Proceedings of the 2011 Winter Simulation Conference
S. Jain, R.R. Creasey, J. Himmelspach, K.P. White, and M. Fu, eds.

APPLICATION OF COALITION BATTLE MANAGEMENT LANGUAGE (C-BML) AND C-BML SERVICES TO LIVE, VIRTUAL, AND CONSTRUCTIVE (LVC) SIMULATION ENVIRONMENTS

Curtis Blais
Modeling, Virtual Environments, and Simulation (MOVES) Institute
Naval Postgraduate School
Monterey, CA 93943, USA

ABSTRACT

Information sharing is a key requirement in Live, Virtual, and Constructive (LVC) simulation environments. Operational plans, orders, and requests from live, virtual, or constructive command and control systems or simulations need to be received by and operated on by receiving LVC systems. Situational reports from the LVC systems need to be received and interpreted or displayed by receiving LVC systems. Many simulation systems have not been developed with capabilities for robust interactions with other simulations beyond federation capabilities obtained through such protocols as the High Level Architecture (HLA) or the Distributed Interactive Simulation (DIS). The Coalition Battle Management Language (C-BML) is an emerging standard from the Simulation Interoperability Standards Organization (SISO) developed to address the need for such information sharing across real-world command and control systems and simulations in LVC environments. This paper provides an overview of the C-BML standard and describes its application to information interchange across LVC systems.

1 INTRODUCTION

Military operations and operational environments are increasingly complex, involving joint and coalition forces possessing a variety and multitude of warfighting capabilities. The environments consist of multiple operating domains—ground, air, surface, subsurface, and space. Today, the cost of sizable field exercises has risen to prohibitive levels, in addition to growing environmental and cultural restrictions on the use of real-world terrain for the conduct of field exercises. To offset these constraints, the military increasingly is turning to modeling and simulation. However, no single simulation is able to provide the breadth and depth of operational representation to meet the needs of training, analysis, experimentation, test and evaluation, and mission planning and rehearsal. Instead, the military needs to pull together multiple simulation systems that together can cover a major portion of the representational requirements for a particular use. A variety of techniques are employed to integrate the various simulations, including the use of framework and protocol standards such as the Distributed Interactive Simulation (DIS), the High Level Architecture (HLA), and the Test and Training Enabling Architecture (TENA).

Moreover, the military needs the ability to exercise a multitude of warfighting skills, from live-fire weaponry to high-level command and control decision-making. Just as a single simulation cannot represent the breadth and depth of the operational environment, neither can a single simulation support operation at the level of the individual warfighter up through higher levels of command. These needs have led to a requirement for an integrated environment consisting of live, virtual, and constructive (LVC) simulations. Live systems permit the warfighters to use their actual equipment in real-world environments. Virtual systems place warfighters in a synthetic environment where they operate simulations (or emulations) of their actual equipment. Constructive simulation systems enable warfighters to operate with and against simulated forces simulated equipment and other resources in simulated environments. The US Depart-

978-1-4577-2109-0/11/$26.00

ment of Defense is engaged in major initiatives to define an LVC roadmap (Allen, Lutz, and Richbourg 2010) for evolution of these critical capabilities.

To achieve representational and operational cohesion across the LVC environment, information must be shared across the systems. For example, operational plans, orders, and requests from live, virtual, or constructive command and control systems or simulations need to be received by and operated on by receiving LVC simulations. Reports of situations from the LVC simulations need to be received and interpreted or displayed by receiving LVC simulations. Many simulation systems have not been developed with capabilities for robust interactions with other simulations beyond federation capabilities obtained through DIS, HLA, and TENA, among others. The Coalition Battle Management Language (C-BML) is an emerging standard from the Simulation Interoperability Standards Organization (SISO) developed to address the need for such information sharing across real-world command and control systems and LVC simulation environments. This paper provides an overview of the C-BML standard and describes its application to information interchange across LVC systems.

2 COALITION BATTLE MANAGEMENT LANGUAGE

2.1 Background

Technical papers and research efforts over the past 20 years delineate a continuing need for improvement in interoperation of Command and Control (C2) and Modeling and Simulation (M&S) systems (SISO 2006). The development of digitized C2 systems and the opportunity to utilize M&S tools for course of action analysis and mission rehearsal, as well as emerging work on robotic forces, has created an increased requirement for interoperability across these systems. In addition, the move to net-centric and network-enabled operations creates new opportunities and contexts within which M&S must support the operational personnel. Military and complex civilian operations are no longer conducted by single services or organizations within a single nation. Rather, they are increasingly joint and collaborative down to the tactical level and likely to be conducted within a coalition or alliance such as the North Atlantic Treaty Organization (NATO) or United Nations (UN). This leads to a requirement for multinational interoperability and the development of standards for inter-system information exchange.

The Coalition Battle Management Language (C-BML) is an emerging standard for expressing and exchanging plans, orders, requests, and reports across (1) real-world and virtual command and control (C2) systems; (2) live, virtual, and constructive modeling and simulation (M&S) systems; and (3) robotic systems, all operating together in Coalition operations. In simple terms, the fundamental information to be conveyed by C-BML expressions consists of Who, What, When, Where, and Why (the so-called “5-Ws”).

