for interchanging command and control information. JC3IEDM serves as a neutral, independent data model that belongs to no single system or community (branch of service or nation of origin), but can serve to describe the basic information exchanged between most of the systems that C-BML is interested in (again, C2, M&S, and robotic systems).

4.2 Model Overview

The JC3IEDM data model comprises two categories: the Generic Hub (GH) and the Sub-Functional Areas (SFA). The data model encompasses information from multiple functional areas in the domain of military operations. All common data, or better said, all data that need to be exchanged by at least two functional areas, become part of the Generic Hub. The remaining data is modeled as extensions of the Generic Hub data into the Sub-Functional Areas.

Initial evolution of the JC3IEDM under MIP included specific inputs from the following functional areas: conventional fire support, barrier engineering operations, communications and electronics, and personnel administration. Operational requirements have been drawn from these as well as other areas, as documented by the MIP (http://www.mip-site.org). JC3IEDM describes all objects of interest on the battlefield, e.g., organizations, persons, equipment, facilities, geographic features, weather phenomena, and military control measures such as boundaries, using a common and extensible data modeling approach.

Fundamental information in JC3IEDM includes such concepts as OBJECT-TYPE, OBJECT-ITEM, ACTION, and CAPABILITY. For example, the battlefield consists of a large number of objects, each with its own set of characteristics. Objects may be described as a class or type rather than as individually identified items. Actual instances are catered for by use of OBJECT-ITEM. Types are recorded as OBJECT-TYPE. While general attributes are collected on the type side, such as general capabilities and abilities, only the instantiation specific values are on the item side. Examples are the caliber of the weapon being specified on the type side, but the actual ammunition state and location are on the item side. The top-level relationship between OBJECT-TYPE and OBJECT-ITEM is shown in Figure 4. The figure also shows the major subcategories of OBJECT-TYPE and OBJECT-ITEM represented in the data model. The subcategories are given by a category code. In addition to the values of FACILITY, FEATURE, MATERIEL, ORGANISATION, and PERSON, there is also an entry of "Not known" for an OBJECT-ITEM which is tracked but has not yet been classified. If additional categories are needed to meet a particular application requirement, the model can be extended through a process managed by the MIP. Extensions may involve adding new attribute values (e.g., new category and sub-category codes for OBJECT-TYPE and OBJECT-ITEM), adding new attributes, adding new tables, or adding new associations.

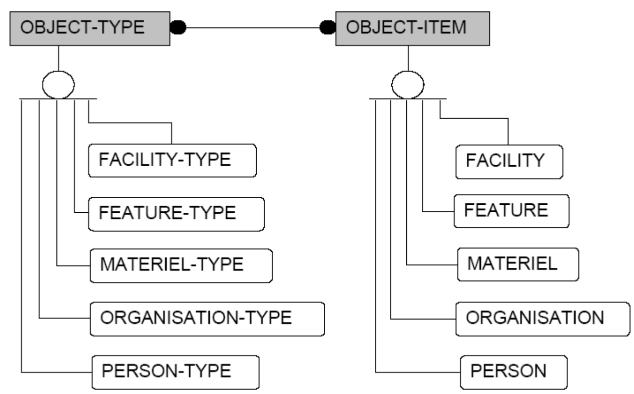

Figure 4. JC3IEDM Object-Type and Object-Item Entities

Clearly, it is not possible to provide a complete description of the JC3IEDM in this paper. Refer instead to the full set of JC3IEDM documentation provided at the MIP site[2]. The following paragraphs provide some additional information as a brief introduction to JC3IEDM structure and terminology and to address how the data model can be extended as needed to support C-BML.

4.3 Data Model Integrity

As the data model is used by a growing community of other systems for information exchange, there is value in keeping the GH portion of the data model intact. While a good goal, it is naïve to think that there will not need to be extensions to what is in the model in order to support the exchange needs of all future C-BML users. Insofar as the hub portion remains inviolate and all such extensions take place in sub-functional areas, then interchange of

[2] See http://www.mip-site.org/publicsite/04-Baseline_3.0

information with other users of the JC3IEDM will be possible, even if they do not access the common reference model via the C-BML standard methods.

The model as presented by the MIP is itself a model that consists of a number of different elements presenting a meaningful whole for enabling data exchange. These elements include the entities (commonly implemented as tables in a database management system (DBMS) implementation of the model), attributes (fields in the tables), relations (relational links between the different tables), and the rules of intended use. These are each described below.

4.3 1 Entities

An entity represents a discrete object in the structure of the data model. Entities can be thought of as nouns. Within the JC3IEDM, entities can, among other things, be representative of some physical thing (OBJECT-ITEM), a class of items (OBJECT-TYPE), some process (ACTION-TASK), or the results of some process (ACTION-EFFECT). The JC3IEDM also uses some relationship entities to grant attribution to a relationship between two (or more) other entities (known within the JC3IEDM as an association); this is described further below. When the model is used as a guide for a database implementation (i.e., within a DBMS) the entities are represented by the tables of the database.

4.3.2 Attributes

Entities and relationships can both have attributes. Attributes are the data elements associated with either the entity or the relationship. Some of these are required for the identity of the entity or relationship. Some examples include physical characteristics for OBJECT-TYPE entities or index values for relationships. Within a database implementation of the model, attributes are the fields within a table.

4.3.3 Relations

Some entities have semantic links to other entities, in order to represent a more complex idea than can be represented in a single entity. Within the JC3IEDM, some of these relations are required and some are optional. In all cases, when a relation exists, the entities so joined become a relationship. All relationships in the JC3IEDM have a phrase that defines the relationship; some examples of these are *is-specified-as* and *is-the-object-of*. These relationships are referred to as associations within the data model.

If the relationship requires attribution, then there may be an intermediary entity that exists only to hold that attribute. For convenience, these may be referred to as relationship-entities. Some examples of a relationship-entity are ACTION-TASK-RULE-OF-ENGAGEMENT (combining ACTION-TASK and RULE-OF-ENGAGEMENT) and CANDIDATE-TARGET-DETAIL-ASSOCIATION (which combines two CANDIDATE-TARGET-DETAIL entities, one as the subject of the other).

