A Web-Based Coordination System for MSDL/C-BML Coalitions

Priscilla McAndrews
Lisa Nicklas
Dr. Mark Pullen

GMU C4I Center
Presentation Outline

• Introduction
  • C-BML/MSDL Coalitions

• Status Monitor and Control:
  • What and Why
  • Design
  • Automated interface

• Experience: I/ITSEC 2011
• Conclusions
Background

• SISO is developing two standards to support C2-simulation interoperation for international coalitions

  • Military Scenario Definition Language (MSDL)
    • Standardizes initialization of simulation systems
    • Recently shown also to be useful with C2 systems

  • Coalition Battle Management Language (C-BML)
    • Orders, Requests, and Reports among C2 systems and simulations

• This paper addresses a problem arising from the complexity of operating such coalitions
  • Need to synchronize operations of the various systems
  • Need to keep the operators informed as to status
BML in NATO Modeling and Simulation Group (MSG)

- MSG has been nexus for BML cooperation
- US Joint BML project with France Aplet
  - Early demonstration 2006
  - Led to Exploratory Team 16
- MSG-048 Technical Activity started 2007
  - Demonstrations at I/ITSEC 2007 and 2008
  - One-week experimentation 2009 (9 nations)
- MSG-085 Technical Activity started 2010
  - Goal: operational use of MSDL/C-BML (12 nations)
Example: MSG-048 2009 Architecture
Status Monitoring and Control

- Lesson learned in MSG-048:
  - It is impractical to coordinate multiple interoperating C2 and simulation systems with only spoken communication

- Solution: a coordinating webpage
  - Shows possible states of each coalition system
  - Master Controller provides coordinating guidance
  - Inputs can come from webpage interface
  - Or Web service client
Generic BML Architecture

Command and Control Systems

BML Messages (Orders, Reports, etc.)

Simulation Systems

C-BML/MSDL Web Services

JC3IEDM and other databases

Web-Based Coordination System
MSDL Data Components

- Geographic Region of Interest
- Force/Sides
- Units
- Equipment
- Installations
- Overlays
- Graphics
Coalition C2-Simulation Architecture

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

*based on Scripted Services

Mobile Web Client
Overall C2
National Ground C2
Air & Maritime C2 Systems
C2 Systems linked using JC3IEDM

Threat Simulation
National Simulations (Including OneSAF)
Air & Maritime Simulations
Simulations federated using HLA or DIS

Web-Based Coordination System
Status Monitor and Control Functions

• Start, pause, and stop all coalition systems in a synchronized way

• Participating systems
  • Coalition Server – accepts XML documents and publishes to participating systems
  • SMC Server – interacts with humans via webpage; may also interact with other systems via XML
  • C2 System Clients – coalition command and control
  • Simulation System Clients – simulate coalition forces
  • Master Controller – provides overall intended status
  • Client System Controllers – operate C2 & simulation
  • Observers – monitor status but control nothing
C2 and Simulation Systems are Different

• C2 systems:
  • Provide orders and display situation
  • May not require MSDL initialization
  • May not have a concept of “stopped” or “paused” since military operations are continuous

• Simulation systems:
  • Generate data from model - representing expected actions of coalition forces and their results
  • May perform checkpoint/restart
Master Controller Inputs

- Stop
- Initialize (start MSDL generation & loading)
- Run
- Pause
**MSG-085 Status Monitor**

**MASTER CONTROLLER STATUS**

**Scenario:** scenario1  
**Current Order:** stop

**Scenario MSDL status:** not started  
**Number of observers:** 2

- Change current order: stop  initialize  run  pause

- Add/change comment:

- Log out client/observers:

- Publish MSDL  Reset All  Master log out

**CURRENT CLIENT STATUS**

<table>
<thead>
<tr>
<th>Stopped</th>
<th>Initializing</th>
<th>Ready</th>
<th>Running</th>
<th>Paused</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>C2IS1</td>
<td>rpt1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C2IS2</td>
<td>rpt2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C2IS3</td>
<td>rpt3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C2IS4</td>
<td>rpt4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C2IS5</td>
<td>rpt5</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C2IS6</td>
<td>rpt6</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C2IS7</td>
<td>rpt7</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Not started
- Offline
- Setting up

- new timeout
- new comment
Client Status Phases

- Stopped
- Initializing
- Ready
- Running
- Paused
MSG-085 Status Monitor

MASTER CONTROLLER STATUS

Scenario: scenario1
Scenario MSDL status: not started
Current Order: off-line
Number of observers: 2

CLIENT STATUS

C2IS1-rpt1 Current Status: stopped
Change status: 
Add/change comment: power down
Client log out

<table>
<thead>
<tr>
<th>Status</th>
<th>Ready</th>
<th>Running</th>
<th>Paused</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>C2IS1</td>
<td></td>
<td></td>
<td></td>
<td>power down</td>
</tr>
<tr>
<td>C2IS2</td>
<td></td>
<td>setting-up</td>
<td></td>
<td>delay</td>
</tr>
<tr>
<td>C2IS3</td>
<td></td>
<td>setting-up</td>
<td></td>
<td></td>
</tr>
<tr>
<td>C2IS4</td>
<td></td>
<td></td>
<td></td>
<td>all OK</td>
</tr>
<tr>
<td>C2IS5</td>
<td></td>
<td>setting-up</td>
<td></td>
<td></td>
</tr>
<tr>
<td>C2IS6</td>
<td></td>
<td>MSDL Pushed</td>
<td></td>
<td>waiting</td>
</tr>
<tr>
<td>C2IS7</td>
<td></td>
<td>off-line</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
SMC Layout

- **General layout:**
  - Heading (identifies system/role)
  - Current directive from Master Controller
  - MSDL Scenario name and status
  - Number of observers logged in

- **Input areas:**
  - Overall directive (Master Controller only)
  - Dropdown to provide state (Client Controller only)
  - Text box for comments

- **Output area:**
  - Last provided status of each client (initially stopped)
Automated Interface

- Human Client Controllers are not good status sources: they experience many distractions
- We have created a RESTful Web Service that lets the C2 and simulation systems interface with SMC
- Input methods:
  - c2SimAdd
  - c2SimModSlientStatus
  - csSimModClient
  - c2SimModSessionKey
  - csSimModAutomated
- Output: C2 and simulations can subscribe to status
Interaction Control Flow

Web-Based Coordination System
SMC Experience: I/ITSEC 2011
MSG-085 I/ITSEC 2011 DEMONSTRATION

MSDL for initialization, C-BML for execution

Demo Harness 1: NOR, UK, US

COALITION TRAINING CAPABILITIES
Combined C2-Simulation Initialization
Automated Order Execution
Automated Reporting

DISTRIBUTED TRAINING

TRAINING VIGNETTES
Air Reconnaissance
Combined Ops with Logistics
Ground Maneuver

Web-Based Coordination System
System Architecture for I/ITSEC 2011 DH1

USA - I/ITSEC & GMU
- ICC/JADOCS Clients
- JSAF GUI
- OneSAF
  - C-BML
  - DIS
  - C-BML + DIS
- GMU WS 2.5 Status Server
  - VPN server
  - C-BML

UK
- ICC/JADOCS Servers
  - C-BML
- JSAF
- UK C-BML Translators
  - C-BML + DIS
  - C-BML

Norway
- NorTAC
  - C-BML

Internet
- VPN Tunnel

Merged MSDL

Web-Based Coordination System
Conclusions

• Experience indicates SMC is useful and necessary
  • Both I/ITSEC and continued coalition testing
• Webpage-based play is simple and effective
• Human status input is a weak link
• For best results, software-to-software interface via Web Service is best
• SMC is available as open source from http://c4i.gmu.edu/OpenBML