The standards development activity is occurring under the auspices of the Simulation Interoperability Standards Organization (SISO). In accordance with recommendations of the C-BML Study Group Final Report (*ibid*), the C-BML specification is being produced in the three phases providing incremental increase in scope and application in each version. The three phases are identified in Figure 1 and described below:

- <u>Phase 1, Data Model</u>: Phase 1 of the C-BML standardization effort defines the basic data model underlying the construction of C-BML expressions (plans, orders, requests, and reports). The data model identifies a sufficient data set, using the Joint Consultation Command and Control Information Exchange Data Model (JC3IEDM) (MIP 2009) as a starting point, for expressing portions of basic expressions that can be unambiguously interpreted by C2, M&S, and robotic systems. Discussion of the data model as a basis for C-BML can be found in (Tolk & Turnitsa 2007). The Phase 1 Specification will also specify a standard for information exchange content and structure in the form of an Extensible Markup Language (XML) schema (Blais 2011), as well as a reference architecture identifying provisions to be met by conforming implementations.
- <u>Phase 2, Formal Structure (Grammar)</u>: Phase 2 of the C-BML standardization effort will extend the Phase 1 products to more completely enable unambiguous expression of plans, orders, requests, and

reports through a formalized grammar (syntax, semantics, and vocabulary). The objective is to formalize the definition of tasks, requests, and reports such that they are rigorous, well documented, and parse-able. Various grammar demonstrations and discussions relevant to C-BML can be found in Schade and Hieb (2006a), Schade and Hieb (2006b), Schade and Hieb (2007), Kunde et al. (2009), Tolk et al (2009), and Schade et al (2010).

- Phase 3, Formal Semantics (Ontology): Phase 3 will involve specification of a battle management ontology to move toward achievement of conceptual interoperability. As described in Tolk and Muguira (2003), conceptual interoperability is the highest of seven levels of interoperability from weakest to strongest capability: Level 0, No Interoperability; Level 1, Technical Interoperability; Level 2, Syntactic Interoperability; Level 3, Semantic Interoperability; Level 4, Pragmatic Interoperability; Level 5, Dynamic Interoperability; Level 6, Conceptual Interoperability. across systems. An early discussion of C-BML ontology issues can be found in (Blais, Turnitsa, and Gustavsson 2006).

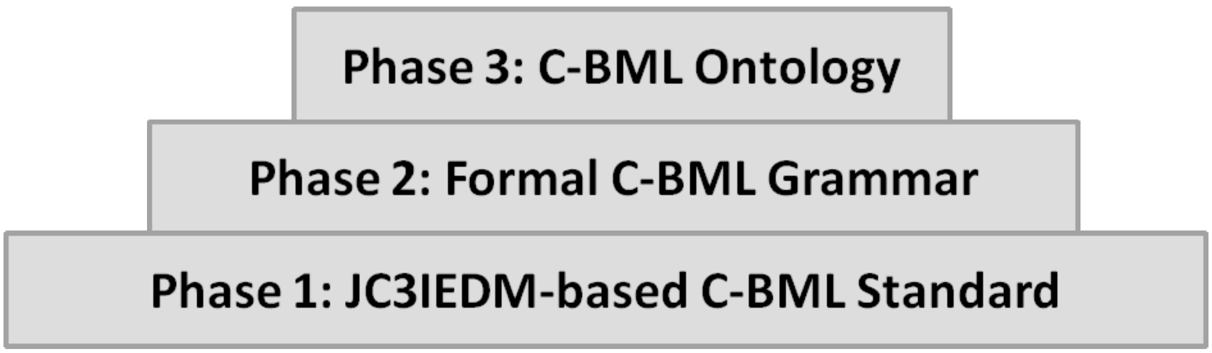

Figure 1: C-BML Development Phases

2.2 Overview

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

Figure 2: Generic System-to-System Interaction

In the C-BML context, 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 live robotics system providing situation report data to a live 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 (to that pair of systems) point-to-point information formats.

For exchange of plans, orders, reports, and requests, the C-BML concept is a standardization of the structure, content, and framework for this information exchange, as shown in Figure 3.

Figure 3: 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. Forces conducting assigned operations are required by doctrine to provide certain information. Doctrine defines what information is sent, by whom, when or under what circumstances, and to whom. 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., standardized military message formats) must be transformed into C-BML expressions, either directly or through some adaptor. In this vision, if all systems eventually adopt the C-BML standard, then only C-BML expressions would be transmitted between and among systems when transferring plans, orders, reports, and requests.

The purpose of the C-BML specification is to define a starting point for standardization of the generation of C-BML expressions. Generation of C-BML expressions depends upon two parts of the specification: (1) the C-BML information exchange structure and content; and (2) the C-BML data model. 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 the specification; namely, a reference architecture which includes rules for the exchange of C-BML expressions (see Section 3 of this paper).

2.3 Current Design Approach

The C-BML Phase 1 Specification describes the basic information content of C-BML expressions, what may be called an "operational" vocabulary or "base" vocabulary, consisting of (1) the basic 5Ws (Who-What-When-Where-Why) at an abstract level tied to the JC3IEDM logical data model; AND (2) a specialization layer providing an "operational context" to the information elements in a C-BML expression.

