Opportunities for Next Generation
BML: Semantic C-BML

Samuel Singapogu
C4I Center, GMU
Outline

• Introduction
• Current state of C-BML
• Need for a semantic C-BML
• Enhancements provided by a semantic layer
• Elements in a Semantic C-BML
• Reasoning through a Knowledge representation
• Future Work and Conclusions
Introduction- BML

• Battle Management Language is an unambiguous language to facilitate the command and control of forces and equipment in a military environment and to provide information for situational awareness.

• BML has a accompanying grammar- Command and Control Lexical grammar (C2LG)

• One of the goals of BML is to provide “Shared Semantics between C2 and M&S via a Common Tasking Description”

• BML is based on work initiated by the C4I Center outlined in [Carey, S., M. Kleiner, M. Hieb, and R. Brown, “Standardizing Battle Management Language – A Vital Move Towards the Army Transformation,” IEEE Fall Simulation Interoperability Workshop, September 2001]
Introduction: C-BML

• C-BML is applying BML to a coalition context
• A standard has been approved following a SISO balloting process
  – Based on a specification provided by the C-BML product development group (Blais, Curtis, et al; SISO Fall 2011 SIW)
  – My work provides insight into the use of OWL in phase 2 standardization
• Phase 1 focused on formalizing syntax in terms of a XML schema
  – Vocabulary based on the JC3IEDM data model
  – Sought to provide full expressivity of the data model
Limitations in the current standard

• An XML based system built on the full expressivity of the JC3IEDM limits the speed of
  – Development/extensions
  – Integration
  – Testing

• Interoperability in Phase 1 is on the syntax level

• The need for a semantic C-BML has been suggested by Blais, Turnitsa, and Gustavsson
  (SISO Fall SIW 2006); my work provides:
  – A path forward in using upper level ontologies
  – A context for the use of reasoning in semantic C-BML
Introduction: Semantic Web

- The semantic web is a framework of linked web data in a shared machine-readable knowledge representation
  - Shared semantics
  - Linked data
  - Machine readable
- Based on W3C standards
- Semantic representation enables:
  - Formalization of shared semantics
  - The ability to infer knowledge
  - The use of a reasoner to check for semantic inconsistencies
Current standards in ontology creation

• Resource Description Framework
  – Based on Uniform Resource Identifier (URI)
  – Any element can be defined (and disambiguated) using a URI
  – Knowledge is represented using a <subject> <predicate> <object> triplet
An example of a semantic representation (in C-BML context)

- Representation of a Unit using URI:
  http://urlNamespaceOfUnit:UnitA

- Representation of a relationship:
  http://urlNamespaceOfUnitRelationship:UnitHasAsCommander

- An RDF axiom (asserted knowledge in the ontology):
  <http://urlNamespaceOfUnit:UnitA>
  <http://urlNamespaceOfUnitRelationship:UnitHasAsCommander>
  <http://urlNamespaceOfUnit:UnitA>

- RDFS is:
  - Flexible
  - Easily scalable
Introduction: OWL

- Web Ontology Language is a current standard to model and represent knowledge in the form of an ontology
  - The goal is to model and represent knowledge in a machine readable fashion
  - Based on RDFS, can be serialized to XML
  - Compliant to description logic, which makes it computationally decidable and has adequate logical expressivity
  - Available reasoners can be used to derive inferred knowledge
- OWL is a W3C standard (http://www.w3.org/TR/owl-features/);
Why does C-BML need an ontology?

- It formalizes the definition and **meaning** of common terms
- It formalizes the **doctrinal** rules for Orders and Reports
- It eases **interoperability** because of a shared vocabulary and defined meaning
- It allows performing powerful **reasoning** on operational semantics
- A model driven(ontology-driven) framework facilitates easy extensions
  - Gupton, Blais and Heffner (International Journal for Intelligent Decision Support Systems, October 2011) suggest model based data engineering as a development approach; my work lays a foundation for OWL as a central data model.
A path to creating semantic C-BML

• Evaluate relevant upper level ontologies
• Extract semantic pieces from existing Phase 1 work
  – Entities recognition (XSD elements, XSD types)
  – Taxonomy classification (subsumption relation, IS-A relationships)
  – C-BML specific relationships
  – Doctrine based axioms
Upper Level Ontologies

• General purpose ontologies that define entities in a particular domain (time, geography, ...)

