A Systems Engineering Approach to M&S Standards Development: Application to the Coalition Battle Management Language

Andy Bowers
The MITRE Corporation
Hampton, VA 23666
757-766-4582
bowersa@mitre.org

Curtis L. Blais
MOVES Institute
Naval Postgraduate School
Monterey, CA 93943
831-656-3215
clblais@nps.edu

Kevin Galvin
Thales
UK BTC
Manor Royal
Crawley
West Sussex, RH10 9HA, UK
+44 1293 580353
kevin.galvin@uk.thalesgroup.com

Kevin Gupton
Applied Research Laboratories
The University of Texas at Austin
10,000 Burnet Rd.
Austin, Texas 78758
512-835-3679
kgupton@arlut.utexas.edu

Kevin Heffner
Pegasus Research & Technologies
PO Box 47552
CP Plateau Mont-Royal
Montreal (QC) H2H 2S8 Canada
514-600-0141
k.heffner@pegasim.com

Lionel Khimeche
Délégation Générale pour
l'Armement (DGA)
16 bis, avenue Prieur de la côte d'or
94114 ARCUEIL CEDEX France
+33 1 79 86 45 46
lionel.khimeche@dga.defense.gouv.fr

Hakan Savasan
Turkish Navy Research Center
Command
Ankara Cadesi No:252 P.K. 252
Pendik-Istanbul, Turkey
+90 216 493 82 83
savasan.h4464@dzkk.tsk.tr

Keywords:

Systems Engineering, Requirements Management, Enterprise Architecture, UML, Standards, Interoperability, C-BML

ABSTRACT: *The development of interoperability standards can facilitate communication among information systems by defining a common way to exchange information. These standards are in fact comprised of normative and informative products that typically specify the details and examples that enable heterogeneous systems produced by different organizations to be integrated successfully and then to interoperate, as per system requirements.*

Identifying and managing such requirements is a key element to building successful standards – those that ultimately are adopted, utilized and meet stakeholder expectations. The systems engineering approach is grounded in the world of requirements and can be applied to M&S interoperability standards development in order to ensure that these standards are indeed successful. The first part of this paper considers some cases where systems engineering principles have been applied to technical standards. The second part then illustrates how this approach is being applied to the development of the Simulation Interoperability Standards Organization (SISO) Coalition Battle Management Language Phase 2 Products with significant contributions from the North Atlantic Treaty Organization (NATO) Modeling and Simulation Group 085 (MSG-085): Standardization for C2-Simulation Interoperation.

1. Introduction

"*A standard is a document that establishes engineering and technical requirements for products, processes, procedures, practices and methods, and has either been decreed by authority or adopted by consensus.*" [1] Standards development organizations (SDO) produce products such as technical specifications and other supporting documentation for the purposes of guiding and/or constraining system development, integration and maintenance or other aspects of a system's life-cycle. These products are not the end-user system, but rather provide assurance that the end-user system will possess certain characteristics (i.e. functionality and quality factors) and thus meets stakeholder expectations. In fact, system *designers* or *developers* generally are the primary users of technical standards products. Therefore, the *system* users and the *standard* users form two distinct user groups. This can present challenges for SDO: How can one ensure that end-user/stakeholder requirements are consistent with the standards users' technical perspective when these two groups likely represent different organizations with different underlying motivations? Technical standards often are requested and/or sponsored by government while industry typically acts as solution providers. Proper communication and coordination among government and industry representatives therefore is critical.

In addition, the development of new technical standards often is influenced or even triggered by the availability of emerging technologies that offer potential benefits to stakeholders. Several authors have identified deficiencies in traditional systems engineering approaches regarding the proper management of changing requirements associated with emerging technologies and/or of evolving operational requirements and stakeholder needs [2][3][4][5][6]. All of these authors prescribe the use of so-called *agile*, iterative system and software engineering processes that address many of these deficiencies. However, comparable methodologies do not exist for *standards* development processes. Nonetheless, reference [3] describes the benefits of applying an agile systems engineering approach for the development of interoperability[1] standards in the Transportation sector.

Lang et al [6] propose an enterprise architecture approach for developing the next generation (i.e. block 4) Multilateral Interoperability Programme (MIP) interoperability solution that builds on the NATO Architectural Framework (NAF) [11]. Gupton and Heffner [9] propose a Standards Development Framework (SDF) for the SISO Coalition Battle Management Language (C-BML) that is based on a similar approach to the one defined by the US Intelligence Community/DoD for a Keyword Query Language Specification [10]. Consistent with [6], the C-BML SDF approach also embodies the enterprise architecture and agile systems engineering methodologies.

This paper discusses experience and lessons learned through a first use of the C-BML SDF, implemented as a Unified Modeling Language (UML) collaborative workspace, including feedback from the NATO Modeling and Simulation Group 085 (MSG-085): Standardization for C2-Simulation (C2-SIM) Interoperation.

1.1. Coalition Battle Management Language (C-BML)

SISO currently is developing C-BML, a standardized formal language for the exchange of digitized military information among command and control, simulation and autonomous systems. Initiated in 2006 with the formation of the C-BML Product Development Group (PDG), SISO's development of C-BML has proven to be a difficult task, as witnessed by the time it has taken to produce an initial balloted Phase 1 specification [2], described in more detail in section 3.2.2.

1.2. Document organization

Following the introduction, section 2 introduces basic systems engineering concepts and reviews several systems engineering or related processes; section 3 describes some of the challenges associated with international technical standards development; section 4 defines the approach advocated by this paper; section 5 provides examples of how the approach has been applied to the C-BML development; and section 6 provides conclusions.

2. Systems Engineering Processes

The term "Systems Engineering" (SE) can be traced back to Bell Telephone Laboratories (circa 1940) while the concepts date back to the 1900s [12]: "…[SE] *has emerged from the post World War II military-industry-academic complex that was embroiled in an accelerating weapons race*…" [1].

2.1. The Systems Engineering Vee Model

The SE Vee Model is more than 20 years old and has been used and re-used in a variety of derived SE methodologies, including iterative approaches, system of system (SoS) approaches, family of systems (FoS), and dual V-Model [13].

The basic seven SE elements comprising the Vee model are shown in Figure 1, although the exact terms have been modified slightly from the original model and generalized for use with software systems. The Vee model is not a standard, but it embodies various SE processes, the simplest of which is an improved or extended waterfall method[2], originally introduced in 1970 as a sequential software engineering methodology [14].

---

[1] NATO definition: "The ability to act together coherently, effectively and efficiently to achieve Allied tactical, operational and strategic objectives."[7].

[2] http://www.waterfall-model.com/v-model-waterfall-model/

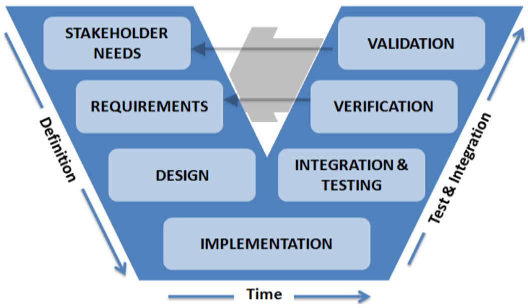

Figure 1-Systems Engineering Vee Model

The waterfall model assumes that requirements do not change during the development process. Although more flexible than the waterfall model, the basic Vee model still has several flaws, and the sequential nature of the activities still is present as a linear progression through the following phases: 1) definition; 2) implementation and 3) integration and testing, with stakeholder needs and requirements definitions activities cross-connected with validation and verification activities, respectively.

2.2. The Iterative Systems Engineering Vee Model

To remediate the basic sequential nature of the Vee Model, the iterative Vee Model, incorporates several "Vee" iterations within each engineering phase, as illustrated in Figure 2 taken from reference [15].

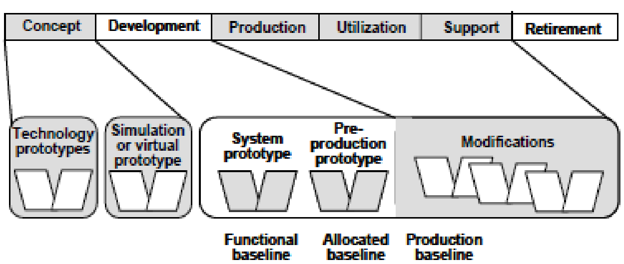

Figure 2-ISO/IEC 15288 Engineering View [15]

The main advantage of the iterative Vee model is that it maintains the rigor and traceability of the Vee model while introducing the flexibility and other benefits of iterative, incremental methodologies. Though the iterative Vee model supports changes in requirements while enabling traceability, Requirements Engineering has emerged as a key component of Systems Engineering and is deserving of further amplification.

