Chapter 3 – C-BML REQUIREMENTS

This chapter presents the principal high-level requirements for C-BML that were identified as part of the MSG-048 Technical Activity. An exhaustive set of requirements for C-BML is out of the scope of this document. In the following sections, highlights of requirements that were identified during the Technical Activity are discussed. These are based primarily on reference [20] and work performed in the last two years of experimentation. These requirements are intended to serve as a starting point for further requirements elicitation and for input to the C-BML standardization development efforts.

Section 3.2 highlights the operational requirements that drive the need for the C-BML language. Section 3.3 presents some considerations and requirements that will be imposed on current and future systems that are designed and/or re-designed to benefit from BML-enabled capabilities.

3.1 CHARACTERISTICS OF A COALITION BML

The requirements have been elaborated by considering the various potential applications for C-BML and how the C-BML producing and consuming systems (e.g. C2, simulation and robotic systems) will collaborate to achieve the operational goals and objectives of the different coalition mission activities. The following sections identify some of the desired properties and characteristics that a BML infrastructure will have to possess.

3.1.1 Common Interface

A driving force behind C-BML has been the need to provide a seamless common interface among C2, simulation and robotic systems. This allows the operational user to interact with a C2 system and apply the same procedures in a real operational context or in a training exercise or for Mission Rehearsal. It allows for a single C2 system to exchange information with multiple simulation systems without requiring different C2-simulation interfaces to each simulation system.

As automated forces and robotic systems achieve higher levels of autonomy and automation, BML-enabled C2 systems also will provide a common interface in support of transmission of requests, transfer of control (e.g. sharing of robotic assets) or sharing of INTEL. For instance, there would be considerable advantages for a dismounted soldier to be able to control a Micro-UAV through a BML-enabled PDA or smartphone for information gathering and to be able to disseminate information rapidly and efficiently within his unit or to other units. Although this example is out of the scope of Coalition BML, this use-case illustrates a lower-echelon utilization of BML that will benefit from the definition of a common interface.

3.1.2 Expressiveness

The definition and specification of C-BML must enable the expression of all relevant actions to be performed by receiving force units (real or simulated) and robotic systems (e.g. tasking, reporting). In particular, it should be able to express a 5-paragraph OPORD and support for the military reports and tactical messages.

3.1.3 Unambiguousness

The unambiguous nature of C-BML expressions will allow for the construction of mathematical or machine representations of information such as tasks and orders such that simulations or robotic forces can process in

an automated manner. This implies the use of a formal grammar and requires significant analysis in order to support concepts such as the *command intent* and *desired end-state*.

3.1.4 Parseable

C-BML expressions must be able to be transferred across information systems with no human intervention in order to enable direct and automatic data transfer between C2, simulation and robotic systems. This will help to eliminate situations involving manual data transfer such as the so-called swivel-chair interface where an operator must assimilate and transfer information from one system to another. Similarly, this will allow for the phasing out of simulator operators that are required to support training scenarios by manually transferring orders to the simulation. These processes are both slow and error-prone.

3.1.5 Usability

The C-BML language must be easy to use: it must be straightforward to construct valid BML expressions and easy to learn to use C-BML interface for the exchange of message. The language should be designed to facilitate easy and quick input for specifying an order and/or submitting a report. C-BML is based on the JC3IEDM as an underlying data model, but should not require that all C-BML users (e.g. developers) be JC3IEDM experts.

3.2 OPERATIONAL CONSIDERATIONS FOR COALITION BML

The following sections provide some of the operational requirements for C-BML.

3.2.1 Support for WARNOs, OPORDs and FRAGOs

C-BML will need to support WARNOs, OPORDs and FRAGOs.

OPORD support should include the NATO five-paragraph OPORD.

Support for WARNOs is required in order to allow simulations to account for differing levels of readiness before executing an OPORD. This would likely translate into varying delays as simulated forces performed additional tasking or mission preparation activities.

FRAGOs are of course essential to many use-cases including, in particular, training and mission rehearsal.

3.2.2 Support for Doctrines

C-BML shall be able to support representation of doctrine. However, it should not be specific to a given national, joint or service doctrine but should provide the necessary elements to support a set of doctrines.

3.2.3 Support for Different Applications and Domains

C-BML should be independent of applications (e.g. training and course of action analysis) and be expandable to include new domains (e.g. maritime domain, air operations).

C-BML should be able to express the contents of tactical messages such as those specified in STANG 5500, the Allied Data Publication-3 (ADatP-3) [54].

Air Operations – C-BML should support air operations and therefore allow for constructing Air Tasking Orders (ATO) and Airspace Coordination Orders (ACO).

Naval Operations – C-BML should support naval operations and therefore allow for constructing messages in formats such as:

- General operational message (OPGEN);
- Operational Task (OPTASK);
- Operational Statistics (OPSTAT); and
- Naval OPORD.