When the data model is implemented in a relational database, the foreign key mechanism is used to designate associations (relations). If the relationship requires attribution, then a relation between the associated entities and the relationship-entity (which contains the attribute) is made, and all are associated via the use of indexes as attributes.

4.4 Intended Use

The JC3IEDM documentation available from the MIP has a great deal to say about how the above elements are used and when they are and are not required to be present. This is documented in not only the JC3IEDM Overview document and Main document, but in great detail within the published annexes. Examples within the documentation cover many of the common uses for the model (the most common forms of C3 data exchange – including tasks, reports, and others), and how the correct elements must be employed. Sequence and structure, where important, are noted, as well as the use of mandatory fields.

4.5 Data Model Extension

The MIP has set forth procedures to process change proposals in the JC3IEDM Guide to Change Proposals [10]. C-BML implementers shall use this document to process change proposals in the future C-BML Product Support Group (PSG). This has two benefits. First, the guide is well thought-out and includes an example. Second, if the C-BML community wishes to propose those changes to the MIP, the change proposals will already be written in the MIP’s desired format. Changes that are determined to be US specific will go through the US-JC3IEDM CM procedures.

The MIP has set forth recommendations for naming entities, attributes, and relationships in the JC3IEDM Annex H, Naming Conventions and Class Words [11]. C-BML users shall follow the guidelines of this document to select names for new data objects.

The IDEF1X data model that formally describes the JC3IEDM can be extended in three ways: through the addition of entities, attributes, and domain values.

Additionally, added attributes may require new domains and validations. These extensions are discussed below.

4.5.1 Minor Extensions – Domain Values

The simplest extension is to add a domain value to an existing enumeration. An enumeration is a validation rule that consists of a set of domain values. For example, the JC3IEDM validation rule DS144_obj_item_cat_code limits the values of the object-item-category-code attribute of the OBJECT-ITEM entity to the values "FA", "FE", "MA", "NKN", "OR", and "PE". Each domain value must have a textual value, a display value, and a definition. In the example of the DS144_obj_item_cat_code validation rule, the "FA" textual value has display value "FACILITY", and definition "An OBJECT-ITEM that is built, installed or established to serve some particular purpose and is identified by the service it provides rather than by its content."

4.5.2 Moderate Extensions – Properties

A more complex extension is to add a new attribute to an existing entity. Each attribute is either inherited from another attribute through a foreign key or is a "base" attribute, not having a reference. Inherited attributes will inherit names and domains from the "parent" attribute. These can be changed, if needed. For example, in the JC3IEDM, the organization-id attribute of the ORGANISATION entity is inherited from the object-item-id attribute of the OBJECT-ITEM entity, then renamed. Each attribute has an attribute name, a column name, and a definition. In the JC3IEDM, each attribute name is lowercase, composed of complete words (often beginning with the entity name), and separated by hyphens. Examples include object-item-id and object-item-category-code. In the corresponding column names, words are abbreviated, with common abbreviations shown in the JC3IEDM Data Model Metamodel [12], and separated by underscores. Column names are lowercase and generally do not include the table name, except in the case of ID (identifier) and IX (index) columns. Continuing the previous example, example column names include obj_item_id and cat_code. Each attribute must have either a datatype or a domain. In the latter case, the domain will have a datatype, which the attribute will inherit.

Each domain must have a logical name, physical name, definition, null indicator code, and parent domain. JC3IEDM domains are either specific to one attribute or generic to several attributes. If specific to one attribute, then the logical domain name is the same as the attribute name and the physical domain name is the same as the column name. For example, the object-item-category-code domain corresponds directly to the object-item-category-code attribute of the OBJECT-ITEM entity. If the domain is generic to several attributes, then the logical and physical domain names are more general than the related attribute names. For example, the domain distance-precision-code is associated with several attributes, including the relative-point-y-precision-code attribute of the RELATIVE-POINT entity.

A number of domains already exist in the JC3IEDM. However, if none of the existing domains correspond to the newly added attribute, then a new domain must be added. The JC3IEDM domains exist in a hierarchy, with various domains descended from the Number and String primitive datatypes. The Number domain has subtypes of angle; cnt (count); coord (coordinate); dim (dimension); dttm (date-time); dur (duration); ord (order, as in a sequence); qty (quantity); rat (ratio); and rate. Each of these is further divided into a mandatory domain and an optional domain. The String domain has subtypes of code, id, ix, and txt (text). Some of these have further subtypes to indicate particular attributes; for example, object-item-id is a subtype of the id domain and object-item-category-code is a subtype of the code domain. A new domain should descend from one of these "parent" domains. If the domain is specific to the attribute, the logical domain name is the same as the attribute name and the physical domain name is the same as the column name. If the domain is generic, the attribute and column names will be more specific.

A domain may have an associated validation rule to restrict the values of the domain. Each validation rule is either a range restriction or an enumeration. A range restriction must either be a maximum, a minimum, or both. Maximum and minimum values may be inclusive or exclusive. As explained in the previous section, enumerations are sets of domain values. Validation rule names begin with "DS", an index number, and the physical domain name, for example DS144_obj_item_cat_code. If a validation rule must be added for C-BML, it will begin with "DS_CBML", followed by a unique, three digit sequence number, then the physical domain name.

4.5.3 Major Extensions – Structure

The most difficult extension to the data model is the addition of a new entity. An entity has an entity name, a table name, a definition, and a primary key. Both entity and table names are uppercase. The entity name consists of complete words separated by hyphens, for example OBJECT-ITEM. The table name consists of abbreviated words, with common abbreviations shown in the JC3IEDM Data Model Metamodel [12], separated by underscores, for example OBJ_ITEM. Each entity has one or more attributes. In order to add attributes, see the previous paragraph. Each entity must have a primary key and a set of non-null attributes that are unique.