2.2.1. Requirements Engineering

The discipline of Requirements Engineering (RE) is traditionally a software engineering process with the aim of identifying, analyzing, validating and documenting system requirements. An integral part of SE, it involves the following requirements activities: elicitation; analysis; documentation; validation and management. It also is particularly relevant to the development of standards. Proper RE assumes that requirements may change over time and should allow distinguishing characteristics such as: description, notes, priority, owner, status, complexity, version, phase etc.

Agile software development methodologies also have RE activities, but software quality factors and non-functional requirements are not always well-handled [17]. Software quality factors include considerations such as maintainability, usability, reliability, efficiency, and portability [18].

2.2.2. Sustaining versus Disruptive Requirements

Short-term and long-term requirements can be collected through requirements elicitation, especially if emerging technologies are considered. By definition sustaining engineering or sustaining requirements aim to improve existing processes, products and tools while disruptive technologies or requirements imply new capabilities or new concepts of employment or new concepts of operation, as illustrated in Figure 3.

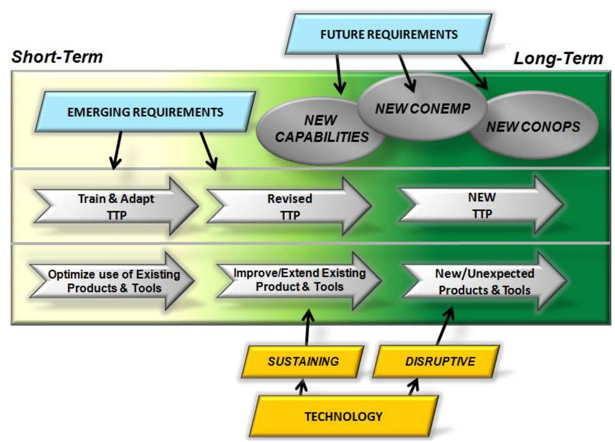

Figure 3 - Sustaining vs. Disruptive Requirements & Technology

In the military domain, short-term *sustaining* requirements pertain to existing or slightly revised Tactics, Techniques and Procedures (TTP) whereas long-term requirements for new capabilities involving disruptive technologies generally require significantly modified or new TTP. In the case of the SISO C-BML development activity, both sustaining and disruptive requirements have been proposed, yet it is not clear that all stakeholders are aware of the sustaining and/or disruptive nature of these requirements. For example, some requirements imply self-synchronization and fundamental changes in the commander's role and the manner in which operations are conducted, while other requirements seek to optimize existing command post training activities by replacing swivel-chair operators by automated systems [16].

Such ambiguity reinforces the need to manage requirements so that they can be analyzed, organized and subsequently validated by stakeholders. The diversity of stakeholders increases the probability of requirements conflicts that need resolution before standards development can occur.

2.2.3. Traceability of Requirements

Traceability of requirements is at the heart of development practices for the aircraft industry, as specified by the aircraft industry so-called airworthiness standards, such as DO-178: *Software Considerations in Airborne Systems and Equipment Certification* [20]. A distinguishing feature of DO-178 compliant software development processes is that traceability from system requirements to all source code typically is required.

For the technical standards development, requirements management helps to clarify aspects such as their relative importance, urgency, priority, etc. and thus facilitates the elaboration of standards products development plans. The ability to link elements of technical standards back to derived and operational requirements also helps to understand why the standard was constructed in a certain manner. Moreover, as requirements for standards evolve over time, the link between elements of the standard and the requirements becomes an invaluable part of a managed change request process. Otherwise, how does one know whether a specific change can be applied without *breaking* the standard, i.e. causing provisions to become inconsistent? That is to say, how can one be sure that proposed changes will satisfy new requirements while satisfying existing requirements?

Enterprise Architecture requirements management approaches now are integrated into UML tools and provide the means for ensuring traceability of requirements [6][21].

3. Challenges in International Technical Standards Development

Technical standards may address different types of methods, processes, practices and conventions and are developed for a variety of reasons: cost-reduction; lead-time reduction; increased security, safety etc. A technical standard may be based on a proven solution to a specific problem that forms the basis for a *de facto* standard. Obviously, the key to establishing a de facto standard lies in obtaining a consensus from stakeholders. In other instances, the need for the standard may come about without agreement among stakeholders on the use of an existing solution. If no commonly accepted solution exists that is satisfactory to all stakeholders, the basis for the standard therefore must be derived from stakeholder requirements. This is the case for C-BML and this has led to many challenges [2].

The greater the diversity of the stakeholders, the greater the difficulty in establishing a coherent set of requirements and developing a standard that meets all stakeholder expectations. In the case of C-BML, stakeholders represent a diverse community composed of scientists, engineers, program managers, military personnel and government stakeholders from many nations, domains and backgrounds. This diversity may lead to conflicts of interest. For example, national C2-SIM interoperability objectives for some nations are based on short-term improvements or *sustaining* engineering developments, while other nations consider C-BML technologies as an integral of a major *disruptive* transformation activity. This creates specific problems with respect to: 1) agreeing on the scope and a set of common requirements for the standard; 2) establishing priorities of requirements; and 3) determining the best technical approach for the standards.

This reinforces the need for a well-defined systematic process with special attention to requirements management and also for a process that provides a flexible, adaptable means for evolving the solution in an iterative manner to meet short-term needs while considering long-term needs and catering to changes and obstacles throughout the standard life-cycle.

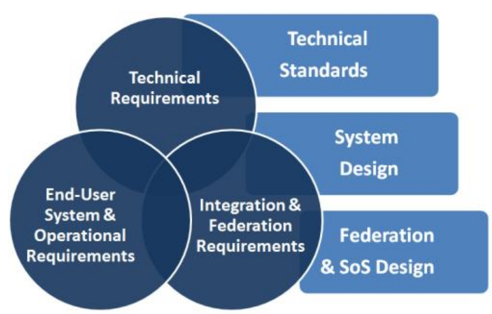

Figure 4-Requirements for Technical Standards Development and System/Federation Design

3.1. Requirements for Technical Standards

Figure 4 illustrates the relationship between requirements and three levels of system specifications: *standards*; *system design*; *SoS/Federation design*. The inputs into the definition of these specifications include technical requirements; end-user and operational requirements; and integration requirements.

However, as part of the requirements elicitation activity, stakeholders will provide all types of requirements and thus, at times, it can be difficult to extract out only the subset of technical requirements that is relevant to the standard. This has proven to be particularly challenging for the C-BML PDG [2].

3.2. Stakeholder Balance

Technical standards can be initiated by the advent of new technologies that sometimes have not reached full maturity. In such cases, the research community and academia, which form one group of stakeholders, can provide valuable guidance to ensure that a sufficient theoretical foundation is laid to apply new technologies properly. As in the case of C-BML, government often represents the main group of end-users and the stakeholders that will specify the use of the standard as part of a procurement process. Government stakeholders provide the primary set of requirements and stakeholder expectations. The third group of stakeholders is from industry; those that apply the standard as part of the system design and development. Industry provides the benefits of practical experience associated with proven solutions.

One of the significant challenges related to technical standards development is to ensure that these three groups of stakeholders are represented properly and are able to coordinate effectively [3]. This must occur throughout the standard product development process and not solely during the initial or final phases.

3.2.1. Case Study 1 – US DOT Intelligent Transportation Systems (ITS) Standard

The US Department of Transportation (DOT) initiated the development of a standard for Intelligent Transportation Systems (ITS) in response to the increased saturation of the roads networks [3]. The initial standard development activity suffered several setbacks due to sub-optimal participation and coordination among stakeholders. In particular:

- The scope failed to capture the full set of stakeholder needs;
- The standard had significant technical deficiencies; and
- The standard was difficult to understand and to apply.
- A subsequent analysis revealed the following root causes:
- Inadequate involvement from end-users;
- Lack of systems expertise in the standards drafting group;
- Most influential members of the standards committee were from industry, not from the end-user group;
- Initial process focused too much on developing the solution without establishing requirements;
- Insufficient validation of stakeholder requirements.

Not only the first version of the standard did not reflect well-understood user needs, but also there seemed to be no guarantee that future revisions would address unmet needs. Fortunately, remedial measures were taken to ensure increased involvement from government stakeholders and included a requirements management activity that provided for verification and validation (V & V) of stakeholder requirements.

These measures included:

- A CONOPS document that focused on identifying the user needs as high-level requirements;
- Ensuring that standard development addressed these high-level requirements;
- A mechanism for testing products that claimed conformance;
- Identifying the quality factors important to the standard;
- Ensuring that, once derived requirements were established, end-user stakeholders were part of validation;
- A mechanism for rapidly updating both requirements and the resulting standard(s) in an incremental way;
- V & V must occur at each major iteration, for completeness, correctness of the CONOPS, requirements and the design (i.e. the standard itself).

Review by external users also was cited as an important aspect.