3.2.4 Support for Levels of Command

C-BML should be applicable to multiple levels of command (i.e. not tied to one specific level such as brigade).

3.2.5 Rules of Engagement

C-BML should support a digitized form of rules of engagement.

3.2.6 Order of Battle and Task Organization

C-BML should support a digitized form of the Order of Battle (ORBAT). This requirement may be partially satisfied by the MSDL standard. During scenario execution, it is also necessary to be able to specify a Task Organization that may be different from one issued previously.

3.2.7 Common Operational Picture

C-BML should support information sharing required for C2 systems and simulations to interoperate and will provide a realistic operational context that includes information elements required to generate a Common Operational Picture (COP).

3.2.8 Weapons and Sensor Performance

C-BML should support information sharing of information elements that include performance data of sensors, weapons and platforms.

3.2.9 Logistic Data

C-BML should support information sharing required for C2 systems and simulations to interoperate and will provide a realistic operational context that includes logistics data.

3.2.10 Geospatial Data and Cultural Data

C-BML should provide for providing information elements of geospatial and cultural data in support of tasking and reporting.

3.2.11 Communications Infrastructure Data

C-BML should provide for providing information elements including data describing the availability of the network and communications infrastructure.

3.2.12 C-BML Expression Persistence

C-BML should be able to be stored and retrieved independently of the existence of an operational database (e.g. JC3IEDM).

3.2.13 Annotations[1]

C-BML should provide for the ability to add annotations to C-BML expressions for the benefit of human-machine interfaces that might display this information to an operator or for maintenance purposes. Even if one of the underlying goals of C-BML is to reduce or even eliminate the need for human intervention, human interactors will remain in the loop and could benefit from such annotations. In addition, support for annotations could be an element of a technology insertion plan that will facilitate the introduction of C-BML technology for use with legacy and new C2 systems.

3.3 C-BML-ENABLED SYSTEM CONSIDERATIONS

In order to fully benefit from C-BML-enabled capabilities, C-BML producing and consuming systems will need to comply with requirements that will allow for the successful exchange and subsequent interpretation of the C-BML expressions. In addition to C2, simulation and robotic systems, some of the requirements for the C-BML communications infrastructure are included in the following sections.

3.3.1 Standardization

C-BML should be made available through an international standards body such that national systems can be modified or extended as per a normative specification. Furthermore, this specification should be considered for adoption as a NATO Standardized Agreement.

C-BML should utilize established C2 standards, such as the MIP-JC3IEDM as applicable. However, this does not imply that C-BML cannot be used without the presence of a JC3IEDM database nor does it preclude the use of C-BML with other databases and data models.

3.3.2 C-BML Infrastructure Requirements

Time Management[2] – The C-BML infrastructure should provide for basic information management operations based on time-stamps that indicate when the message was issued (e.g. internal to the expression) and/or disseminated or published.

1) **Multiple Time-References** – The C-BML infrastructure should support several simultaneous time references, including: Physical time (i.e. the time being modelled), Simulation time (i.e. the simulation's representation of physical time) and Wall-clock (i.e. the time when the simulation is executed).

[1] The need for including free-text annotations is not unanimous within the group and therefore requires further analysis and clarification and will therefore be addressed as part of the MSG-085 Technical Activity.

[2] Definitions are taken from Fujimoto.

2) **Synchronization** – The C-BML infrastructure should provide for time synchronization across C-BML connected systems (e.g. using coordinated universal time UTC).
3) **Publication Time** – The C-BML infrastructure should provide for a publication time for each C-BML message that is published (e.g. Wall-clock and/or Physical time from C2/robotic systems and Simulation Time from simulation systems).
4) **HLA Time Management** – Although it cannot be assumed that all simulations will interoperate within an HLA federation, C-BML infrastructure time management services should be at least consistent with HLA time management services (e.g. available from HLA Run-Time Infrastructures (RTI)).

Persistence – In addition to the operational requirement to be able to store and retrieve C-BML expressions, there are several technical requirements that the C-BML infrastructure should also support.

1) **Storage of BML Expressions** – The C-BML infrastructure should provide for retrieving messages based on time-stamps, as discussed above.
2) **Filtering** – The C-BML infrastructure should provide for filtering criteria:
   - Scenario/simulation run;
   - Organization affiliation;
   - Expression type (e.g. position report, task status report, order);
   - Time criteria (e.g. wall-clock or physical time); and
   - User-defined filtering tag.

Validation – The C-BML infrastructure should ensure that published C-BML messages contain valid C-BML expressions (e.g. comply with the schema and business rules). Validation may also be required by C-BML expression-consumers.

Acknowledgement – The C-BML infrastructure should provide a mechanism for acknowledgement to publisher (i.e. C-BML message producer), when messages have been successfully received by C-BML message consumer.