As abstract concepts, the 5Ws are fundamental to the expression of plans, orders, requests, and reports for any doctrine of any service, nation, or organization:

- `Who`: C-BML information component identifying the battlespace object that is directed to perform an action (plan or order), has been observed or is reporting an action (report), is requested to perform an action, provides the authority or authorization for a plan, order, request or report, or is the object of an action.
- `What`: C-BML information component identifying an action to be performed (plan, order, or request) or that has been performed (report).
- `When`: C-BML information component describing the timeframe in which an action is to occur (plan, order, or request) 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, order, or request), 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, or the desired end state of a planned action.

The 5Ws constitute a portion of the C-BML "doctrine view": expressions of plans, orders, requests, and reports using terminology particular to a specific nation, service, or organization and for particular kinds of operations. This abstraction of fundamental information components in the content of doctrinal expressions of plans, orders, requests, and reports facilitates future employment of the standard by any service, nation, or organization.

Some of the word senses for the various terms have been suggested by prior work; for example, the Command and Control Lexical Grammar (C2LG) (Schade et al 2010), Joint Battle Management Language (JBML) (Levine et al 2007), Integrated Battle Management Language (IBML), and NATO Modeling and Simulation Group 048 (MSG-048) (Pullen et al 2008). Additional terms may come out of parallel work being performed jointly by the SISO Military Scenario Definition Language (MSDL) (SISO 2008; Blais 2010) and C-BML Product Development Groups to define a common tasking grammar. There is an additional layer of specialization suggested by work such as JBML, where terms like `Taskee` can be an item of equipment or an organization, and concepts like `time` can be absolute or relative (e.g., to an H-hour). Other vocabulary that needs to be addressed for what could be called "expressional context" are constraints, controls, or restrictions (such as rules of engagement, control measures, etc.) and other conditions or performance measures (i.e., success criteria (Abbott and Goldman 2008)) important to specification of tasks.

Each "W" information component takes on a certain word sense in each expression of a plan, order, request, or report. For example, in the context of an order, one sense for `Who` is the identity of the authority giving an order (tasker), while another sense for `Who` is the identity of organization that will carry out the order (taskee). These distinctions in meaning of a "W" in a specific C-BML expression result in different semantic mappings to the underlying data model. For example, the Phase 1 Specification defines:

- the abstract `Who` specialized to terms such as `Tasker`, `Taskee`, and `Affected`;
- the abstract `What` specialized to terms associated with tasks, actions, and events;
- the abstract `When` specialized to terms such as `StartWhen` and `EndWhen`;
- the abstract `Where` specialized to modes such as absolute, relative (e.g., range and bearing from an absolute location), and indirect (e.g., unit aboard a ship);
- the abstract `Why` specialized to terms associated with concepts such as purpose, objective, desired end state, and intent.

The information exchange content and structure portion of the Phase 1 Specification is described in the World Wide Web Consortium (W3C) XML Schema language (W3C 2004). A portion of the schema is described below. Full description of the proposed XML schema is not possible within the length constraints for this paper (interested readers are invited to contact the author for a copy of the schemas providing in the Trial Use Package).

The XML Schema 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 (i.e., to ensure the format and content of an XML document containing C-BML expressions conform to the language specification described by the XML schema). Furthermore, the use of XML facilitates widespread adoption of the C-BML standard.

The C-BML XML representation of the 5Ws provides information elements for use in expressing portions of plans, orders, requests, and reports that can be exchanged across systems through a variety of mechanisms (see Section 3 for discussion of the C-BML reference architecture).

Data structures in the Phase 1 XML schemas are not intended to constrain the structure of expressions built from the data structures—such constraints are the objective of the Phase 2 effort to specify a formalized grammar for C-BML expressions. Instead, the Phase 1 schemas provide building blocks, related directly to the underlying JC3IEDM, for construction of such expressions. Top-level declarations of the 5Ws and associated "operational context" terms are shown in the XML excerpt below:

```
<!-- ************* Who Elements *************-->
<xs:element name="Who" type="cbml:WhoType"/>
<xs:element name="WhoRef" type="cbml:WhoRefType"/>
<xs:element name="ExecuterWho" type="cbml:ExecuterWhoType"/>
<xs:element name="TaskerWho" type="cbml:TaskerWhoType"/>
<xs:element name="TaskeeWho" type="cbml:TaskeeWhoType"/>
<xs:element name="ReporterWho" type="cbml:ReporterWhoType"/>
<xs:element name="RequesterWho" type="cbml:RequesterWhoType"/>
<xs:element name="RequestedWho" type="cbml:RequestedWhoType"/>
<xs:element name="AffectedWho" type="cbml:AffectedWhoType"/>
<xs:element name="AffectedWhoLight" type="cbml:AffectedWhoLightType"/>
<!-- ************** What Elements ********** -->
<xs:element name="What" type="cbml:WhatType"/>
<xs:element name="WhatRef" type="cbml:WhatRefType"/>
<xs:element name="TaskWhat" type="cbml:TaskWhatType"/>
<xs:element name="TaskWhatRef" type="cbml:TaskWhatRefType"/>
<xs:element name="EventWhat" type="cbml:EventWhatType"/>
<xs:element name="EventWhatRef" type="cbml:EventWhatRefType"/>
<!-- ************** When Elements ********** -->
<xs:element name="TaskWhen" type="cbml:TaskWhenType"/>
<xs:element name="TaskWhenLight" type="cbml:TaskEndWhenLightType"/>
<xs:element name="EventStartWhen" type="cbml:EventStartWhenType"/>
<xs:element name="EventEndWhen" type="cbml:EventEndWhenType"/>
<xs:element name="ReportedWhen" type="cbml:AbstractReportedWhenType"/>
<!-- ************* Where Elements *************-->
<xs:element name="Where" type="cbml:WhereType"/>
<xs:element name="AtWhereLight" type="cbml:AtWhereLightType"/>
<xs:element name="RouteWhere" type="cbml:RouteWhereType"/>
<xs:element name="RouteWhereLight" type="cbml:RouteWhereLightType"/>
<!-- ************* Why Elements *************-->
<xs:element name="Why" type="cbml:WhyType"/>
<xs:element name="WhyLight" type="cbml:TaskWhyLightType"/>
```