3.2.2. Case Study 2 – Lessons Learned from C-BML Phase 1 Standard Development

Formed in 2006, the C-BML PDG has taken many years to produce the Phase 1 product. Reference [2] offers the following reasons:

- Process:
  - Involved many stakeholders from different backgrounds, organizations, nations, locations, leading to coordination difficulties;
  - No deliberate (spiral) process was defined or followed; a waterfall type approach implicitly was employed;
  - A well-defined process to properly capture and track stakeholder requirements was lacking;
  - No Change Control Authority was defined;
- No lead architect, capability owners or team leaders were defined; and

- Large scope; no capability packages or other modularity was defined;
- No formal data model was specified;
- Inadequate resources were available.

The remainder of this paper presents the Systems Engineering Enterprise Architecture Approach for standards development approach that was developed as preliminary work conducted by the C-BML Phase 2 Drafting Group [9]. The approach has been implemented as an UML-based distributed, collaborative workspace that was made available to the MSG-085 Technical Group in support of their 2012 Common Interest Group (CIG) experimentation activities. Results from this work will be shown in the subsequent sections.

4. Systems Engineering Enterprise Architecture Approach to Standards Development

Several architectural frameworks[3] (AF) exist for specifying the *operational context* of military *capabilities* in the form of a *Military Enterprise Architecture* that is understandable by all stakeholders and includes the means to describe requirements, information flows, organizations, processes, interfaces, data, protocols etc.). The approach outlined in this paper for the development of an international standard is based, in part, on the NATO Architecture Framework (NAF), that seems to be the appropriate choice for an international standard [11]. Other frameworks include the US Department of Defense Architecture Framework (DODAF) and the UK Ministry of Defence Architecture Framework (MODAF). All frameworks provide a number of views or viewpoints that can be used that facilitate system design. The UK's MoDAF contains seven views, at the highest level are the Strategic Views (StV) and the others include; Acquisition Views (AcV) Operational Views (OV), System Views (SV), Service Views (SOV) and Technical Views (TV) as illustrated in Figure 5.

DODAF Version 2.0 has renamed views to viewpoints and includes many more including a Project Viewpoint (PV) and the TV in MODAF now is called the Standards Viewpoint (StdV). The change in terminology was in order to align with ISO Standards. Revision 3 of the NAF, promulgated in November 2007, is identical to MODAF at its core, but extends the framework by adding views for Bandwidth Analysis, SOA and standard configurations. The seven views are:

- NATO All View (NAV);
- NATO Capability View (NCV);
- NATO Operational View (NOV);
- NATO Service-Oriented View (NSOV);
- NATO Systems View (NSV);
- NATO Technical View (NTV); and
- NATO Programme View (NPV).

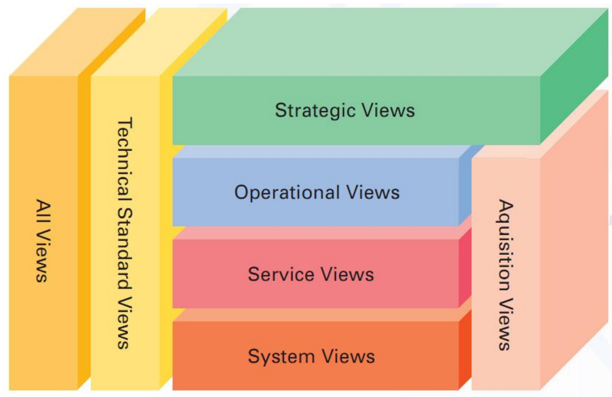

Figure 5 – MODAF Views

4.1. NATO Architecture Framework (NAF)

In effect, the NAF is comprised of structural and behavioral views that include conceptual, logical and physical model views. The conceptual model (CM) is akin to an ontology and represents an operational view of what information is exchanged. The logical model (LM) is a Platform Independent Model (PIM) or *domain model*; it is a complete, normalized description of the information identified in the conceptual model. The LM may be broken down into additional horizontal/vertical layers (e.g. message models, capability packages). The physical model (PM) is a Platform-Specific Model (PSM) sometimes referred to as a message model. The LM is implementation independent while different PMs may be created to support various payload types (e.g. SOAP/XML RESTful/JSON, DDS/IDL, etc.).

The NAF, and other similar AF such as DODAF and MODAF, link operational needs to technical requirements and provide the necessary traceability to the solution through the definition of a formal model. This approach is quite flexible and therefore facilitates evolution of the requirements, model and or derived products.

4.2. Operations Based Requirements Management

The SE methodology for standards development must include a RM activity that is grounded in operational requirements. These requirements in turn must be traceable to derived requirements that finally are traceable

[3] http://www.iso-architecture.org/ieee-1471/afs/frameworks-table.html

to the specific elements of the standard to which they relate.

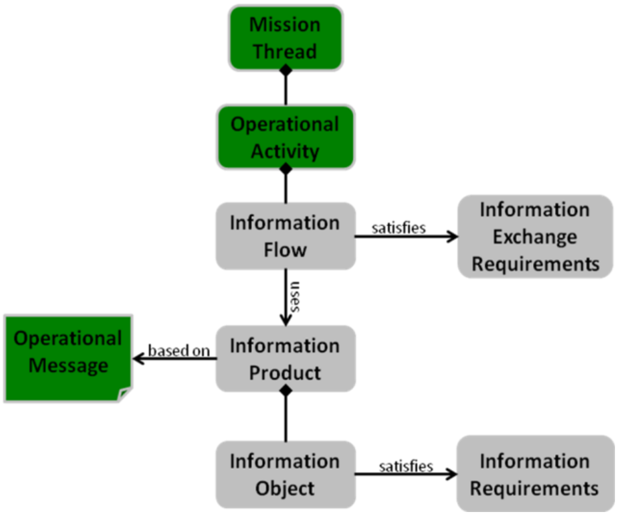

Figure 6-C-BML Requirements Map

Consistent with the NAF, Figure 6 illustrates the underlying requirements elicitation mechanism. Requirements are derived from *information flows* that enable specific *operational activities*. The information flows involve the exchange of *information products* that are comprised of *information objects*. To maintain operational relevance, information products are based on actual operational messages as per existing procedures. In many instances not all of the information elements in a given operational message are required by the information product since the latter is intended to communicate a subset of the information in the operational message. For example, in the case of C-BML, simulations generally cannot parse free-text elements of operational messages intended for human consumption and therefore all free-text elements should not be included, by default. Nonetheless, responses from simulated forces may include free-text fields indicating, for example, the reason for a negative acknowledgement of a specific task execution. In general, two types of requirements are identified: *information requirements* (IR) and *information exchange requirements* (IER). In general, IER may be operational requirements, system-specific requirements or technical requirements. For the purposes of this approach, IER are those requirements that are associated and/or derived from the operational information flow. IR refers to the set of lower-level requirements related to specific information elements or data elements.

4.3. Maintainability of the Standard

The previous sections discuss considerations for establishing a well-defined set of validated, traceable requirements based on continued stakeholder involvement as the foundation for the proposed approach. Moreover, these requirements evolve over time and must be taken into account in a timely and flexible manner. This is consistent with the C-BML Standard Development Framework, illustrated in Figure 7 [9] that defines a *reference architecture*, comprised of: a content model; message structures; interaction protocols; and service components.

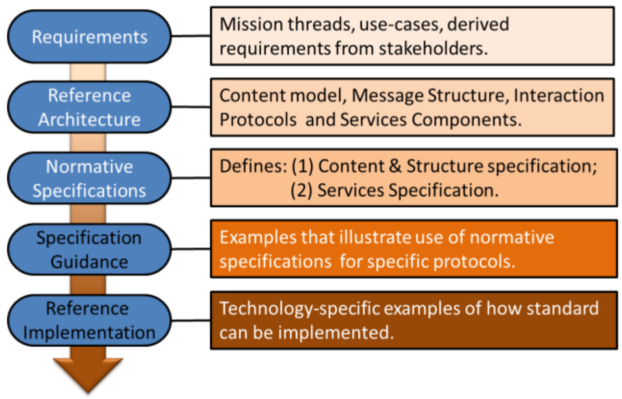

Figure 7 - C-BML Standard Development Framework [9]

Although operational messages were used as one of the sources for generating requirements, the messages defined as part of the reference architecture may have different characteristics. In the case of C-BML, for example, the message structures refer to formal expressions formulated in accordance with production rules defined by a grammar. Here it is important to define the means by which the normative and informative specifications will be specified. For example, Figure 8 proposes a basic classification of elements or things to be used as the vocabulary for C2-SIM interoperability standards. Influenced by the JC3IEDM [28], this set consists of entities and events (upper half) and attributes and properties (lower half) that are associated to these entities and events.