4.6 Community Processes

Following approval of the C-BML Phase 1 Specification, SISO will initiate a Product Support Group (PSG) to maintain the specification and to provide assistance to the user community. As early adopters begin employing the C-BML specification and as specification development proceeds into Phases 2 and 3, changes to the JC3IEDM will be needed. Users will request extensions to the data model by preparing a Change Proposal in accordance with MIP instructions and will submit that proposal to the C-BML PSG. The C-BML PSG Configuration Management process will determine if a recommended extension is useful, from a C-BML information exchange perspective, to two or more member nations (the extension applies to the doctrine of two or more nations) and either reject the proposal or approve the proposal for inclusion in the evolving C-BML data model (JC3IEDM plus approved C-BML extensions). When extensions are made in the C-BML Data Model (documented in IDEF1X for conformance with JC3IEDM documentation), corresponding changes will be made in the C-BML Information Exchange Content and Structure (XML schemas) and the C-BML Information Exchange Mechanism (WSDL). Changes that are determined to be valuable for recommendation to the MIP for incorporation in the official JC3IEDM will be forwarded by the PSG to the MIP using established forms and procedures for that purpose.

If a recommended extension would only be useful to one particular nation (based only on the doctrine of this one particular nation), that extension can be added to the nation-specific JC3IEDM or C-BML data model, but it would not be included in the official SISO-managed C-BML Data Model (core). Such a nation-specific extension could be used for information exchanged between systems of the same nation, but such extensions would not be approved for C-BML expressions between systems of two different nations.

5. C-BML Phase 1 Information Exchange Content and Structure Specification

To begin specification of C-BML, the Phase 1 Specification describes the basic 5Ws: Who, What, When, Where, and Why. In the abstract, this information is fundamental to the expression of primary elements of plans, orders, and reports for any doctrine of any service, nation, or organization. The following constitutes definition of the 5Ws for purposes of the Phase 1 Specification:

- Who: *C-BML information component identifying the battlespace object directed to perform an action (plan or order), that has been observed or has performed an action (report), or on which an action is to be performed (e.g., target).*
- What: *C-BML information component identifying an action to be performed (plan or order) or that has been performed (report).*
- When: *C-BML information component describing the timeframe in which an action is to occur (plan or order) or when an action or event has occurred (report).*
- Where: *C-BML information component providing the location of an object in the battlespace (*C-BML Who*), the location where an action is to occur (plan or order), or the location where an action or event has occurred (report). The location may be a complex object, such as an area or a sequence of locations.*
- Why: *C-BML information component describing the rationale or purpose of an action to be performed (plan or order), or the desired end state of a planned action.*

As fundamental information components of real-world plans, orders, and reports, the 5Ws constitute part of the doctrinal view of C-BML. The abstract nature of the terms facilitates employment of the components by any service, nation, or organization. A formalism for describing the C-BML information content and structure is the XML Schema language. This language provides a precise description of the information structure and content that can be used to validate XML documents containing C-BML expressions encoded in XML. Also, as discussed earlier in the document, use of XML further facilitates widespread adoption and deployment of the C-BML standard. The C-BML XML expression of the 5Ws provides the common format for expressing portions of plans, orders, and reports that can be exchanged across systems through a variety of mechanisms (such as the standard information exchange mechanism for C-BML described in section 6). Application of the approach for any specific service, nation, or organization requires transformation of current expressions (e.g., textual or binary message formats), some of which may already use defined XML tagsets, into the C-BML XML structures. Legacy systems will require adapters to produce and consume C-BML expressions. Over time, however, as C-BML becomes widely adopted, systems will emerge that natively “speak” C-BML, directly producing and processing C-BML expressions in place of older formats. Either way, systems will obtain the benefits of a shared, common structure and content for the expression of selected information elements in plans, orders, and reports.

The following subsections describe the C-BML XML schema description of the 5Ws and the relationship between the information elements described in the XML schema and the underlying JC3IEDM data model.

5.1 C-BML XML Schema Description of the 5Ws

The XML Schema definition of the 5Ws is provided in Annex B of the draft specification and discussed briefly below. Three separate schemas are defined: one describing the data structure for Who information (CBML_WHO_Logical.xsd), one describing What and When information (CBML_WHAT_WHEN_Logical.xsd), and one describing Where information (CBML_WHERE_Logical.xsd). These schemas reference (using the XML Schema *include* statement) the JC3IEDM logical XML schema JC3IEDMEntities.xsd, which in turn references data constructs in the JC3IEDMSimpleTypes.xsd and JC3IEDMCodes.xsd schemas. Initially, the schemas employ the same XML namespace; namely http://www.mip-site.org.[3]

The C-BML 5W schemas define information elements that can be employed to construct a variety of expressions. This is important because the C-BML Phase 1 Specification is not based on any specific doctrine and thus does not mandate any specific message structure and content. Rather, the Phase 1 Specification defines these basic information elements that are found in many different doctrinal expressions. Early adopters can employ these information elements in any structures they wish by invoking the applicable XML schema and declaring the C-BML namespace. The context and requirements of the information exchange will dictate what elements a system requires in order to correctly function. However, once an element in the top level is selected, users have to conform to the requirements and constraints defined by the schema. As an example, consider the C-BML "Who". The schema excerpt shown in Figure 5 says that a "Who" can be any of the optional elements defined at the top level (Unit, Road, Bridge, etc.). There is no mandate that a "Who" be any of those elements; however, if for example systems want to exchange information about a unit, they must follow the structure defined by the schema (i.e., a unit must at least have a name, a type and a formal abbreviated name as shown in the schema excerpt in Figure 6).

[3] Alternatively, the C-BML XML vocabulary can be defined in a SISO namespace such as "urn:sisostds:bml:draft:cbml:1" and can reference the JC3IEDM XML schemas using the XML Schema *import* statement.