In each case, the C-BML term is declared by a specific data type that is further defined later in the schema. Consider, for example, the `TaskerWho` element. It is declared to be of type `cbml:TaskerWhoType` (note: the “cbml:” prefix is used in the XML schema document to indicate the `TaskerWhoType` is defined in the C-BML namespace). The declaration of this element is shown in Figure 4 below.

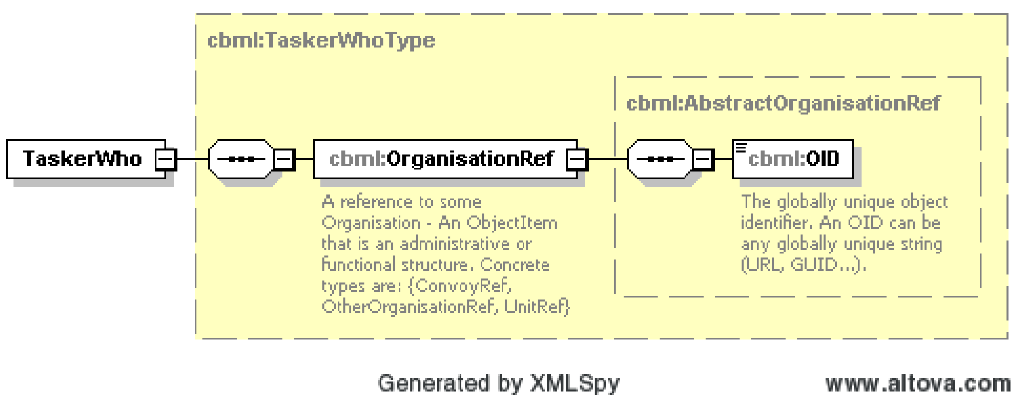

Figure 4: Declaration of the C-BML `TaskerWho` Element

As shown, the `cbml:TaskerWhoType` is a structure consisting of a `cbml:OrganisationRef` element, which is defined as type `cbml:AbstractOrganisationRef` consisting of an object identifier (OID). Elements that can be used as organizations for `TaskerWho` in C-BML expressions are identified in a separate schema file. For example, the `Unit` element declaration is shown in Figure 5.