The separation of concerns is an important aspect of developing interoperability solutions, as described by Lang et al [6], as well as for organizing the model and standard in a modular form. Standards serve different purposes for different users from various communities. In the case of C-BML, the air/land/maritime domains each have specificities that translate into different requirements for C-BML, as described in section 5.

The modularity of any solution is one of the keys to ensuring its maintainability. Concerning standards development, another important aspect is the ability to rapidly generate new revisions of the standard based on revised requirements.

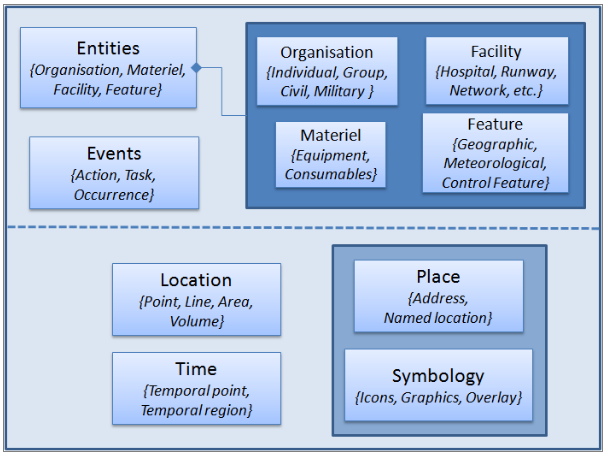

Figure 8 - C2-SIM Domain Modeled Concepts

The current paper also advocates the use of UML as a means to formalize requirements and reference architecture. UML tools, such as *Sparx Systems Enterprise Architect* now include UML profiles and add-ins for requirements management, model transformations, eXtended Markup Language (XML) schema generation, code generation and other actions. UML tools also can support a distributed collaborative development environment based on readily available version control systems, such as *Subversion*. One of the main benefits of employing an UML-based standard development approach is the use of built-in document generation capabilities. Being able to generate standards product artifacts in an automated fashion can contribute greatly to both the maintainability and the usability of the standard, as described in the next section.

4.4. Automated Artifact Generation

Once a process has been defined for producing a set of products or *artifacts* comprising a technical standard, it then is possible to consider automating the generation of these artifacts.

4.4.1. Process Documentation

The process must be well documented and well understood to be utilized successfully by stakeholders. The process can be captured as part of the UML model itself. Using automated documentation generation capabilities, the process description can be exported as a set of Hypertext Markup Language (HTML) pages.

4.4.2. Requirements Specification Generation

Requirements specifications can be generated automatically at regular intervals in order to facilitate organization of requirements and internal and external validation.

4.4.3. Model Description

XML has become one of the *de facto* choices for representing all kinds of structured data. XML Schema Description (XSD) documents often are used to define interoperability standards such as SISO Military Scenario Definition Language (MSDL), C-BML and also the National Information Exchange Model (see https://www.niem.gov). However, at the heart of an interoperability standard, there is a model and it is not always easy to conceptualize or understand the relationships of the various model elements by inspecting of the XSD. Although XSD are model representations, they are not necessarily *normalized* models[4]. For complex standards, maintaining schemas manually is labor intensive and can be error-prone [9].

Therefore there are benefits associated with a structured approach of developing *normalized* PIM using languages such as UML. Consistent with the Model-Driven Architecture (MDA) approach, XML schemata and other desired outputs (e.g. PSM) can be generated from the UML PIM. XSD is one of several possible model transformation outputs but other PSM can then be generated, such as the JavaScript Object Notation (JSON), often used in conjunction with RESTful style web services. Also, UML profiles now are available for architectural frameworks such as the NAF, for various platform-specific language and other technologies (e.g. C++, C#, JAVA OWL, DDS, WS etc…) as well as for SE with the *Systems Modeling Language Profile for UML* (SysML). Finally, UML vendor tools generally offer automated documentation generation features as well including exporting model descriptions to RTF and html formats.

5. Application to C-BML Standard Development

The following sections describe the results of recent efforts that have contributed to the C-BML standard development activity consistent with the approach outlined in this paper.

As described in Figure 6, common to these activities is the emphasis on defining the relevant information flows based on specific operational activities and consistent with currently utilized operational messages.

5.1. Call-For-Fire Operational View and Use-Case

Kicked-off in May 2012, the C-BML Industry Task Team (CITT) has been formed by SISO and the Net-Centric

[4] http://gerardnico.com/wiki/data_modeling/normalization

Operations Industry Consortium NCOIC) as an extension of the SISO C-BML Phase 2 Drafting Group (DG). The aim of this group is to bolster the resources of the C-BML Phase 2 DG while increasing industry awareness and involvement in the C-BML Phase 2 drafting activity. This group already has produced several draft documents including an OV1a diagram to illustrate the Call-For-Fire (CFF) process and a use-case description for an Artillery CFF interaction. The OV1a (NOV in NAF) is shown in Figure 9.

Figure 9 – OV1a Call-For Fire

The Call-For-Fire (CFF) use-case (see Table 1) documents a CFF trainer used for training Forward Observers (FOs) to properly call for fire. As per Figure 6, the use-case documents a CFF "mission thread" as a PIM/domain model with operational messages expressed in natural language with the intent that C-BML would subsequently replace the natural language to support the information flows and satisfy the IERs.

5.1.1. CFF Operational Activity

The operational activity described by the CFF mission thread is the call-for-fire. A CFF is a message sent by the FO to the Fire Direction Center (FDC) requesting that a target be attacked with indirect fire, artillery or mortar fire. The FO composes and sends the CFF in segments, both for speed and clarity, and each segment is recorded and read back by the FDC recorder so that the FO can verify the accuracy of the transmitted information. The CFF is a request for fire, not an order, as the FO lacks situational awareness of competing requirements on the FDC, ammunition availability, and so on. The necessity for speed and clarity, and the respective situations facing, and alternatives available to the FO and FDC, dictate formal communication, the CFF, and thorough training [22].

Table 1. CFF Use Case Step 1

Heading	Cell Entry
Event	FO identifies Target in Battlefield
Actor Activity and Tools Used	FO, Target, Battlefield, map and compass, Radio, Speech Recognition System (SRS), Translator, FDC FO transmits CFF Warning Order (WO) Radio, SRS, Translator, FDC
Description of Processing	FO identifies Target in Battlefield and decides that he will need to adjust fire (i.e. he is not sufficiently confident of the target location to justify a FFE warning order). He transmits the CFF WO. SRS converts speech to text. Translator validates WO text to ensure proper formating. If valid, it translates the validated text to a C-BML message, and sends the C-BML message to the FDC.
Information Producer	FO
Information Receiver	FDC
Information to be Exchanged	A57 this is A71, Adjust Fire, over
Notes	

5.1.2. CFF Use Case Construction

The CFF use-case lists actors, assumptions, pre-conditions, a normal sequence, exceptions or alternate sequences, post conditions, activity diagrams, references and outstanding issues.

While all sections of the use case are necessary for internal completeness, it is beyond the scope of this paper to present them in their entirety. Instead, the normal sequence is described as it illustrates the potential of the use case as a means of deriving information products from operational messages and, more generally, the value of the systems engineering approach described in section 4. The normal sequence documents the primary use case as an incident-free CFF consisting of a set of discrete steps in the CFF process. Events triggering each step are identified as well as actors, activities, a description of processing associated with the event and, most importantly for our purposes, information exchanged during the step between a specified producer and receiver. Table 1 shows the entries for step one of the CFF normal sequence as an example of the information contained in the use case. The entry corresponding to "Information to be Exchanged," in this case, "*A57 this is A71, Adjust Fire, over*" is the operational message requiring conversion to C-BML.

5.2. MSG-085: Standardization for C2-Simulation Interoperation

The NATO MSG-085 Technical Activity (TA) has been mandated by the NATO Collaboration Support Office (CSO) as follow-on activity to the MSG-048 (C-BML) TA [22]. With participation from 13 nations, MSG-085 has been working in the area of C2-SIM interoperation since 2010 and currently is slated to run through 2013. MSG-085 is working in the areas of military scenario definition, initialization, and execution using C-BML and also the SISO MSDL. The main objectives of the MSG-085 TA are as follows:

- Clarify and complement existing C-BML and MSDL requirements;
- Propose a set of C-BML orders and reports to serve as a common reference set;
- Assess and leverage available C-BML implementations;
- Address C2 and simulation initialization requirements; and
- Demonstrate the operational relevance and benefits of the approaches considered.

MSG-085 is tasked with assessing the *operational relevance* of C-BML and to assist in increasing the *Technical Readiness* Level of C-BML technology to a level consistent with operational employment by stakeholders. To accomplish this mission, MSG-085 has formed two sub-groups: the Operational Sub-Group (OSG) and the Technical Sub-Group (TSG), that focus on operational and technical requirements for C2-SIM interoperability. Moreover representation is present from each service (Air, Land, and Maritime) to ensure the operational relevance of C-BML for multi-national and multi-service use.

