C2SIM Infrastructure

Dr. Mark Pullen

APPROVED FOR PUBLIC RELEASE
Presentation Overview

• C2SIM overall architecture
• Role of schema
• Role of C2 systems (C2IS)
• Role of simulations
• Role of servers
• Need for interfaces
BML: Definition
WHERE WE STARTED

• BML - an unambiguous language to:
  
  ➢ Command and control live and simulated forces conducting military operations, and
  
  ➢ Provide for situational awareness and a shared, common operational picture.

*Shared Semantics between C2 and M&S via a Common Tasking Description*
BML Architecture

Command and Control Systems

BML Messages
(Orders, Reports, etc.)

Simulation Systems

BML Web Services
+ Initialization and Synchronization

real-time database
A C2SIM coalition is a system of systems.
Example: MSG-048 2009 Architecture

The diagram illustrates the Architecture for the MNF C2IS (Military Networked Forces Command and Control Information Systems) and the OPFOR C2IS (Opposing Force Command and Control Information Systems) with interconnected components such as BattleView, ICC, NORTaC, SICF, C2LG, and ABCS. The DIS SIM EXERCISE includes OneSAF, UAV-Sim, JSaf, SIMBAD, and APLET. The diagram also indicates that ICC, C2LG, and JSAF are provided by specific entities.
Later Example: MSG-085 Demonstration
What is a Web Service?

• Generalization of servers that provide webpages in the Internet
• Functions on a single transaction basis (stateless)
  • Get/Pull and Post/Push
• Makes a TCP connection for each one
• Can be seen as a Remote Procedure Call
  • Using Simple Object Access Protocol (SOAP)
• Or as a way to share and fetch XML documents
  • Could use Representational State Transfer (REST)
  • REST is more efficient so is used more often
Publish/Subscribe for Web Services

• Message must go to all interested systems
• Polling by clients is inefficient
• Streaming Text Oriented Messaging Protocol (STOMP) takes care of distribution
  • Systems subscribe for Topics of interest
  • STOMP server copies each message to subscribing systems
Why XML?

- Extensible Markup Language (XML) uses a generalization of the HyperText Markup Language (HTML) used for webpages
  - Information expressed as `<tag>value</tag>`
  - For example:
    - `<title>Battle Management Language</title>`
  - Permissible tags defined by a “schema”
- Defined set of metadata tells how to use document
  - Data about the data
- Document organized as a “tree” starting from “root”
XML Example Report

```xml
<?xml version="1.0" encoding="UTF-8"?>
<BMLReport>
  xmlns:jc3iedm="urn:int:nato:standard:mip:jc3iedm:3.1a:oo:2.0"
  xmlns:bml="http://netlab.gmu.edu/IBML"
  xmlns:msdl="http://netlab.gmu.edu/JBML/MSDL">
  <Report>
    <CategoryOfReport>StatusReport</CategoryOfReport>
    <TypeOfReport>GeneralStatusReport</TypeOfReport>
    <StatusReport>
      <GeneralStatusReport>
        <ReporterWho>
          <bml:UnitID>1-22</bml:UnitID>
        </ReporterWho>
        <Hostility>FR</Hostility>
        <Executer>
          <bml:Taskee>
            <bml:UnitID>1-22</bml:UnitID>
          </bml:Taskee>
        </Executer>
      </GeneralStatusReport>
    </StatusReport>
  </Report>
</BMLReport>
```
<OpStatus>MOPS</OpStatus>
<WhereLocation>
  <bml:GDC>
    <bml:Latitude>33.424079</bml:Latitude>
    <bml:Longitude>44.682716</bml:Longitude>
    <bml:ElevationAGL>2.054</bml:ElevationAGL>
  </bml:GDC>
</WhereLocation>
<When>20070101000000.000</When>
<ReportID>506</ReportID>
<Credibility>
  <bml:Source>HUMINT</bml:Source>
  <bml:Reliability>A</bml:Reliability>
  <bml:Certainty>RPTFCT</bml:Certainty>
</Credibility>
</GeneralStatusReport>
</StatusReport>
</Report>
</BMLReport>
XML Schema

- **Schema** is a representation of the organization and format that is allowable for a given XML file, for example this slice:

  ```xml
  <xs:complexType name="ReportHeaderLightType">
    <xs:sequence>
      <xs:element name="ReporterWho" type="cbml:ReporterWhoType"/>
      <xs:element name="ReportingData" type="cbml:ReportingDataType">
        <xs:choice>
          <xs:element name="AbsoluteReportedWhen" type="cbml:AbsoluteReportedWhenLightType"/>
          <xs:element name="RelativeReportedWhen" type="cbml:RelativeReportedWhenLightType"/>
        </xs:choice>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
  ```