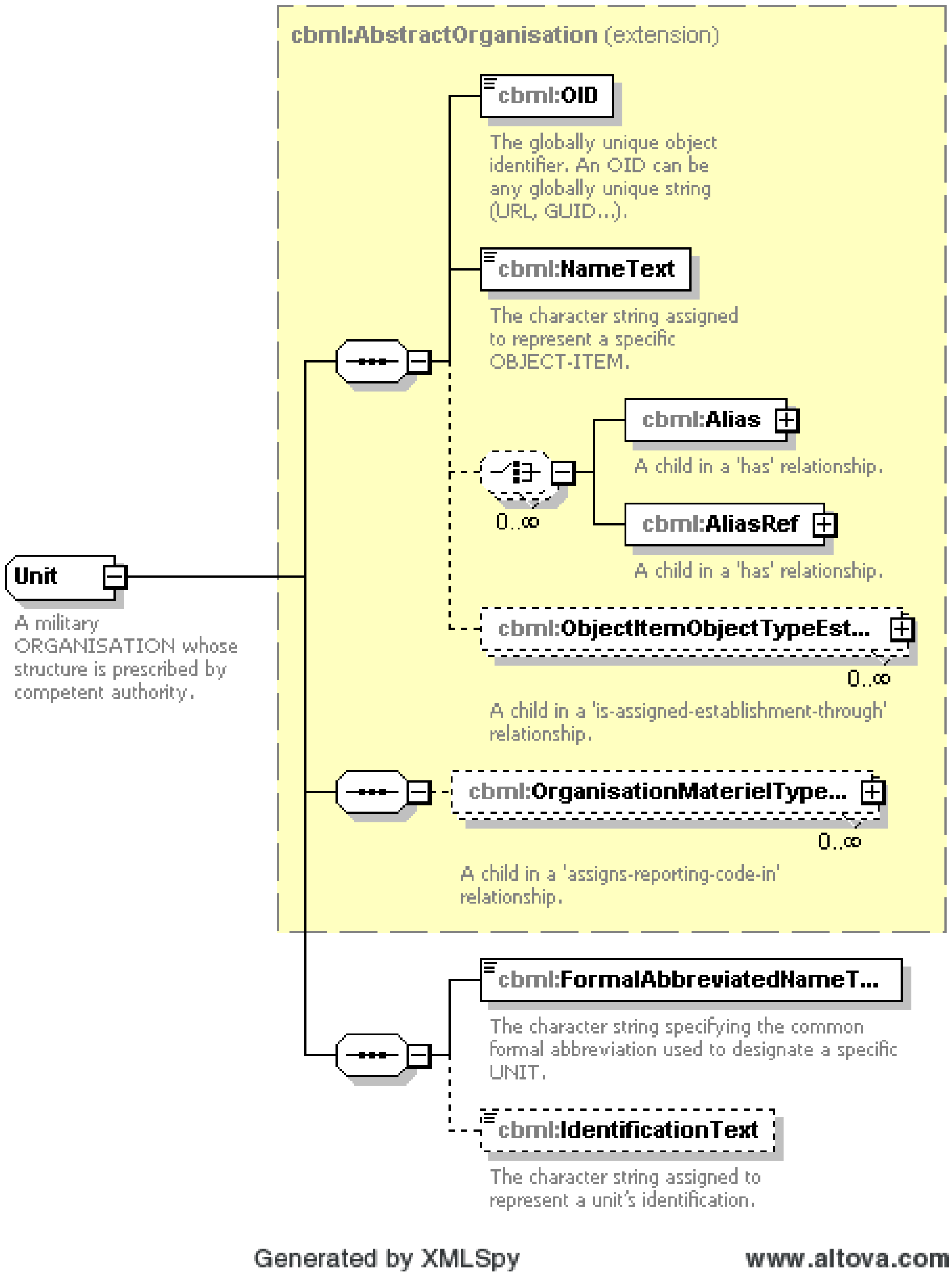

Figure 5: Declaration of the C-BML `Unit` Element

The Unit element is a concrete extension of the cbml:AbstractOrganisationType that can be referenced through the OID for use as a `TaskerWho` in a C-BML expression.

The C-BML Phase 1 schemas also specify structures for tasks that can be used in basic C-BML expressions of orders and requests. For example, the declaration for OrderTaskType is shown in Figure 6.

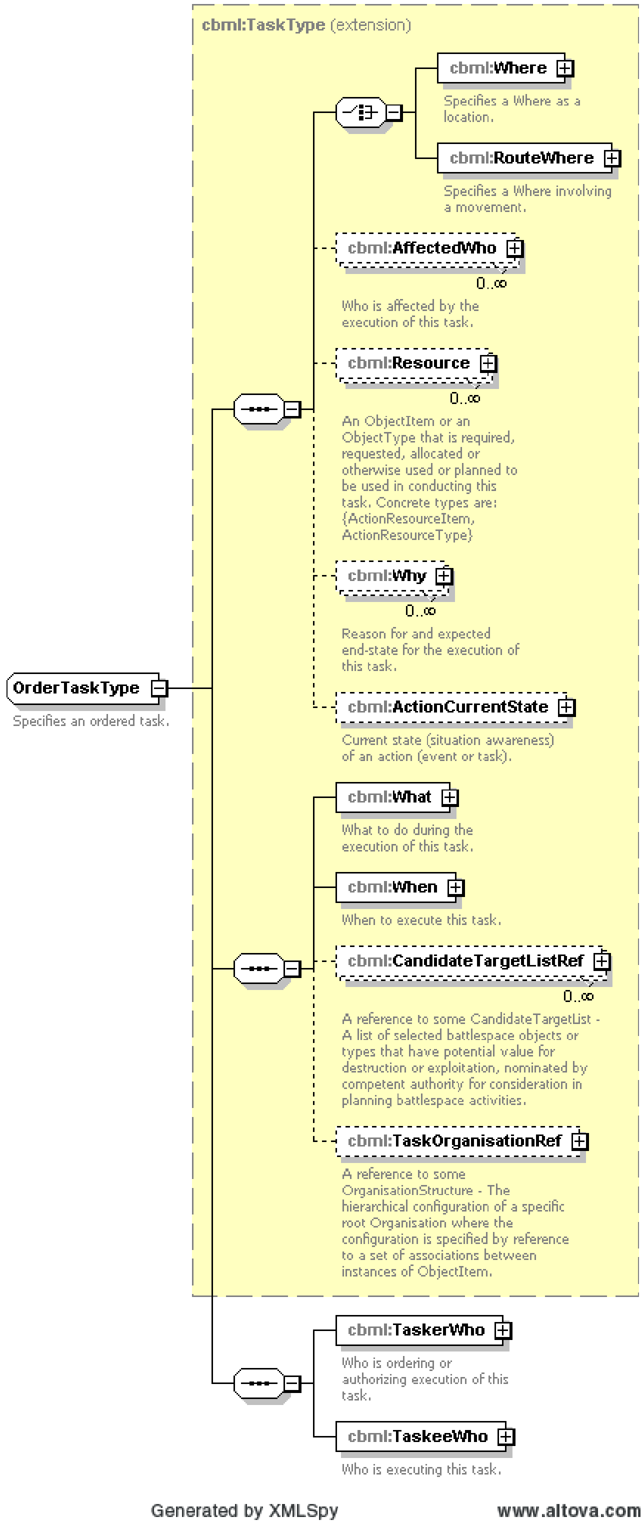

Figure 6: Declaration of the C-BML OrderTaskType

This structure shows the familiar concepts of `TaskerWho`, `TaskeeWho`, `What`, `When`, `Where`, and `Why` common to general C-BML expressions.