The C-BML schema uses the logical constructs of the JC3IEDM. It shows the logical dependencies of the entities and leaves the physical implementation to the user. This is a deliberate choice made to allow for a flexible implementation of the standard while maintaining and enforcing a common high level (logical) view between all C-BML compliant systems. In fact, in early stages of adoption, C-BML users will likely have a legacy system that they will turn into a C-BML compliant system. Such users need not throw away existing solutions which are at the physical level (already implemented) but can generate a logical view of their system that can fulfill the information exchange requirements mandated by the C-BML logical view. As an example, consider two systems A and B that need to exchange initial information about a unit using the C-BML "Who" schema. The schema excerpt shown in Figure 6 specifies that a unit has to have a name, a type, and a formal abbreviated name. While at the the logical level this information is sufficient, it is ambiguous when one has to turn it into a physical item. As a result, additional requirements found in the JC3IEDM business rules must be taken into account. In JC3IEDM, business rules "specify constraints that either cannot be expressed in formal IDEF1X notation or those that are not explicitly structured as a design choice" [13]. In this case, the user has to at least unambiguously identify the unit once using a globally unique identifier. In addition, it is mandated that a unit type be defined before that type is referenced; thus the physical implementation has to implement a way to insure that this mandate is respected. Consequently, the following algorithm can be employed to transition from the logical view to the physical view:

1. Check that the unit type being referenced exists.
2. If yes, continue to 3; if not, reject the information as incomplete.
3. Check the unique identifier. If the object already exists, accept the information and move to 5.
4. If the object does not exist, create the object and assign a unique identifier.
5. Exit

The implementation of this algorithm is left to the user. The most straightforward and common way is to use a relational approach; however, the C-BML standard does not advocate one approach other another as long as the information exchange requirements are fulfilled.

The C-BML schema adheres to all of the business rules specified by the MIP. Users are encouraged to familiarize themselves with those rules in order to accurately use C-BML. The C-BML schema for each of the 5Ws is independent (except for the What and the When). That is, exchanging one of the information elements (a "W") does not depend on any other W (except for the closely related What-When information).

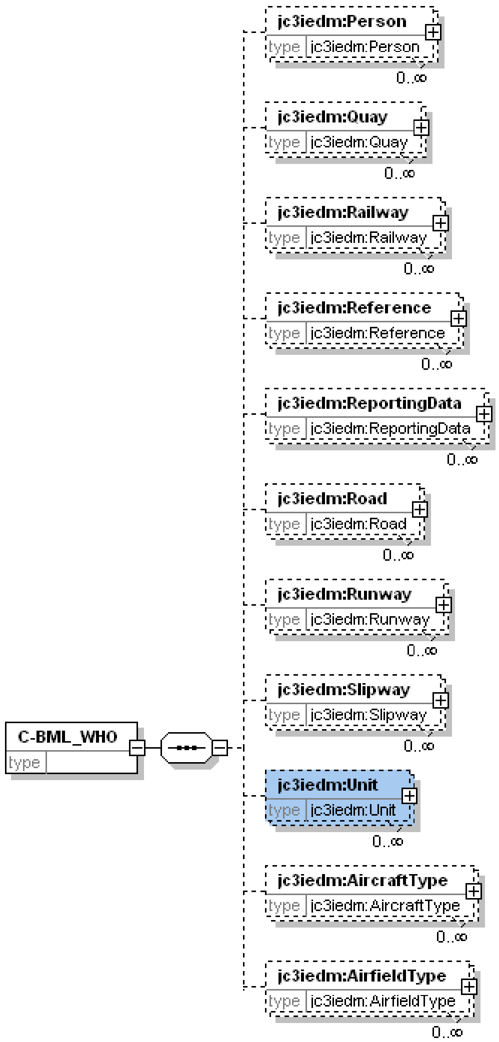

Figure 5: Sample C-BML WHO

In addition, each of the C-BML schemas is platform-independent, meaning each schema only represents the logical information exchange requirements. All physical implementations shall remain consistent with the logical schema it implements and shall not add additional constraints that are neither defined nor enforced by the schemas. All mandatory elements in the schema shall remain mandatory, and no optional element shall become mandatory when transitioning from the logical to the physical view and vice versa.

The Why component of the 5Ws is a particular case that can be broken down into two facets:

- *Global Why*: The Global Why represents the desired end state or purpose of a mission. The NATO standardization agreement (STANAG) for instance states that a mission is "a clear, concise statement of the task(s) to be accomplished and its purpose". The Global Why is usually accomplished through a series of tasks that are described in additional details such as in the concept of operation in the STANAG.
- *Local Why*: The Local Why is the reason for accomplishing an atomic task, where an atomic task is defined as a task that cannot be further broken into sub-tasks. In this sense the Local Why is the intended end state of the atomic action.

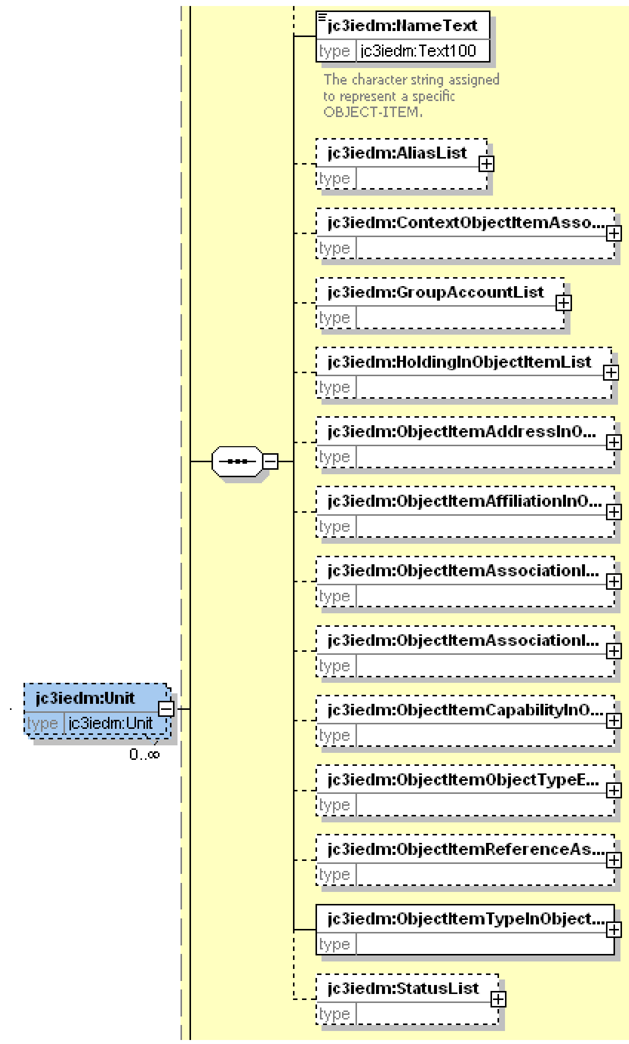

Figure 6: Sample JC3IEDM Unit

C-BML has to represent interactions at three levels: human-to-human, human-to-machine, and machine-to-machine. The common expression of the commander's intent in free text covers only meets the first of these interactions; namely, human-to-human. In order to express the Global Why, it is necessary to decompose the order into a series of tasks and sub-tasks that are then functionally related. When a task can no longer be decomposed into sub–tasks it is considered to be an elemental task with its Local Why. This artifact has two main advantages:

- The commander's intent can still be specified in free text using the C-BML What_When Schema. The commander's intent can be interpreted and carried out differently depending on doctrine and available forces. The C-BML Phase 1 Specification does not endorse, enforce or recommend any doctrine or implementation of the commander's intent. Thus, in the event that a system is not able to implement a series of tasks that are supposed to capture the commander's intent in one doctrine, it can always refer back to the free text expression and map it to its doctrine. As far as C-BML is concerned, the logical view stays unchanged, meaning the information exchange requirements between the systems stay the same even though they are expressed differently at the individual system level.
- While the commander's intent can always be expressed as free text within the C-BML What_When schema, the Global Why and the Local Why are captured in the C-BML Why schema. As a result, C-BML is able to support all interactions, human-to-human, human-to-machine, and machine-to-machine. The C-BML Why schema allows users to exchange the functional relationships between different tasks which facilitates Course of Action Analysis (COA) and After Action Review (AAR). It also unambiguously expresses the concept of operations or the desired end state of a mission in a machine understandable way.

As an Illustration, consider the vignette shown in Figure 7. The commander's intent is to "rid the area of insurgents as a part of a coordinated operation in Bahaullah province." The mission is to "occupy the city of Fallujah, capture Al-Qaeda facilities and neutralize their operations." In this case the end-state is to neutralize the operations of Al-Qaeda. In order to reach this end-state, we must (1) occupy to city of Fallujah; (2) capture Al-Qaeda facilities; and (3) neutralize their operations. The concept of operations further describes how the mission is to unfold and assigns specific tasks to units. The mission can now be decomposed into a series of tasks and sub-tasks with their respective Local Why; i.e., the desired end state of a particular task.

Figure 8 shows the functional breakdown of the mission described in Figure 7. It is important to note that this breakdown is valid with respect to available units and the chosen doctrine (in this case, United States Army). However, as stated before, the C-BML Phase 1 Specification does not mandate or advocate any doctrine over another; therefore the vignette could be implemented quite differently depending on the nation that undertakes it. The goal of the Phase 1 specification of C-BML is to allow fundamental information elements to be expressed and exchanged unambiguously regardless of the underlying doctrine.

5.2 Mapping C-BML Information Components to the Underlying Data Model

As indicated in the use of the "jc3iedm" namespace in Figure 5 and Figure 6, and seen in the full XML schema provided in Annex B of the draft specification, the specification of the C-BML 5Ws directly ties to the underlying data model. The implementation of information elements from the logical view to some physical implementation of the data model is left to the user; however, the physical implementations made available by the MIP are certainly recommended. The logical view must remain consistent with no information loss in the transition from the logical to the physical representation. Otherwise stated, to comply with the C-BML Phase 1 Specification, only valid information elements can be exchanged, where validity is based on fulfillment of all of the mandatory requirements as specified in the C-BML logical schema.

5.3 Constructing C-BML Information Components from the Underlying Data Model

While the transmission of C-BML expressions by a system requires valid constructs to be sent, the user is free to extract information elements as best suits his/her respective systems when gathering information from a C-BML compliant system. Users are encouraged to build contextually relevant information elements for their particular purposes. As an example, systems that are only interested in receiving the location and status of a given unit are encouraged to extract only those information elements from more complete expressions. However, the providing system must still produce valid constructs about the unit, the location, and the status as described earlier. The business objects shall be used to interface C-BML to a JC3IEDM system; it is not permissible for external processes to access individual JC3IEDM entities/tables for the purpose of C-BML interactions without using business objects defined in the Phase 1 Specification and be in conformance with the specification.

6. C-BML Phase 1 Information Exchange Mechanism Specification

Web Services is a methodology for invoking remote software execution that is becoming widespread across industry and military applications. Web services enable disparate system architectures and operating environments to initiate processing and exchange information in a standard way.

3. Execution:

Commander's intent: We will rid the area of insurgents as a part of a coordinated operation in Bahaullah province. Follow the rules of engagement and keep civilian casualties and collateral damage to a minimum.

a. Concept of the Operation: TF Tun Tavern will cordon Fallujah prior to first light, move to specified objective areas and seize Al-Qaida intelligence information (See EEI) at first light. Operation will last for 3 hours at which time control will be handed over to Iraqi Police.

i. 1st Marine Battalion, move from current position to a position from grid1234 to grid 4567. Alfa Company occupy the northern sector from grid 1234 to 1335; Bravo Company from grid 1335 to 3456; C Company from grid 3456 to 4567. All companies occupy position areas no later than 0445 hours.

b. A Company 1/503 AASALT, move from LZ Arnhem to insertion points on the eastern sector at grid nnnn; grid nnnn; and nnnn. Move to and occupy objective areas Hammer, Mallet, and Sledge no later than 0630. Do not cross coordination line Gold.

c. Det 4 SAS (UK) Occupy houses at nnnn, NLT 0645. Capture Abou Nasser; neutralize other forces. Egress to grid nnnn for extraction NLT 0800 hours.A Btry 6-15 FA priority of fires to 1st Marine Bn.

d. Occupy position areas 7 and 9 NLT 0400.