Error-handling – The C-BML infrastructure should comply with a standard set of error-codes that provide feedback concerning errors with C-BML message validation or dissemination (e.g. acknowledgement). In the latter case, if there is a network disturbance, the C-BML message producer may want to be notified that his message has not been able to be received and also why it has not been able to be received (e.g. system failure, network anomaly, C-BML messaging service error, etc.).

3.3.3 C-BML Language Requirements

IEM Independence – C-BML language should be independent of C-BML Information Exchange Mechanisms (IEM).

XML-Based – Consistent with the NATO's orientation toward XML to promote the use of standardized message formats for military information exchange, C-BML should support an XML-based language [51].

[3]**Formal Language** – In order to be parseable, as mentioned in the first section of this chapter, C-BML must be a formal language; it therefore should be based on a formal grammar with a set of production rules.

[3] The possibility to include annotations is not contradictory with the requirement that C-BML be a formal grammar. It implies that they are parsed and possibly redirected to a graphical user interface, but are not interpreted by the C-BML message consumer.

3.3.4 Simulation Requirements

Faster Than Real-Time Execution – In order to support activities such as Course of Action Analysis (COAA) in a timely manner (e.g. in support of decision-support systems) it is necessary to run the simulation at rates that largely exceed real-time – otherwise it likely not to be possible to analyse a sufficient number of own COA and enemy COA fast enough to satisfy the commander's planning or decision-support needs. This requirement also has implications on the possible need to control simulation reporting rates, discussed below.

Simulation Report Management – Simulations may produce reports at rates that are higher than those that operationally relevant or realistic. It may be required by simulation systems to be able to restrict the rates at which reports are published in order to avoid overloading C2 systems that may not have been designed to accept high reporting rates.

Measures of Performance (MOP) and Measures of Effectiveness (MOE) – In the context of COAA, it is unlikely that the reports generated by simulations will be able to be used directly by C2 systems in order to rank the different plans and scenarios. As suggested above, C2 systems generally are not designed to process and display data originating from reports that are generated at high rates. It has been suggested by Abbott et al. [52] that the evaluation of plans based on simulation results will require the simulations to be equipped with metrics that can measure task performance based on Measures of Performance (MOP). These measures can then be used as the input for calculating the mission effectiveness in terms of higher-level metrics that measure the extent to which the mission goals have been achieved, i.e. Measures of Effectiveness (MOE). The definition of MOP and MOE as part of the C-BML language itself should be explored.

Simulation Initialization – Before executing a COA issued by a C2 system, simulation systems must be initialized with data including some or all of the following: scenario ID, time definition, weather, terrain, friendly/enemy/neutral/unknown organization and equipment status and position and initial tasking. Many of these requirements are covered by the existing SISO Military Scenario Definition Language (MSDL). Further analysis and additional work should be undertaken to adapt MSDL and C-BML for use together.

Simulation-to-Simulation C-BML Exchange – Although the focus of C-BML is not on simulation-to-simulation interoperability, there is potential benefit for simulations participating in the same federation to be able to exchange C-BML messages. Therefore, a case could be made for the definition of C-BML Federation Object Model (FOM) and the possible use and benefits of a C-BML FOM should be explored. This is consistent with the requirement that the C-BML language be defined independently of the IEM that is used as a vehicle for exchanging C-BML expressions (see Section 3.3.3).

3.3.5 Command and Control System Requirements

The following considerations do not apply to all C2 systems, but rather offer some general insight based on the observations and experience of those that participated in the MSG-048 experimentation programme.

Increased Usability – As C-BML empowers C2 systems with the capability to rapidly assess plans, in some instances, they may need to be modified in order to provide for rapid plan modification – which may not have been a requirement at the time the C2 system was designed.

Native C-BML Interfaces – Many benefits of C-BML may be achieved without modifying existing C2 systems, as demonstrated during the MSG-048 experimentation programme. However, in time, in order to

fully benefit from advanced C-BML-enabled capabilities, C2 systems may require upgrades to include native C-BML interfaces, either in current or future systems.

3.3.6 Robotic Forces and Automated Systems Requirements

Robotic Force systems, such as Unmanned Air Systems (UAS), have similar interface requirements as simulations, but there are significant differences. Existing standards for interfaces to UAS, such as STANAG 4586 [53], should be examined and analyzed in order to determine how C-BML expressions could be leveraged for tasking UAS and for receiving reports from UAS. In particular, STANAG 4586 calls for the use of a sub-set of ADatP-3 as the basis for tactical messages for the UAV Ground Control Station Command and Control Interface (UAV GCS CCI). Similarly, the Joint Architecture for Unmanned System (JAUS)[4] specification currently under development by the SAE International also defines a set of interfaces for tasking Unmanned Vehicle Systems (UVS).

Further work is required to investigate the suitability of C-BML to interface directly with UVS.

[4] http://www.openjaus.com/understanding-sae-jaus.