The above provides only a very brief introduction to the approach and content of the XML schemas from the Phase 1 C-BML Specification development. Interested readers are encouraged to contact the author for more information.

3 SPECIFICATION: REFERENCE ARCHITECTURE

A common practice for implementing a battle management language information exchange mechanism employs web services specified in the Web Services Description Language. Implementation (by any service, nation, or organization) of C-BML applications conformant to the Phase 1 specification will require transformation of respective information elements in current expressions (e.g., textual or binary message formats), some of which may already use defined XML tag sets, into the C-BML XML structures. Legacy systems will generally 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 certain information elements in plans, orders, requests, and reports.

Rather than specifying a particular information exchange mechanism, as originally proposed for the C-BML standard, the SISO C-BML Product Development Group determined that it would be better to describe a reference architecture to which implementations should conform in order to support seamless integration of multiple systems employing C-BML. Early adopters of C-BML are encouraged to use existing standards and tools that best suit their particular needs. However, it is possible to define a set of rules (provided below) that, if followed, can increase the level of interoperability of systems that exchange C-BML-compliant expressions (note: a C-BML compliant expression is an XML document that contains only types that are specified in the C-BML information exchange structure and content specification and that obeys the business rules specified therein).

A system that produces and/or consumes valid C-BML expressions can send and/or receive these expressions using many different architectures or information exchange mechanisms (IEM). The exchange of C-BML expressions across systems that produce and consume valid C-BML expressions is not limited to a specific IEM. For illustrative purposes, a Simple Mail Transfer Protocol (SMTP) email-based exchange mechanism is perhaps the simplest IEM that can accomplish this objective. Other possible IEMs include but are not limited to: High-Level Architecture (HLA), the Object Management Group (OMG) specified Data Distribution Service (DDS), the Multilateral Interoperability Programme (MIP) Data Exchange Mechanism (DEM), WC3 Web Services using Simple Object Access Protocol (SOAP), and Representational State Transfer (REST) services over Hypertext Transfer Protocol (HTTP). In fact, C-BML expressions can be exchanged either synchronously or asynchronously and, furthermore, the orchestration and execution of systems using C-BML constructs is beyond the scope of the C-BML standard.

While a specific architectural description is outside the scope of the C-BML standard, it is still useful to provide a depiction for a generic C-BML message exchange architecture or "reference architecture" in order to establish a common terminology and framework. Based on this description and terminology, a set of rules are proposed such that independent C-BML messaging infrastructures will be able to interoperate with minimal integration efforts.

The proposed reference architecture is shown in Figure 7 as a layered architecture representation loosely based on the Open Systems Interconnection (OSI) stack and the Transmission Control Protocol / Internet Protocol (TCP/IP) model, which identifies Network Access, Internet, Transport, and Application layers. Figure 7 illustrates the various entities and relevant specifications involved in the exchange of C-BML expressions through the use of a C-BML messaging infrastructure. System A represents a C-BML producer. Consuming systems receive the C-BML expressions over the network. In this depiction, an "expression" is the Information Exchange Content and Structure-compliant payload of a C-BML message, which is specific to the messaging infrastructure. Note that the C-BML Messaging Interface must comply with C-BML Services Specification that dictates high-level functionality such as validation, error handling, message receipt acknowledgement, etc. A services specification is not currently in the scope of the Phase 1 Specification.