Figure 7: Example from Paragraph 3 of STANAG Operations Order

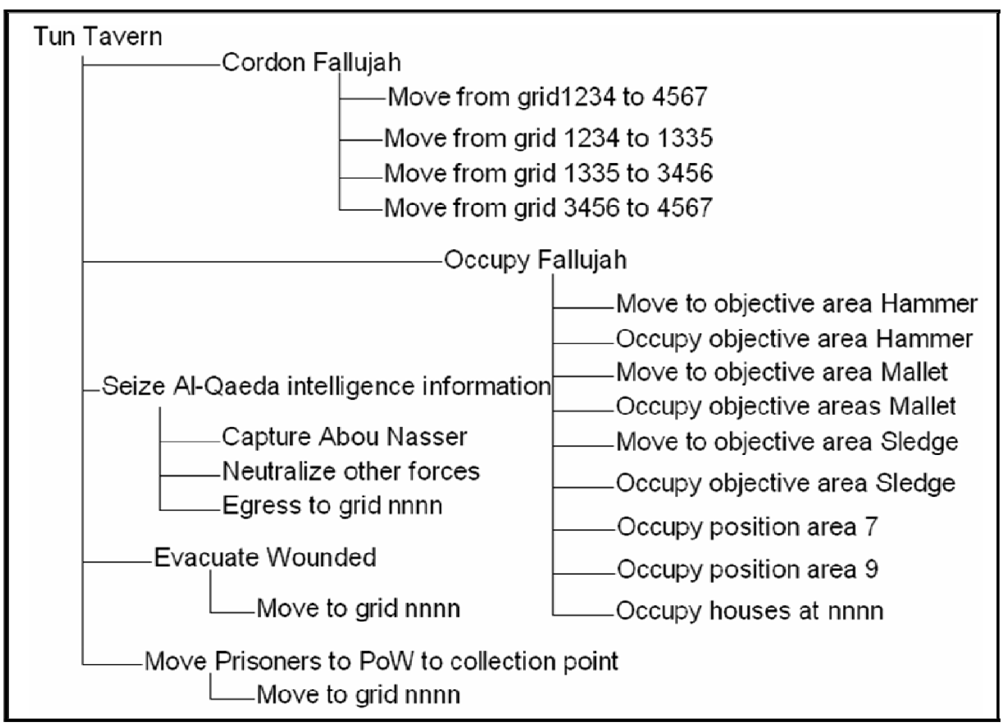

Figure 8: Functional Breakdown of the Operations Order Example

Service descriptions provide information about the interface offered by the service provider. A standard for describing services is the Web Service Description Language (WSDL). WSDL separates a service description into two parts [14]: (1) an abstract interface identifying the operations supported by the service, the operation parameters, and abstract data types; (2) a concrete implementation binding the abstract interface description to an actual network address, a protocol, and concrete data structures. Because of the widespread adoption of Web Services, the technique provides a basic mechanism that can be used to exchange C-BML information. Users can invoke a service to "push" C-BML information elements and compositions of those elements so that the information can be stored for future access or acted upon by some other system. Users can also employ a service to "pull" C-BML elements or compositions of elements conforming to the XML schemas. These capabilities enable systems to readily adopt C-BML for exchanging plans, orders, and reports as proven in numerous demonstrations and development programs (e.g., [15]). To further promote initial adoption and application of the C-BML standard, the C-BML Phase 1 Specification describes a set of web service operations as a standard information exchange mechanism for C-BML.

Implementation-independent C-BML service descriptions are specified in WSDL in Annex C.1 of the Phase 1 Specification. This file provides no service binding information and no location to access the service. The abstract data types are defined based on the C-BML schema given in Annex B of the specification. The WSDL describes operations for "pushing" each of the C-BML 5W information components to the service. The WSDL assumes a SOAP (Simple Object Access Protocol) implementation for transport. It follows the document literal style for the XML content in the SOAP message body (i.e., a document style message which is not encoded).

Annex C.2 of the Phase 1 Specification provides a sample WSDL that is more implementation-specific and describes an additional operation for pulling a C-BML expression from the service. This WDSL is provided for illustration only, since no specific service implementation is required by the C-BML Phase 1 Specification.

The intent is to require that if a Web Service implementation is developed to support C-BML information exchange, then that service must, at a minimum, provide the operations specified in Annex C.1 of the specification. It is anticipated that actual implementations, as in the example in Annex C.2 of the specification, will specify other operations as well. To be clear, C-BML expressions aligned with the Phase 1 Specification can be employed without Web Service implementations; but when Web Services *are* implemented, the minimum set of operations specified must be provided to be considered compliant with this specification.

7. Phase 1 C-BML Usage Guidelines

A second part of the C-BML Phase 1 standard is a Guidelines document. At time of this writing, drafting of the Guidelines was on hold pending completion and general consensus on the content of the C-BML Phase 1 Specification. The Guidelines document will present approaches for early adoption of the standard through incremental introduction of the C-BML Phase 1 information exchange structure and content and the information exchange mechanism into current or emerging applications. If the current C-BML Phase 1 Specification content is approved, the initial Phase 1 Guidelines will likely provide some of the following initial guidance to early adopters:

- Since the Specification does not specify specific doctrinal expressions or combinations of the basic 5W information components, users will be free to employ some or all of the components in any combination. The individual elements will have to conform to the respective XML schemas, but the elements can be used in isolation or within the structure of other XML expressions. Phase 2 grammar specification will impose structure on the use of the basic C-BML information elements while also extending those base structures and adding new information elements as needed.
- Development of applications under the Phase 1 Specification for particular doctrinal expressions will likely result in identification of necessary extensions to the JC3IEDM. This will enable early execution of C-BML PSG management practices in reviewing change proposals and accepting or rejecting those proposals, with concomitant modification of the XML schemas and WSDL specifications when changes are approved.

The Phase 1 Guidelines will present a reference implementation of the specified services and provide guidelines for adaptation of existing applications to the Phase 1 Specification.

8. Issues

A number of issues have been discussed during the C-BML Phase 1 Specification drafting effort. Some of these have been addressed in the Specification to the satisfaction of the originator of the issue, while some remain open for PDG discussion and resolution. The following paragraphs provide a brief description of some of the more interesting issues that have been discussed and summarize their current disposition. At the Spring 2008 SIW, the DG will present remaining open issues to the PDG for discussion and decision (resolution at the meeting, creation of Tiger Teams to come up with recommended resolution, etc.).

8.1 Scoping the C-BML Phase 1 Specification Effort

There is some concern that the DG has approached the Phase 1 Specification too generically and that some minimal grammatical form, for some limited real-world doctrinal expressions, with an initial set of JC3IEDM extensions, must be specified or the Phase 1 Specification will be inadequate. It has been difficult to see where to draw the line – which doctrinal expression(s) (the NATO OpOrder has been suggested), how extensive a grammar (the JBML Domain Configured Schema adapted for MSG-048 experimentation has been suggested), involving what extensions to the JC3IEDM (again, those used in the JBML/MSG-048 effort are suggested) – and the DG did not believe it could make that decision given the initial direction (in particular, holding grammar specification to Phase 2). At the time of this writing, this scoping issue remains open for PDG resolution.