Recent research and experimentation conducted by MSG-085 has been conducted by Common Interest Groups (CIG) formed to focus on specific areas of interest. CIGs were established for each of the Air, Land and Maritime domains. CIGs also were formed for Joint Mission Planning and C2-SIM Infrastructure. The OSG, TSG and the CIGs have contributed to establishing requirements for C2-SIM interoperability through the use of the approach described in this paper. The OSG has led the elaboration of a set of Operational Concept Description (OCD) documents [32][33], one for training and the other for mission planning (course of action analysis). The TSG has contributed to an UML-based collaborative workspace for organizing and tracing requirements for subsequent MSDL/C-BML language development. Finally the Air Ops, Land Ops and Maritime Ops CIGs have established domain-specific requirements for extensions to existing C2-SIM interoperability standards.

Some of the results of these activities recently were presented at the NATO MSG-119 C2-SIM Interoperability Workshop that took place in December 2012 [16] and included live and recorded demonstrations with C2 and simulation systems interoperating in the context of multinational command post training, command post planning and coalition joint fires support experimentation. In addition, detailed analyses were performed by operational subject matter experts to identify requirements and propose domain-specific C-BML extensions. The following subsections highlight the results of this work.

5.2.1. MSG-085 Operational Sub-Group CONOPS

To ground the operational requirements of C-BML on a strong basis, MSG-085 developed two OCDs.Prior to development of OCDs, a set of operational use-cases and usage scenarios where C-BML provides better military efficiency were produced. Operational use-cases formed the basis to develop operational requirements in the form of OCDs. The OCDs are high level overviews of the operational characteristics and capabilities of systems enabled by C-BML and derived from the operational use-cases. For different application domains the impact of C-BML on Doctrine, Organization, Training, Materiel, Leadership. Personnel and Facilities (DOTMLPF) are assessed in the OCDs.

Additional to defining operational requirements for C-BML, OCDs also are aiming to:

- Provide a high level overview of the operational characteristics and capabilities of a system of systems equipped with a C-BML interface.
- Define the environment in which the system will operate,
- Provide the criteria to be used for validation of the C-BML systems.

The OCDs serve as input to technical requirements analysis and form the basis for developing technical specifications. Thus, a set of technical requirements derived from OCDs will specify the needs in terms of lexicon, grammar, data formats, and services for information exchange, architecture, performance and effectiveness. The analysis is dealing with the currently available SISO standards and their capability to fulfill the technical requirements.

5.3. MSG-085 Maritime Operations CIG

An overarching goal of the MSG-085 Programme of Work (POW) objectives is to ensure the operational relevance of C-BML: (1) investigation of multi-level and multi-service use of C-BML. (2) identification of operational and technical requirements to ensure that C-BML supports multi-national and multi-service use. Based on these objectives and goals, MSG-085 Maritime CIG focuses on investigating the use of C-BML in the Maritime domain and aims to develop and test a preliminary Maritime extension to C-BML that will be used for expressing and exchanging plans, orders and reports specific to the Maritime domain. One of primary goals of the Maritime CIG is to establish a set of requirements for a Maritime extension to the C-BML standard. A requirements driven approach has several advantages, including ensuring that the resulting C-BML product is grounded in military operational procedures and is consistent with the associated operational message flows. The following sections provide a brief overview of this work. For more details concerning the MSG-085 Maritime Operations CIG activity, see reference [24].

5.3.1. Maritime Operations CIG Objectives and Research Methodology

The methodology applied by the Maritime CIG can be summarized as follows:

1. Preparation of example operational messages in accordance with a Naval operational scenario;
2. Development of an initial set of prioritized IERs based on message templates in APP-11(C);
3. Mapping of the prioritized IERs to C-BML expression/ elements using a scenario-based modeling approach;
4. Development of a maritime tasking grammar using Command and Control Lexical Grammar (C2LG) [24].

5.3.2. Maritime Ops CIG Operational Scenario

The Naval operational scenario is derived from the scenario of the VIKING 2011 exercise, which uses the real-life geography of central and southern Sweden, renamed to a fictional country called BOGALAND and neighboring countries [26]. In the scenario, Coalition Task Force (CTF) 401, which includes three Task Groups consisting of frigates with organic helicopters, mine hunters, mine sweepers and LPDs, is responsible for conducting Maritime Interdiction Operations (MIO) ashore to control maritime traffic. CTF 401 also is responsible for supporting amphibious forces in a pre-landing phase and protecting Mine Counter-Measure (MCM) forces against air and surface threats. Existing research covers only the exchange of selected parts of the two naval operational messages, Operational General Matters (OPGEN) and Operational Tasking of Antisurface Warfare (OPTASK ASUW). OPGEN is used to promulgate general matters of policy, instructions and expectations of Officer-in-Tactical Command (OTC) to all types of warfare. OPTASK ASUW is used to promulgate detailed plans, tasking and instructions to conduct Anti-Surface Warfare [27]. Both of these messages are issued by the OTC to the subordinate commanders.

5.3.3. Conduct of Research

The following paragraphs outline the tasks being executed by Maritime CIG to conduct preliminary research on the Maritime C-BML.

Identify and document an initial set of C-BML IERs for the Maritime domain. IERs allow for the specification of information objects (i.e. C-BML expressions) that comprise information products (i.e. C-BML messages). As part of the maritime C-BML requirements activity, both information requirements and information exchange requirements were identified and documented. As an example, Figure 10 shows a partial view of the requirements for expressing a Naval Task Organization in the form of the SysML requirements profile for the UML.

Conduct basic C-BML/MSDL modeling based on the IERs. The concepts used for C-BML and MSDL models are depicted in **Error! Reference source not found.**. These concepts are consistent with the JC3IEDM [28], developed by the Multilateral Interoperability Programme (MIP) and the NATO Joint Symbology Standards, such as NATO APP-6 (C) [29].

Construct C2 Lexical Grammar for the Maritime Domain. The insights and results from the scenario-based modeling can be collected, and generalized in a formal way in order to formulate a grammar. Such a grammar helps to formalize and standardize C-BML in general and the maritime extension of C-BML in particular.

Identify C2-Sim initialization requirements for Maritime domain. Military enterprise activities such as training and experimentation require the definition of operational scenarios that are executed across a set of interconnected C2 and simulation systems that can be considered as a C2-SIM federation. As with simulation federations such as the High Level Architecture (HLA), when developing a C2-SIM federation it is necessary to consider all aspects of the scenario life-cycle: definition, refinement, initialization, execution, post-scenario analysis, consistent with the Distributed Systems Engineering and Execution Process (DSEEP) [30] The current work considers C2-SIM initialization requirements that are intended to guide the development of future versions of standards, such as MSDL, for scenario definition, refinement and initialization. This initial set of requirements forms a core that then is extended with additional requirements that support scenario execution, i.e. for defining the C-BML Maritime Domain extension.

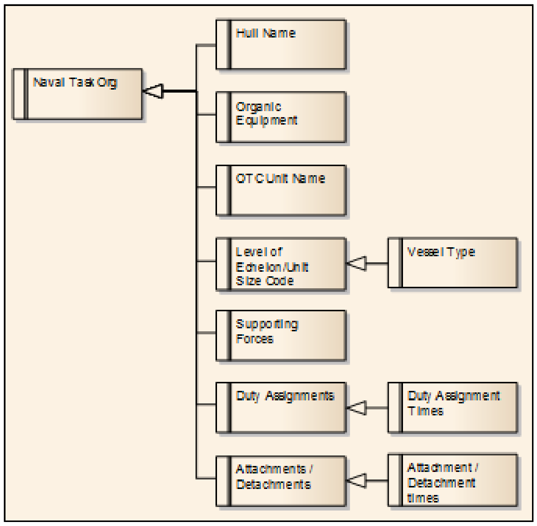

Figure 10-Example Requirements for Naval Task Organization

Experiment/demonstrate the initial Maritime C-BML capability in a relevant environment. An experimentation event for the practical observation of the deliverables of the research effort is being planned within the MSG-085 Final Experimentation. The experimentation environment will consist of Naval C2 systems (or surrogates), C-BML servers and Naval simulations.

5.3.4. Maritime Operations CIG Initial Findings

Identification of correct operational requirements is essential for the successful operational deployment of C-BML. For that reason, Maritime CIG choose a requirements-driven approach to conduct this research. An operational scenario and relevant operational message samples based on that scenario are useful for the validation of the operational requirements. Also, operational requirements are to be refined in collaboration with technical personnel in order to identify technical requirements for C-BML modeling.