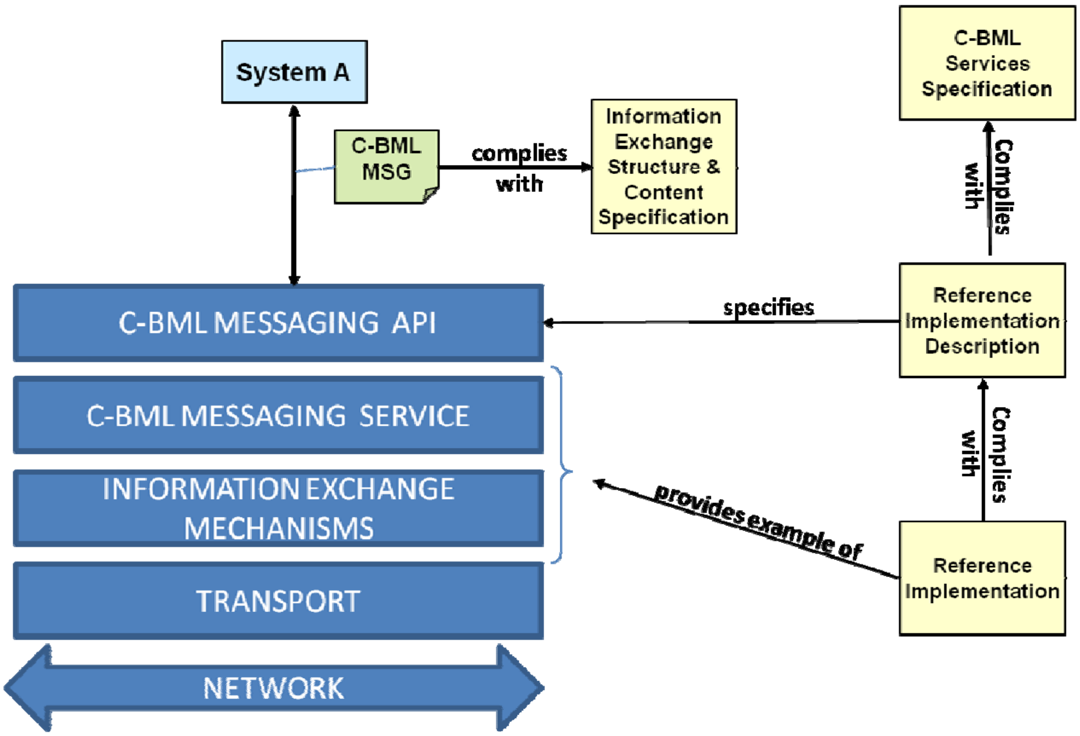

Figure 7: C-BML Reference Architecture Stack

The architecture allows for execution environments that employ different transport and network mechanisms. As shown, implementations must specify the C-BML messaging interface layer and provide an example "instantiation" of the C-BML Messaging Service and Information Exchange Mechanism layers.

In the case of C-BML, the figure illustrates that the C-BML messaging service will ultimately expose an interface that is compliant with the C-BML Services Specification while the underlying messaging service will utilize an IEM that can be based on various transport mechanisms and network topologies. For example, in the case of autonomous unmanned systems, this may include wireless networks characterized by highly variable communication quality.

Even if the IEM that is used to exchange C-BML expressions is independent of the normative specification that dictates how to construct valid C-BML expressions, it is still useful to establish a set of rules in order to ensure that C-BML expression producers and consumers can exchange expressions effectively. Applicable rules may include:

- Rule 001: The exchange of C-BML expressions shall not change, modify, or otherwise alter the content and structure of said expressions. Any additional elements required to exchange C-BML expressions that are specific to a given implementation and/or a given IEM are not considered to be normative and therefore should be able to be removed and/or ignored when processed by other systems and/or disseminated using another IEM.
- Rule 002: C-BML expressions shall be independent from the IEM or the architecture in which they are used. For example, HLA Time Management or Data Distribution Management data elements should not be included as part of the C-BML expression since the elements are not present in all IEMs and cannot be generalized to all architectures.

- Rule 003: The exchange of C-BML expressions shall be lossless. C-BML expressions that are sent by a system shall be received in their entirety without modification. However, receiving systems are not required to deal with all information in an expression.
- Rule 004: All C-BML expressions must be valid with respect to the C-BML schema and business rules.

The Phase 2 C-BML effort is revisiting community requirements for C-BML and will address this issue in more detail. The NATO MSG-048 Technical Activity Final Report (NATO 2010) is a good source of C-BML "infrastructure" requirements and is providing a starting point for Phase 2 requirements activities for further advancement of the C-BML specification.

4 CONSIDERATIONS FOR USE IN LVC CONTEXT

It is interesting to note that current implementations of runtime interoperability have largely dealt with (1) "reporting" information between C2 and M&S systems; i.e., conveying a synchronized common operational picture for situational awareness between C2 and M&S systems; and (2) conveying "ground truth" between instrumentation and M&S systems. The other types of utterances (plans, orders, requests) are hardly addressed in today's integrations. The principal integration frameworks deal with state information, not command information. "Reports" in C-BML enable the passing of state data (ground truth or perceived). However, the overall syntactic construct meets the larger need of expression of plans, orders, and reports to create greater opportunity and capability in interactions across LVC systems.

5 SUMMARY

C-BML is an emerging standard that will provide a common information model for interchanging plans, orders, reports, and requests across LVC systems. As the draft standard approaches balloting in the modeling and simulation community, several organizations are conducting trial use to examine its robustness and completeness. Others interested in participating in this activity are invited to contact the author for more information. Only through community efforts investigating the application of the language across a variety of LVC use cases can a fully functional standard be developed that will benefit the widest set of users.

ACKNOWLEDGMENTS

The opinions expressed in this paper are those of the author and are not necessarily the position of NPS or sponsoring organizations. Portions of this paper have been adapted from the Phase I C-BML Specification, currently in draft form under development by the C-BML Phase I Drafting Group of the C-BML Product Development Group in the Simulation Interoperability Standards Organization. The author of this paper is Co-Lead Editor of the Phase I C-BML Drafting Group and co-author of portions of the draft specification used for this paper. The author acknowledges other members of the Phase I C-BML Drafting Group, principally Dick Brown, Sidney Chartrand, Saikou Diallo, and Kevin Heffner, with respect to the portion of the draft specification used for this paper.

REFERENCES

Abbott, J., and R. Goldman. 2008. "Task Based Approach to Planning." Paper 08F-SIW-033. In *Proceedings of the Fall Simulation Interoperability Workshop*. Simulation Interoperability Standards Organization. Orlando, FL. September.

Allen, G. W., R. Lutz, and R. Richbourg. 2010. "Live, Virtual, Constructive Architecture Roadmap Implementation and Net-Centric Environment Implications." *International Test and Evaluation Association (ITEA) Journal*, 31, 355-364.

Blais, C., C. Turnitsa, and P. Gustavsson. 2006. "A Strategy for Ontology Research for the Coalition Battle Management Language (C-BML) Product Development Group." Paper 06F-SIW-003. In *Proceed-*

ings of the Fall Simulation Interoperability Workshop. Simulation Interoperability Standards Organization. Orlando FL.