8.2 Commander's Intent

The main issues in representing the commander's intent arise when considering the human-to-machine or machine-to-machine interaction. Before we delve into the details of the problem, it is important to first look at a definition of the term. According to the NATO Standardized Agreement (STANAG) the commander's intent "*is a concise expression of the purpose of the operation which describes the desired end state. It should be understandable two echelons down and helps his subordinates focus on what has to be accomplished in order to achieve success so that mission accomplishment is possible in the time available and is absent of additional communications or further instructions*". While this definition makes perfect sense for human-to-human interaction, it underscores to difficulty in trying to capture the commander's intent in a machine understandable way for two main reasons:

- Machines do not need a reason or purpose in order to execute tasks or orders. Furthermore, machines must be told how to perform tasks (manually or in semi-automated fashion). Consequently, the commander's intent is inserted during simulation planning, and reflected during the execution phase.
- Machines cannot make independent decisions in the absence of communication or further instructions as the definition in the STANAG states. This further argues against the need to represent the commander's intent for machines, simply because the current state of technology does not support thinking machines in the pure sense of the term.

Based on these factors, the DG made the decision to distinguish between the C-BML Why and the commander's intent and agreed to leave the commander's intent as free text to serve in human-to-human interactions.

8.3 C-BML *Why* Component

While the decision has been made to keep the commander's intent as free text, it is important to note that its stated purpose is to describe a desired end state. While machines do not need a purpose in order to perform a task, the expression of the desired end state provides an additional informational element that needs to be captured. The desired end state can be expressed as a stopping condition for a simulation or an exit condition for a state within an entity. In addition, the desired end state as envisioned by the commander at a high level depends on the fulfillment of desired end-state at lower levels of the echelon. Otherwise stated, there is a functional dependency between higher level and lower level tasks. This dependency between tasks can also be captured in a machine understandable way in the form of alternatives, prerequisites or other conditionals. For example, we can capture that a fire event must be generated in response to a call for fire. In other words the reason for the fire was in response to a call for fire.

Based on this analysis, the DG strived to capture both aspects of the Why in order to express the desired end state of individual tasks and capture the desired end-state of a group of tasks as a functional dependence. However, representing the "Why" in a way that is completely machine understandable remains a challenge.

8.4 Existing BML Implementations

There have been two recent (2007) demonstrations of BML in a real world operationally meaningful scenario with fielded systems. The first one was the US Joint BML (JBML) demonstration that was focused on integrating Ground, Air and Maritime orders information. The second one was the NATO MSG-048 demonstration that was focused on interoperability between systems and simulations from five different nations. These demonstrations and their supporting efforts have further shown the value of a BML standard in the international community and can perhaps be adopted as the starting point for the C-BML specification. As discussed in 8.1 above, this is a departure from what the DG understood for the Phase 1 Specification, and will be addressed by the PDG in the near future. It is certainly the case that the prior work will be useful in the Phase 2 grammar specification, as that work will need to accommodate a wide variety of doctrinal expressions in specification of plans, orders, and reports.

8.5 Expressing *Reports*

There has been discussion of postponing expression of reports to Phase 2. However, the current generic approach allows 5W elements to be employed in expressions of reports, so the full "plans, orders, and reports" phrase is in the Phase 1 Specification.

8.6 C-BML for Robotic Forces

There has been discussion that we are so early in the deployment of robotic forces for military operations that specification of C-BML for robots should be postponed to later phases of specification development. However, here, too, the DG believes the current generic specification of the 5Ws as information elements allows them to be used in expressions for robotic forces. Some early work relating C-BML grammar research to a general robot command language is discussed in [16]. Even such early work indicates C-BML will be able to express plans and orders for robots, so it seems reasonable to address the full spectrum (live, constructive, and robotic forces) in the Phase 1 scope.

8.7 What-When – the Temporal Aspect of Actions and Events

One of the main issues in trying the represent temporal components independently is the fact that they are most often represented in conjunction with other concepts. In the case of the "When," a strong argument can and has been made to identify it as a stand-alone element. However in the JC3IEDM, the temporal aspects of tasking are related to the task. The same is true about reports and orders. In some cases the temporal aspect is mandatory (the status of an order); in others they are optional (task). The decision to embed the temporal aspects within specific elements points to the fact that operationally, it does not make sense to talk about temporal components outside of a specific entity. As a result, the DG decided to couple the "What" and the When into a single "What_When" element (which was also done in earlier JBML work). It is important to note that the temporal component (When) is optional in the "What_When" so this could have been referred to simply as the "What." This is still an area of discussion, but the current Phase 1 Specification continues to define the What_When construct pending specific recommendations for consideration.

9. Summary

C-BML is at an important stage in its formulation as an international standard. The draft Phase 1 C-BML Specification lays out the following standard structure, content, and practices for the C-BML:

- JC3IEDM Edition 3.1a as the underlying data model to define a common set of terms and concepts for constructing C-BML expressions;
- Established MIP practices for identifying and describing proposed changes to the JC3IEDM for C-BML applications;
- XML Schemas describing the required structure and content of the fundamental C-BML information components (the 5Ws);
- WSDL description of basic operations to be provided in Web Service implementations of C-BML information exchange mechanisms.

The draft specification provides basic building blocks for early adopters of the C-BML standard. Follow-on phases in C-BML standard development will formalize the grammar for expressing plans, orders, and reports creating greater precision in the composition of the C-BML 5Ws to encode particular doctrine, and will formalize the semantics of plans, orders, and reports to enable automated checking of the operational validity of the content of C-BML expressions. Pending resolution of issues, the DG is striving to have a complete draft specification before the 2008 European SIW in June so that it can go into balloting by the SISO community. We expect the Guidelines document to follow a short time after, targeting its balloting by or before Fall 2008 SIW in September.

10. References

[1] Simulation Interoperability Standards Organization: C-BML Study Group Terms of Reference.

[2] Simulation Interoperability Standards Organization: Coalition Battle Management Language (C-BML) Study Group Final Report, SISO-REF-016-2006.