- Each Type specifies data order and format for a “chunk” of XML
Interconnected C2SIM Systems

• Command and Control systems
• Simulation systems
• Servers
• Graphic User Interfaces
• Status monitoring and control
MSG-085
Coalition C2SIM

Coalition C2-Simulation Web Services*
Startup Synchronization
Initialization Aggregation
Orders & Reports Repository

*one or more servers

Mobile Web Client

Overall C2

National Ground C2

Air & Maritime C2 Systems

C2 Systems possibly linked using JC3IEDM

Threat Simulation

National Simulations (including OneSAF)

Air & Maritime Simulations

Simulation systems possibly federated using HLA or DIS
Interfacing C2 Systems

• C2 systems produce Orders, consume Reports
• To enable C2SIM, add an interface module that follows the agreed schema so the C2 system can:
  • Send the server an XML document for each Order
  • Subscribe to Reports distributed by the server and present them as situational awareness
• Clearly identify when running in simulated mode
• Support start/stop of simulated operation
Interfacing Simulation Systems

• Simulation systems accept Orders, produce Reports

• To enable C2SIM, add an interface module that follows the agreed schema so the simulation system can:
  • Send the server an XML document for each status change that requires a Report
  • Subscribe to Orders distributed by the server and follow the directions they contain
  • Start/stop simulation operation under coalition control
C-BML/MSDL Servers
Primary Server Functions

• Accept Push/Post of XML documents and store
  • C-BML Orders and Reports; MSDL scenario files
• Accept Subscriptions by Topic
  • e.g. all General Status Reports
• Publish documents to subscribers as they arrive
  • And respond to Get/Pull for them
Additional Server Functions

• **Namespaces**
  - XML tag names can be qualified by addition of a “namespace” code: `<bml:Report>`
  - This allows tag names from different sources to work together safely

• **Schema validation**
  - Server confirms that each document received conforms to the schema
  - This identifies a likely source of incompatibilities
  - However, it slows the service

• **Filtering data**
  - Restrict delivery based on user-defined criteria
More Server Functions

- **Logging/replay**
  - Server writes a file showing every transaction it receives, with time stamps
  - Server is capable of replaying this file to recreate the original sequence of Orders and Reports at original time intervals

- **Distributed servers**
  - Multiple servers can be tied together to increase load capacity and geographic scope of the C2-Sim coalition
Simple Distributed Server Architecture

B2B: back-to-back client
Server Schema Translation

• Needed because developing organizations are reluctant to change their interface each time a new schema is developed
  • So we end up with C2 and simulation systems interfaced to different (but mostly equivalent) schemas

• Server parses the XML document according to appropriate schema; produces output conforming to different designated schema
  • Possible only where data support the same semantics

• This capability allowed MSG-085 to interoperate C2 and simulation systems that had been interfaced under various previous schemas
MSDL Server

- MSDL inputs must be aggregated from all participating systems
  - In a coalition each C2 and simulation system can have different initialization requirements
  - A consolidated MSDL file is needed for consistency
  - Server can aggregate them automatically
  - A change on any system is reflected to all
MSDL Server Operation

- **Administrator** Initialize
- **Client** Add Units and Relations
- **Client** Add Units and Relations
- **Client** Add Equipment and Relations
- **Administrator** Publish

Master Controller

SBML Server

MSDL

Publish
Other Supporting Software
For C-BML/MSDL Communications
Graphical User Interfaces (GUIs)

- During development it is very useful to have a generic way to generate and inspect BML documents
  - Create an Order, Report, Request etc. and introduce it to server
  - Accept and display Order, Report, etc. from client system or server
  - Edit either of the above
- The GUI can serve as a limited/surrogate C2 system for experimentation
BML GUI Functions

- Editing a C-BML or MSDL document
- Merging MSDL documents
- Serialization of document
- Grammar validation of document
- Schema validation of XML document
- Auto-configuration to schema

- Pulling a document
- Pushing a document
- Subscription to server Topics
- Retrieving latest reports
- C2 capability
- Displaying maps with overlays
- Geolocation entry from maps
BML GUIs in Use Today

- **C2LG GUI**
  - by Fraunhofer FKIE
  - Additional features for C2 of robots

- **BML C2 GUI**
  - Open source by GMU
  - Generic
BML C2 GUI
Virtual Private Network (VPN)

- Enables a private, distributed enclave over Internet
  - Commercial and open source versions available
  - Good for sensitive but unclassified information
- Used by MSG-085 for
  - Development and testing
  - Remote participation in Final Demonstration
- We are working to establish C2SIM service
  - available by VPN 24x7