It should be noted that, the naval warfare domain is a very large area that includes inherent complexity due to the behavior of the operational area, diversity of the threats and requirements to conduct different types of warfare simultaneously within a task force. To cover all types of Naval Warfare within a Maritime C-BML study, substantial resources are required. To mitigate risks and optimize resource allocation, an incremental and iterative development methodology is advised [31].

5.4. MSG-085 Air Operations CIG

5.4.1. Air Ops CIG Objectives and Methodology

The MSG-085 Air Ops CIG has developed C2-SIM interoperability capabilities to support the air component scenario initialization and scenario execution in the context of multinational training and Joint Fires Support experimentation. The capability that was developed was based on a detailed analysis of operational information flows including the Air Tasking Order (ATO) and the Airspace Control Order (ACO) operational messages. One of the underlying assumptions for this work was that the operational Command and Control Information System (C2IS) could not be modified and therefore operational messages were taken as inputs in US Message Text Format (USMTF) and NATO APP-11(C) [27] formats. Thus the ATO and ACO were translated into C2-SIM interoperability formats (i.e. C-BML) using translators that subsequently could share air tasking and airspace control information with the rest of the C2-SIM federation. An example of high-level requirements for air domain messages is shown in Figure 11.

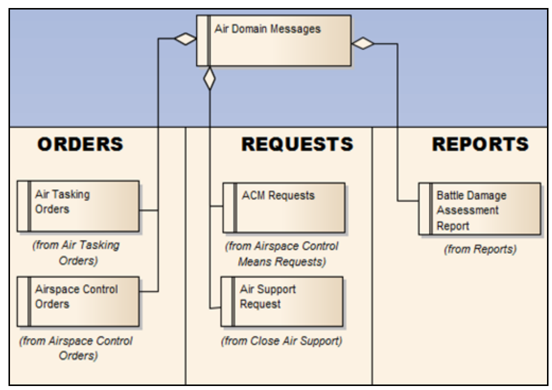

Figure 11 – Example of High-Level C2-SIM Air Operations Requirements

5.4.2. Air Ops CIG Results and Findings

Air operations can include *preplanned*, *on-call* and *immediate* missions. The execution of these missions in an operational training or experimentation environment will require a C2-Simulation federation wherein simulated air assets can be controlled by a combination of live, virtual and constructive actors. Preplanned missions such as Air Interdiction (AI) generally can be simulated in a highly automated manner whereas the execution of support missions such as Close Air Support (CAS) requires coordination between the aircraft and the supported unit (i.e. the Forward Air Controller (FAC)), who may be human or simulated. On-call and immediate missions cannot completely be planned for in advance and often require *dynamic tasking*; new or modified tasks or missions that arise during an existing mission to meet evolving battlespace requirements. For example, an aircraft performing an Intelligence, Surveillance, and Reconnaissance (ISR) task may report the position of a High-Value Target (HVT) and subsequently may be re-tasked to engage the target.

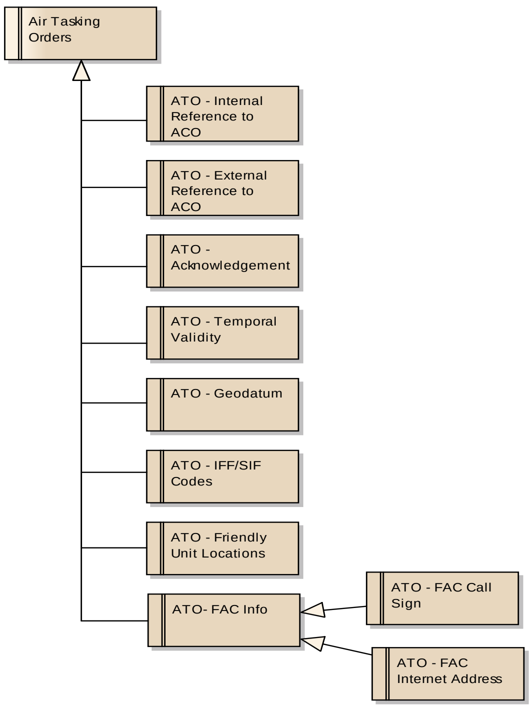

Figure 12 – Snapshot of subset of C2-SIM Air Ops Requirements (Air Tasking Order)

A first benefit of constructing a C-BML representation of the ATO and ACO was that it alowed for a direct use of the initial existing operational ATO (see Figure 12) and ACO to initialize the simulated blue forces air component. Feeding the ATO and ACO as defined by the air C2IS directly to the C2-SIM federation can provide for rapidly specifying the initial aircraft types, locations, missions and relating them to pre-defined Airspace Control Means (ACM) as specified in the ACO without the need for a simulation operator or other human-in-the-loop. This represents a significant cost-reduction compared to the manual process currently employed and also allows for accelerating significantly the scenario development and refinement process. The use of technologies such as C-BML was shown to support both preplanned and dynamic mission tasking. In particular, C-BML has been used effectively for the execution of preplanned missions by simulated assets while maintaining the flexibility to include human actors, as required to satisfy training and/or experimentation goals, depending on the target audience. This reinforces the point that C2-SIM interoperability technologies support the Live Virtual Constructive (LVC) training paradigm.

The use of actual operational messages as issued by real C2IS also greatly facilitated the integration of air C2IS as part of the C2-SIM federation. However, current air operations messages include, in some instances, free-text fields for important coordination purposes. Similar to findings from other CIGs, the need to transport free-text must be addressed by C2-SIM interoperability standardization efforts.

5.5. MSG-085 Land Operations CIG

The Land Ops CIG activity included multi-national experimentation for command post training and command post training operational activities using C2-SIM interoperability technologies. The focus of the current paper is on the approach used to develop and share requirements with other C2-SIM interoperability standards stakeholders as part of a collaborative standards development process. For more detailed information on the MSG-085 Land Ops CIG experimentation capability, see reference [35].

5.5.1. Land Ops CIG Objectives and Methodology

The objectives for the Land Ops CIG are centered on land-focused training and were identified thanks to the lessons learned from past experimentation [36].

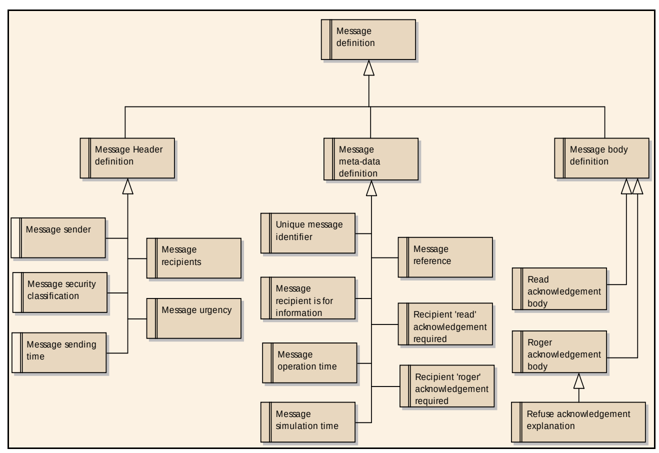

Figure 13- Land Ops C2-SIM General Message Requirements

In particular, the 2012 efforts focused on the following objectives:

- Enhancing land maneuver logistics (sustainment of fuel and personnel);
- Exchanging Request/Order/Report Messages for Artillery Support;
- Extending the list of tasks that C-BML is able to support with low intensity missions;
- Exchanging with legacy C2 systems using operational interfaces that comply with Command Post (CP) flow of information; and
- Refining the systems initialization process.

As part of the Land Ops CIG capability development process operational requirements were established for CP training and planning activities directly based on the operational message information flows and then detailed technical requirements for the C2-SIM standards were derived, (see Figure 13).

These requirements then were used to construct the XSD C-BML Message Schema, shown Figure 14, and included as part of the an Interface Specification Document (ICD) [37].

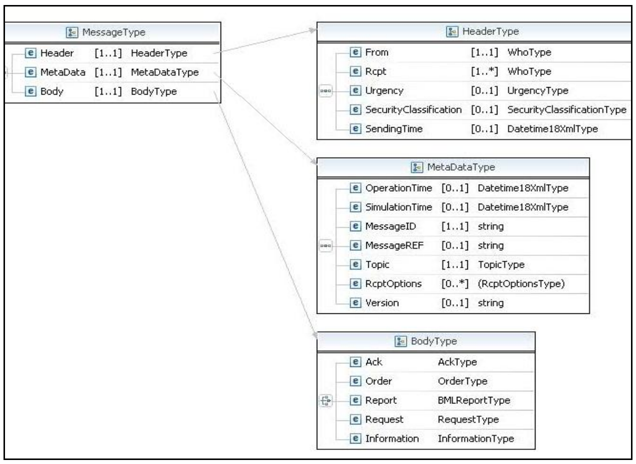

Figure 14 – Land Ops C2-SIM General Message Schema [37]

5.5.2. Land CIG Operational Scenario

The operational scenario that was developed for the purposes of recent experimentation was developed by Spain, The Netherlands and France based on the Viking 2011 scenario [26].