[3] Multilateral Interoperability Programme (MIP): The Joint C3 Information Exchange Data Model (JC3IEDM), Edition 3.1a, 16 February 2007. (available at http://www.mip-site.org/publicsite/04-Baseline_3.0/JC3IEDM-Joint_C3_Information_Exchange_Data_Model/)

[4] Tolk, A., S. Diallo, and C. Turnitsa: "A System View of C-BML," Paper 07F-SIW-054, Fall Simulation Interoperability Workshop, Simulation Interoperability Standards Organization, Orlando FL, September 2007.

[5] Schade, U. and M. R. Hieb: "Development of Formal Grammars to Support Coalition Command and Control: A Battle Management Language for Orders,

Requests, and Reports, *Proceedings of the 11th International Command and Control Research and Technology Symposium*, CCRP Press, 2006.

[6] Schade, U., and M. R. Hieb: "Formalizing Battle Management Language: A Grammar for Specifying Orders," Paper 06S-SIW-068, Spring Simulation Interoperability Workshop, Simulation Interoperability Standards Organization, Huntsville, AL, April 2006.

[7] Schade, U., and M. R. Hieb: "Battle Management Language: A Grammar for Specifying Reports," Paper 07S-SIW-036, Spring Simulation Interoperability Workshop, Simulation Interoperability Standards Organization, Norfolk, VA, April 2007.

[8] Blais, C., C. Turnitsa, and P. Gustavsson: "A Strategy for Ontology Research for the Coalition Battle Management Language (C-BML) Product Development Group," Paper 06F-SIW-003, Fall Simulation Interoperability Workshop, Simulation Interoperability Standards Organization, Orlando FL, September 2006.

[9] Hopcroft, J. E., and J. D. Ullman: *Formal Languages and Their Relation to Automata*, Addison-Wesley Publishing Company, Reading MA, 1969.

[10] Multilateral Interoperability Programme (MIP): Guide to Change Proposals for MIP Data Specifications (JC3IEDM-GuideToCP-UK-DMWG), Edition 3.1a, 16 February 2007.

[11] Multilateral Interoperability Programme (MIP): JC3IEDM Annex H, Naming Conventions and Class Words (JC3IEDM-AnnexH-UK-DMWG, Edition 3.1a, 16 February 2007.

[12] Multilateral Interoperability Programme (MIP): The Joint C3 Information Exchange Data Model Metamodel (JC3IEDM-Metamodel-UK-DMWG), Edition 3.1a, 16 February 2007.

[13] Multilateral Interoperability Programme (MIP): The Joint C3 Information Exchange Data Model (JC3IEDM), Edition 3.1a, Annex G1: Compendium of Text Business Rules and Annex G2: Compendium of Coded Business Rules, 16 February 2007.

[14] Zimmermann, O., M. Tomlinson, and S. Peuser: *Perspectives on Web Services: Applying SOAP, WSDL and UDDI to Real-World Projects*, Springer-Verlag, Berlin Heidelberg, 2003.

[15] Levine, S., C. Blais, M. Hieb, J. Kearley, C. Pandolfo, M. Pullen, and J. Roberts: "Joint Battle Management Language (JBML) – Phase 1 Development and Demonstration Results," *Proceedings of the Fall 2007 Simulation Interoperability Workshop*, Simulation Interoperability Standards Organization, Orlando FL, September 1007.

[16] Davis, D., C. Blais, and D. Brutzman: "Autonomous Vehicle Command Language for Simulated and Real Robotic Forces," Proceedings of the Fall 2006 Simulation Interoperability Workshop, Simulation Interoperability Standards Organization, Orlando FL, September 2006.

Acknowledgements

The authors of this paper wish to thank the many other people who have contributed to the technical development and review of the draft Phase 1 C-BML specification and who continue to perform research and development of C-BML concepts and technical approaches. Special thanks to Gary Farmer, Kevin Gupton, Carolyn Hare, and Dick Brown for their participation in the DG efforts and to the officers of the C-BML PDG, Kevin Galvin, Per Gustavsson, Andreas Tolk, and Chuck Turnitsa, for their leadership and support through the standardization process to this point. Opinions expressed in this paper are solely those of the authors and do not necessarily represent the position of any of their respective organizations, SISO, or the C-BML PDG.

Authors' Biographies

CURTIS BLAIS is a Research Associate in the Naval Postgraduate School Modeling, Virtual Environments, and Simulation (MOVES) Institute. Mr. Blais is a Ph.D. candidate in MOVES conducting research into the application of current and emerging Semantic Web technologies to obtain valued information for military decision-makers in net-centric architectures and to improve interoperability across C2 and M&S systems.

SAIKOU DIALLO is a Ph.D. candidate at the Virginia Modeling, Analysis, and Simulation Center (VMASC) of the Old Dominion University (ODU). He received his M.S. in Modeling & Simulation (2006) from ODU. His Ph.D. research focuses on generative grammars and model-based data engineering for M&S interoperability.

GARY FARMER is a research engineer in the Signal and Information Sciences Laboratory, Applied Research Laboratories, The University of Texas at Austin. He has over 18 years' experience in data engineering, database modeling and design, and database administration in the defense community. Mr. Farmer was one of the principle designers of the US Army C4I and Simulation Initialization System (ACSIS).

STAN LEVINE is Research Professor with the Center of Excellence in C4I at George Mason University. He also serves as a senior consultant to several Army organizations in the areas of information system technologies, architectures, System of Systems acquisition, and interoperability. He has over 33 years experience in systems acquisition. He holds a BSEE and a

MS degree in Physics from Monmouth University, and a PhD in Engineering Management from Madison University. He is a recipient of the Army's three highest Civilian Service Awards. He is also a member of the Federal 100 top executives who had the greatest impact on the government information systems community.

CHUCK TURNITSA is a Ph.D. candidate at the Virginia Modeling Analysis and Simulation Center (VMASC) of the Old Dominion University (ODU). He received his B.S. in Computer Science (1991) from Christopher Newport University (Newport News, Virginia), and his M.S. in Modeling & Simulation (2006) from ODU. His Ph.D. research focuses on the domain of dynamic and fractal ontology models for M&S interoperability.