Blais, C. 2010. “Integrating the Coalition Battle Management Language (C-BML) into the Military Scenario Description Language (MSDL).” Paper 10S-SIW-003. In *Proceedings of the Spring Simulation Interoperability Workshop*. Simulation Interoperability Standards Organization. Orlando, FL.

Blais, C. 2011. “Coalition Battle Management Language (C-BML) Phase 1 Specification: Road to Balloting.” Paper 11S-SIW-001. In *Proceedings of the Spring Simulation Interoperability Workshop*, Simulation Interoperability Standards Organization. Boston MA.

Kunde, D. T., Orichel, A. Tolk, U. Schade, and M. Hieb. 2009. “Harmonizing BML Approaches: Grammars and Data Models for a BML Standard.” Paper 09S-SIW-046. In *Proceedings of the Spring Simulation Interoperability Workshop*. Simulation Interoperability Standards Organization. San Diego, CA.

Levine S., J. Pullen, M. Hieb, C. Blais, C. Pandolfo, J. Roberts, and J. Kearley. 2007. “Joint Battle Management Language (JBML) – Phase 1 Development and demonstration Results.” Paper 07F-SIW-051. In *Proceedings of the Fall Simulation Interoperability Workshop*. Simulation Interoperability Standards Organization. Orlando, FL.

MIP (Multilateral Interoperability Programme). 2009. The Joint C3 Information Exchange Data Model (JC3IEDM Main), Edition 3.0.2, 14 May. http://www.mip-site.org/publicsite/04-Baseline_3.0/JC3IEDM-Joint_C3_Information_Exchange_Data_Model/JC3IEDM-Main-DMWG-Edition_3.0.2_20090514.pdf

NATO (North Atlantic Treaty Organisation). 2010. Research and Technology Organisation: Modelling and Simulation Group 048 (C-BML) Technical Activity Final Report. RTO-TR-MSG-048. May.

Pullen, J. M., M. Hieb, U. Schade, K. Kruger, M. Frey, and T. Orichel. 2008. “Enabling the MSG-048 Multinational Demonstration 2007 with the Command and Control Lexical Grammar and JBML Web Services.” Paper 7. In *Proceedings of the NATO Research and Technology Organisation Modeling and Simulation Group Conference*. Vancouver.

Schade, U. and M. Hieb. 2006a. “Development of Formal Grammars to Support Coalition Command and Control: A Battle Management Language for Orders, Requests, and Reports.” In *Proceedings of the 11th International Command and Control Research and Technology Symposium.* CCRP Press.

Schade, U., and M. Hieb. 2006b. “Formalizing Battle Management Language: A Grammar for Specifying Orders.” Paper 06S-SIW-068. In *Proceedings of the Spring Simulation Interoperability Workshop*. Simulation Interoperability Standards Organization. Huntsville, AL.

Schade, U., and M. Hieb. 2007. “Battle Management Language: A Grammar for Specifying Reports.” Paper 07S-SIW-036. In *Proceedings of the Spring Simulation Interoperability Workshop*. Simulation Interoperability Standards Organization. Norfolk, VA.

Schade, U., M. Hieb, M. Frey, and K. Rein. 2010. Command and Control Lexical Grammar (C2LG) Specification. Technischer Bericht ITF/2010/02. Fraunhofer FKIE. July.

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

SISO. 2008. Standard for: Military Scenario Definition Language. SISO-STD-007-2008, 14 October.

Tolk, A., and J. A. Muguira. 2003. “The Levels of Conceptual Interoperability Model (LCIM).” 03F-SIW-007. In *Proceedings of the Fall Simulation Interoperability Workshop*. Simulation Interoperability Standards Organization. Orlando FL.

Tolk, A., S. Diallo, and C. Turnitsa. 2007. “A System View of C-BML.” Paper 07F-SIW-054. In *Proceedings of the Fall Simulation Interoperability Workshop*, Simulation Interoperability Standards Organization, Orlando FL.

Tolk, A., C. Blais, S. Diallo, and C. Turnitsa. 2009. “Conceptual Requirements for Command and Control Languages.” Paper 09F-SIW-075. In *Proceedings of the Fall Simulation Interoperability Workshop*. Simulation Interoperability Standards Organization. Orlando, FL.

W3C (World Wide Web Consortium). 2004. XML Schema Part 0: Primer Second Edition. W3C Recommendation. 28 October. http://www.w3.org/TR/xmlschema-0/

AUTHOR BIOGRAPHY

CURTIS BLAIS is a Research Associate in the Modeling, Virtual Environments, and Simulation (MOVES) Institute at the Naval Postgraduate School. He has over 37 years' experience in modeling and simulation instruction, management, design, development, and application. Current projects and research interests include data modeling for development of a virtual environment for evaluation of emergency management policies and procedures, design and validation of irregular warfare models, and specification and evaluation of the Coalition Battle Management Language (C-BML). Mr. Blais has a B.S. and M.S. in Mathematics from the University of Notre Dame. He is a member of the Military Operations Research Society (MORS), the Simulation Interoperability Standards Organization (SISO), the Association for the Advancement of Artificial Intelligence (AAAI), the Association for Computing Machinery (ACM), and the Institute of Electronic and Electrical Engineers (IEEE). His email address is clblais@nps.edu.