The scenario involves TF V (Task Force V) containing 3 companies (STF A, STF B, STF C), a RECCE platoon (PLT E), a mortar platoon (PLT F) and an artillery battery. Figure 15 shows the order of battle, with the C2 systems (shown in green) used for the different CPs.

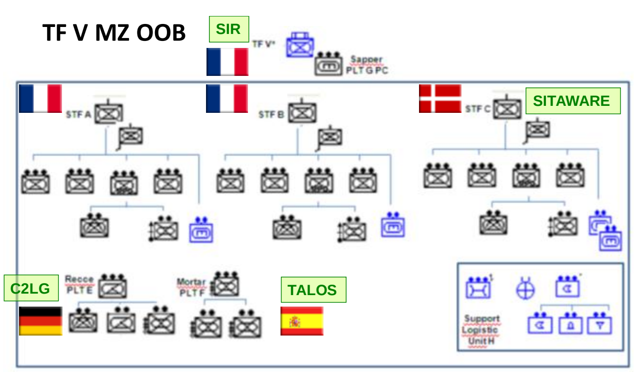

Figure 15 - TF V Order of Battle

After disembarkation at Oxelösund and movement to the waiting area (WA), the scenario guidelines for TF V were to seize and secure Skavsta airport (APOD). The operational order made by the TF V Battalion command post has several phases. Each one has been used to show achievements of different objectives.

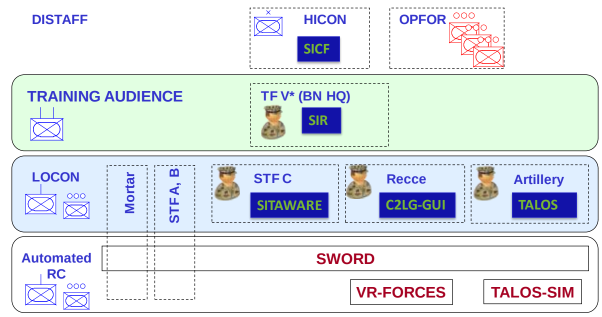

Figure 16 - Command Post Training Organization

Figure 16 shows the training organization including the units and their roles, with the systems and people involved. The training audience is the Battalion command post, using the SIR system, the regular French C2IS. The low controller (LOCON), (*aka* Secondary Training Audience (STA) for France) provides the interface between the trainees and the synthetic environment. All subordinate units of TF V (except STF A and B) are represented in this cell: the STF C company in SITAWARE (regular Danish C2 system), the RECCE platoon with C2LG-GUI (German surrogate C2IS), and the Artillery unit with TALOS (regular Spanish C2 system). For these units, only their subordinates are represented in the simulations.

There is no officer in the LOCON for STF A and B companies as well as for the Mortar units. They are directly represented in the SWORD simulation and hence are part of the automated RC (Response Cell).

Several simulations are used in the response cell:

- SWORD (MASA Group): provides the whole ground operational environment with a command agent allowing to order company and section/platoon
- VR-Forces (The Netherlands): used to display a disaggregated view (RECCE vehicles) of the battlefield around the airport
- TALOS-SIM (Spain): provides the means to compute artillery fires and send HLA detonations interactions.

5.5.3. Land Ops CIG Results and Findings

Based on the configuration and scenario described above, a Battalion CP training event was conducted and highlighted several advantages to employing C2-SIM interoperability technologies, including virtually eliminating the need for swivel-chair simulation operators in the Response Cells

(RC) of the Lower Controller (LOCON) and the Higher Controller (HICON). The event illustrated the execution of a three-phase operation to take control of an airport under enemy control. It was based on a realistic and seamless exchange of military messages between superior and subordinate commanders, some of which were simulated commanders. These exchanges included: orders (with order acknowledgement); artillery call-for-fire requests and subsequent messages; Unmanned Air Vehicle (UAV) tasking; intelligence reports; and logistic and situation reports.

The simulations executed the orders in a realistic and timely fashion, but when an intentional error was introduced into an order created by a C2IS operator commanding a simulated force, the simulated commander replied to his (human) superior with an acknowledgement reply message indicating that he was unable to execute the order. The simulation behavior included the reason for the (negative) acknowledgement reply as part of the acknowledgement reply message, consistent with the requirements shown in Figure 13. Within the machine message, the "reason" is represented as a free-text field and is required so that it can be displayed to the human C2IS operator, as per the training requirements and consistent with current operational systems. Once the (human) error was corrected, the simulation then was able to successfully execute the order and the operation was able to proceed. This also is consistent with the initial projected use of C-BML by both machines (e.g. C2/simulation systems) and by human operators (e.g. Battalion C2IS operator) [38].

This demonstration also highlighted the additional C2-SIM interoperability benefit of increased realism, as demonstrated through the incorporation of logistics information in the scenario. Initial equipment and supply quantities first were specified as part of the scenario initialization process using operationally representative means, such as the NATO Stock Number (NSN) reference. Then, during scenario execution, information such as quantities was updated through the use of appropriate operational messages, such as: Logistic Report (LOGREP) and Situation Report on human resources (SITEFF). The demonstration gave an overview of a Battalion CP training capability that encompassed significant levels of detail covering several domains (e.g. Intelligence, Logistics, and Fires Support) and was based on requirements derived directly from real operational messages.

One of the conclusions of this work was that in order to maximize the usability and achieve greater benefits of the C2-SIM interoperability technologies, C2-SIM interoperability standards products such as XML schemata should be derived from operational requirements. Another conclusion of this work was that although these XML schemata are necessary to perform the development and integration tasks, it is important that they be part of a larger reproducible process that includes traceability back to the operational and technical requirements and therefore can support the need for evolution over time. It also was concluded that the C2-SIM interoperability process include a federation agreement.

6. Conclusions & Future Work

The development of international technical interoperability standards for multiple domains and communities from the C2 and simulation worlds is a labor-intensive and complex endeavor. Past experience has shown that producing such standards can take many years unless a dedicated process is established that ensures proper stakeholder involvement.

Recent standards development organizations have reported that applying Systems Engineering methodologies coupled with an Enterprise Architecture approach can provide a framework and assist in reuniting the necessary and sufficient conditions for success. One of the keys to ensuring that a successful standard is developed is to establish a requirements management process wherein requirements are grounded in stakeholder operational needs, properly organized, and traced to standards artifacts. Furthermore, as stakeholders needs evolve, it must be straightforward to update the requirements and rapidly generate a new set of standards artifacts. The results of recent independent work performed by the NATO MSG-085 Technical Group and by the C-BML Industry Task Team have demonstrated that there are clear benefits to be gained in applying a requirements-centric systems engineering approach toward establishing operational and technical requirements to guide the development of the SISO C-BML standard.

Moreover, the development of complex technical standards such as C-BML clearly requires a standards development process and this process must be well-documented, well-understood, it must be iterative and involve regular involvement of stakeholders.

Requirements may originate from analysis of operational information flows or from lessons learned from experimentation. However, in both cases, the results of these activities must be expressed in terms of requirements that then can be reinjected into the standards development process. Extraneous requirements and unjustifiable positions based on academic perspectives must be avoided in order to maintain a manageable and verifiable scope. Requirements for future potentially disruptive capabilities must be handled separately from requirements for short-term support of existing military activities using currently employed operational systems.

The use of a distributed, collaborative UML-based environment is an efficient means of implementing technical interoperability standards development processes such as the one outlined in this paper. Such an environment has been prototyped and utilized by the MSG-085 Technical Group in establishing C2-Simulation interoperability requirements for military scenario initialization and scenario execution.

The automatic generation of the requirements specifications and the normative and informative standards artifacts can help to reduce the time between iterations. The production of documents such as XML Schemata should be done in automated manner so as to avoid possible human error and inherent difficulties in maintaining large complex sets of related files. Although XSD provide an implicit model representation in the form of a PSM, this does not replace the need for platform-independent conceptual and logical models.

Future work includes the continued development of a UML-based C2-SIM interoperability standards development environment that will leverage the MIP Information Model and toolset for the elaboration of future versions of the MSDL and C-BML standards.

7. References

[1] C. Hallam: "An Overview of Systems Engineering: The Art of Managing Complexity", MIT Engineering Systems Division-Research Seminar in Engineering Systems, October 2001.

[2] J. Abbott, S. Levine, M. Pullen: "Answering The Question Why A BML Standard Has Taken So Long To Be Establishes?", Fall 2011 SIW, Orlando USA.