• Why use upper level ontologies?
  – Reusability
  – Easier extension management
  – Easier mapping between systems that use the same upper level ontology

• Gupton, Blais and Heffner (IJIDSS, October 2011) have proposed the use of upper level ontologies; my work:
  – Identifies applicable upper level ontologies and their alignment with C-BML
  – Provides a context for reasoning on semantic C-BML
Upper level ontologies relevant to C-BML

• C-BML vocabulary is based on the 5Ws (‘Who’, ‘What’, ‘When’, ‘Where’ and ‘Why’)

• OWL-Time (*relevant to ‘When’*)
  – Upper level ontology that represents time in different forms, temporal constraints and axioms
  – The core class “TemporalEntity” has two sub classes: Instant and Interval

• Geo-OWL (*relevant to ‘Where’*)
  – Upper level ontology to represent geometric shapes
  – The top class geometry can be of type Point, LineString, Polygon…
Mapping between C-BML and OWL Time

Schema definition of C-BML ‘When’

- cbml:TaskWhenLightType
  - MinimumDuration
    - The numeric value that represents a quantity of time in milliseconds for the minimum permissible period of...
  - EstimatedDuration
    - The numeric value that represents a quantity of time in milliseconds for the estimated period of effectiveness of a...
  - MaximumDuration
    - The numeric value that represents a quantity of time in milliseconds for the maximum permissible period of...

- StartWhen
  - Specifies the start time of a task.

- EndWhen
  - Specifies the end time of a task.

Entities in OWL Time

- Duration description
- DateTimeDescription

ICCRTS-2014 paper 046
Advantages of using OWL-Time

• Rich expressiveness of time entities
  – Models both time instants and time intervals
  – Temporal constraints can be established

• Powerful reasoning over temporal concepts
  – Task1 after Task2 can be modeled
  – New temporal relationships can be inferred using a reasoner
Mapping between C-BML and Geo-OWL

Schema definition of C-BML ‘AtWhere’

Entities in Geo-OWL

- Point
- LineString
- Polygon
Reasoning in semantic C-BML

• Reasoning has two main goals
  – Checking for semantic inconsistencies
  – Gaining inferred knowledge

• OWL captures knowledge in a way that existing reasoners (HermIT, Jena..etc) can automatically derive new knowledge

• OWL reasoners are based on First order predicate logic
Semantic C-BML reasoning example 1

• Checking for semantic inconsistencies
  – An example in the “reports” context: “Executer(x),
    hasAsReportedTime(x,t1) ∧ Executer(x),
    hasAsReportedTime(x,t1) → (hasAsLocation(x,l1)=hasAsLocation(x,l2))
  (SWRL syntax)

  Checks to make sure that multiple reports provide consistent reported data locations of a “Executer”

• A ontology provides a formal, machine understandable way to check for semantic inconsistencies
Semantic C-BML reasoning example 2

Deriving inferred knowledge:

Example asserted axioms:

Axiom 1: $\text{ObjectProperty}(a:\text{isAUnit}\ \text{domain}(a:\text{Tasker})\ \text{range}(a:\text{Unit}))$

Axiom 2: $\text{ObjectProperty}(a:\text{isAsubordinateOf}\ \text{domain}(a:\text{Taskee}) (a:\text{Tasker}))$

The knowledge represented by these Object properties are:

1: A Tasker should be a Unit (as opposed to a Equipment)

2: A Taskee is subordinate to a Tasker
Semantic C-BML reasoning example 2 continued

Example Task:
Tasker: “1060: 1st Battalion Commander”
Taskee: “1062: Company A”

Using the two axioms, we can infer the knowledge:

“’1060: 1st Battalion Commander’ is a Unit who is a commanding officer to ‘1062: Company A’”
Semantic C-BML reasoning example 3

**Axiom**: ObjectProperty(a:isAfterTask domain(a:Task) range(a:Task))

*Note: Object Properties are transitive*

**Assertion1**: Task1 isAfterTask Task2,
**Assertion2**: Task2 isAfterTask Task3

The inferred knowledge is:

*Task1 isafterTask Task3*

*Note: This inference can be derived in the Ontology even if the two tasks are in separate C-BML Orders/Tasks*
Conclusions

• Semantic C-BML can help in:
  – Model driven development and ease of scalability and extension management
  – Better interoperability with shared semantics and common, formal conceptualization
  – Formalization of doctrinal rules/axioms and semantic restrictions
  – Checking for semantic inconsistencies

• OWL is an adequate language to model C-BML
Future Work

• Development of C-BML in a model driven framework
  – OWL could be used as the central semantic model
  – Alignment with the MIP information Model and Change Proposals (CP)

• Abductive reasoning as a way to hypothesize knowledge based on reported data

• Explore ways to extract semantic elements from Phase 1 specification
  – How do XML schema schema elements relate to entities?
  – What relationships can be extracted from XML schemas?