[3] P. Gonzalez, B. Christie, J. White: "Applying Systems Engineering Principles to the Development of Transportation Communication Standards", RITA-Intelligent Transportation Systems Joint Program Office, Final Report, FHWA-JPO-089, April 2011.

[4] A. Bridgewater: "Interview with Jim Highsmith: From Apollo Mission to agility manifesto", Computerly Weekly, December 2011.

[5] A. Mostashari, J. Sussman: "Engaging Stakeholders in Engineering Systems Representation and Modeling", MIT Engineering Systems Symposium, March 2004.

[6] B. Lang, M. Gerz, O. Meyer, D. Sim: "An Enterprise Architecture for the Delivery of a Modular Interoperability Solution", NATO RTO-MP-IST-101, Information Systems Technology Panel Symposium, Oslo 2011.

[7] NATO Glossary of Terms and Definitions, AAP-6 (2010).

[8] M. Kennedy, D. Umphress: "An Agile Systems Engineering Process", People Solutions to Software Problems, CrossTalk, May-June 2011.

[9] K. Gupton, K. Heffner: "A Standards Development Framework for C-BML Phase 2 and Beyond", Fall 2012 SIW, Orlando USA.

[10] US Intelligence Community and Department of Defense-Content Discovery & Retrieval Integrated Project Team: "IC/DoD Keyword Query Language Specification", V2.0-20121003, October 2012.

[11] NATO Architecture Framework, Version 3, 2007.

[12] INCOSE: "Brief History of Systems Engineering", International council on Systems Engineering (INCOSE) Web Site, Last accessed 10-January-2013.

[13] J. Clark: "Systems of Systems Engineering from a Standards, V-Model, and Dual V-Model Perspective", Systems and Software Technology Conference, April 2009.

[14] B. Satalkar: "Waterfall Model Vs. V Model", http://www.buzzle.com/articles/waterfall-model-vs-v-model.html, Last accessed 10-Jan-2013.

[15] ISO/IED Software & Systems Engineering Secretariat: CANADA (SCC): PDTR 19760 Systems Engineering – Guide for ISO/IEC 15288 (System Life Cycle Process), July 2002.

[16] K. Heffner: "NATO MSG-119 C2-Simulation Interoperability Workshop Technical Evaluation Report", To be published.

[17] F. Paetsch, A. Eberlein, F. Maurer: "Requirements Engineering and Agile Software Development", Proceedings of the 12th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises, IEEE 2003.

[18] ISO/IEC 9126-1:2001, "Software engineering – Product quality – Part 1: Quality Model", 2001.

[19] ISO/IEC 25010 Draft: "Systems and software engineering – Systems and software Quality Requirements and Evaluation (SquaRE) –System and software quality models", 2010.

[20] E. Trejos: "Introduction to DO-178B" July 2010

[21] G. Zoughbi, L. Brian, Y. Labiche: "A UML Profile For Developing Airworthiness-Compliant (RTCA DO-178B) Safety-Critical Software", Carleton University, TR SCE-06-19.

[22] Field Manual (FM) 6-30, Tactics, Techniques, and Procedures for Observed Fire, Headquarters, Department of the Army, Washington, DC, 16 July 1991
[23] NATO MSG-048 (C-BML) Final Report, February 2012.
[24] H. Savasan, A. Caglayan, F. Yildiz, OM Mevassvik, G. Sletten, U. Schade, B. Haarmann, K. Heffner: “Towards a Maritime Domain Extension to Coalition Battle Management Language: Initial Findings and Way Forward”, Spring 2013 Simulation Interoperability Workshop, 13S-SIW-022, San Diego, April 2013.
[25] U. Schade and M.R. Hieb: “Formalizing Battle Management Language: A grammar for specifying orders,” 2006 Spring Simulation Interoperability Workshop, Paper 06S-SIW-068, Huntsville, AL, 2006.
[26] VIKING 11 Exercise Specification, Version 2.0, 1 March 2010.
[27] NATO: APP-11(C), NATO Message Catalogue, NATO Unclassified, 2009.
[28] Multilateral Interoperability Programme: Overview of the Joint C3 Information Exchange Data Model (JC3IEDM Overview), Version 3.1.4, Greding, Germany, 14 February 2012.
[29] NATO: APP-6(C), NATO Joint Military Symbology, NATO Unclassified, 2011.
[30] IEEE Recommended Practice for Distributed Simulation Engineering and Execution Process (DSEEP), IEEE-Std-1730-2010.
[31] INCOSE Systems Engineering Handbook Version 3.2.2, October 2011.
[32] IEEE Std 1362-1998 IEEE Guide for Information Technology - System Definition - Concept of Operations (ConOps) Document.
[33] MIL-STD-498 Software Development and Documentation, Operational Concept Description (OCD).
[34] A. Brook, K. Heffner: “Air Operations CIG”, Spring 2013 Simulation Interoperability Workshop, 13S-SIW-009, San Diego, April 2013.
[35] B. Gautreau, L. Khimeche, T. Remmersmann, J. Martinet, D. Muniz, T. Serrano, E. Pedersen, J. Lillesoe, N. de Reus, H. Hendersen, D. Liberg: “Lessons learned from NMSG-85 CIG Land Operation demonstration”, Spring 2013 Simulation Interoperability Workshop, 13S-SIW-031, San Diego, April 2013.
[36] T. Remmersmann, U. Schade, L. Khimeche, B. Gautreau, M. Pullen, R. El Abdouni Khayari: “Lessons Recognized: how to combine BML and MSDL”, Spring 2012 Simulation Interoperability Workshop, 12S-SIW-012, Orlando, FL, 2012.
[37] NATO MSG-085 Land Operations CIG Interface Specification Document, December 2012.
[38] H. Argo, E. Brennan, M. Collins, K. Gipson, C. Lindstrom, S. MacKinnon: “Level 1 Model for Battle Management Language (BML-1)”, 1999.

Author Biographies

ANDY BOWERS is a Lead Simulation Modeling Engineer in the MITRE Corporation’s Modeling & Simulation Engineering Department. He is a retired United States Army officer and has more than 15 years of experience in simulation and federation design and development for military training. He has supported several NATO Modeling and Simulation Group (NMSG) efforts and currently supports the Joint Staff J7's Joint Live, Virtual, Constructive (JLVC) Training Federation System Engineering Team. His interests include multi-resolution modeling and simulation interoperability. Mr. Bowers has a Bachelor of Science degree from the United States Military Academy and a Master of Science in Operations Research Systems Analysis and a Master of Science in Civil Engineering from Stanford University.

CURTIS BLAIS is a Research Associate in the Naval Postgraduate School Modeling, Virtual Environments, and Simulation (MOVES). He has over 38 years of experience in M&S research, development, management, and education. His research interests include application of web-based standards to C2 and M&S system interoperability and the development and use of human social culture behavior (HSCB) models. Mr. Blais holds Bachelor of Science and Master of Science degrees in Mathematics from the University of Notre Dame.

KEVIN GALVIN is the Architecture and Architecture Frameworks Lead in the Thales UK BTC based in Crawley, West Sussex. He joined Thales after 34 years military service in the British Army. He has over 18 years experience in architecture development and C2 interoperability including C2 to simulation. He holds a Bachelor of Social Science in Economic History and a Master of Science in Defence Modelling and Simulation and is a qualified Software Engineer.

KEVIN GUPTON is an Engineering Scientist in the Modeling and Simulation Information Management Group at the University of Texas at Austin, Applied Research Laboratories (ARL:UT). He has over 10 years of experience in enterprise system engineering, data modeling, and knowledge management. Mr. Gupton has developed net-centric applications and data services utilizing C4I and M&S common data standards and ontologies. He holds a Bachelor of Science in Mathematics and a Master of Science in Computer Science from Texas A&M University.

DR. KEVIN HEFFNER holds a BS in Engineering from the State University of NY at Buffalo and a Ph.D from the University of Paris VI. He has worked in the field of modeling and simulation for over 25 years. He has worked extensively in the areas of: aerodynamics, flight dynamics, flight simulator architectures, military modeling and simulation, C2-simulation interoperability and future concepts for Unmanned Aerial Systems. He supports NATO Modeling and Simulation Group activities including MSG-085.

LIONEL KHIMECHE is an R&T program manager in the field of M&S for planning and forces readiness at DGA (Direction Générale de l'Armement). His main topic of research deals with C2IS-Simulation interoperability. He co-chairs the NATO Technical Activity on C-BML (MSG-085) and leads several international projects and groups under bilateral cooperation (COMELEC, CAPRICORN) and the European Defence Agency (ATHENA, EUSAS).

CDR. HAKAN SAVASAN is Modeling and Simulation Head Engineer at Turkish Navy Research Center Command. He leads project teams responsible to design and develop distributed simulation systems to test and validate Naval Combat Systems. He is actively involving in Modeling and Simulation more than a decade.