Evaluating the Proposed Coalition Battle Management Language Standard as a Basis for Enhanced C2 to M&S Interoperability

Dr. J. Mark Pullen
Dr. Michael R. Hieb
C4I Center
George Mason University
Fairfax, VA, USA

Michael W. Powers
US Army Engineer R&D Center
Topographic Engineering Center
Fort Belvoir, VA, USA

Lionel Khimeche
DGA/DETSAIS
Arcueil, France

Major Kevin Galvin
SO2 Live and Virtual Training
UK MoD
London, UK

Presentation Outline

- Definition and History of BML
- MSG-048 relationship with SISO C-BML
- US JBML Phase 1 Project
- MSG-048 Activities
- Summary/Conclusions

Definition and History of BML

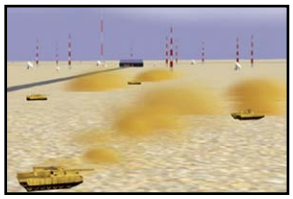
The Problem

- Our current BML is a loosely knit “language” tailored to interpersonal communication
- Vocabulary is found in Doctrine Manuals (such as US Army FM 101-5-1), but it lacks clearly delineated rules governing its use (semantics and syntax)
- It is riddled with ambiguity and overlapping definitions
- As such, it is incapable of transitioning to the full range of automation that the DoD is implementing
- It will not support the integration of advanced modeling and simulation with “digitized” command and control

The Requirement

If we are to train as we fight, then we must be able to communicate command and control information via the same C4I devices in all environments:

- Live training and operations (among humans)
- Simulation training, mission rehearsal, and decision aids with the C4I devices stimulating and being stimulated by simulations. (Live, Constructive, Virtual simulation)

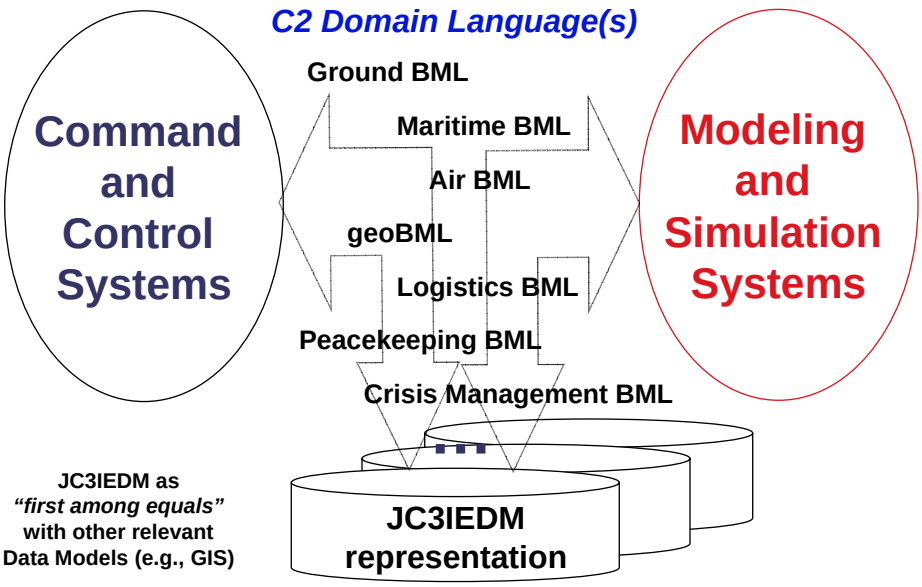


Battle Management Language (BML)

- 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 Vision



JC3IEDM as
"first among equals"
with other relevant
Data Models (e.g., GIS)

MSG-048 and SISO C-BML

BML Activities

	Specification	Ground	Air	Naval	Implementation	Software Services	International
C-BML	X	X	X	X		X	X
ET-016		X			X		X
MSG-027		X			X	X	X
JBML	X	X	X	X	X	X	
geoBML	X	X			X		
XBML		X			X	X	X
Army BML		X			X		
AOBML			X		X		
MIP/JC3IEDM	X	X	X	X			X

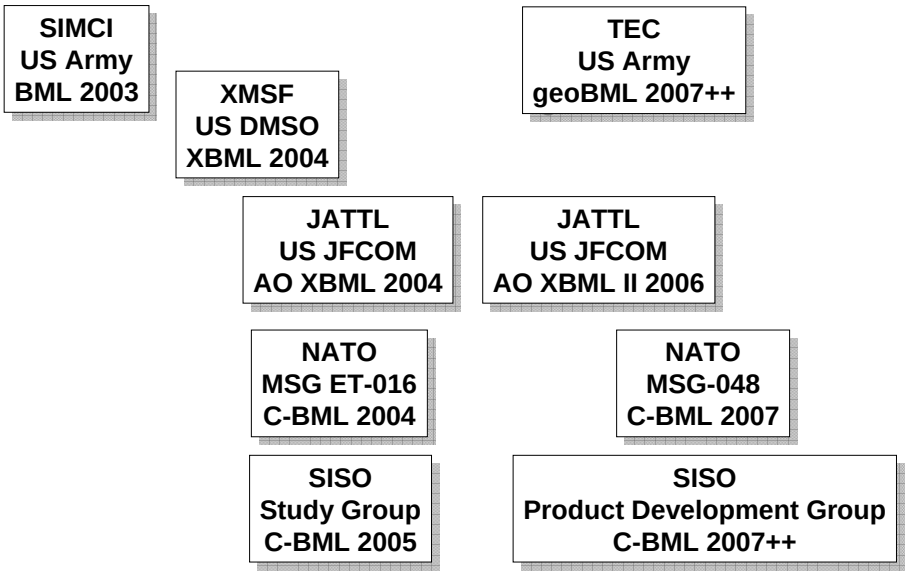
SISO C-BML

- The Simulation Interoperability Standards Organization (SISO) Coalition Battle Management Language
  - Study Group 2005-2006 recommended development of a standard for C2-Simualtion interoperability
  - Product Development Group charter now pending
- Three phases:
  - Phase 1: XML schema using C2IEDM/JC3IEDM
  - Phase 2: Give the language a grammar
  - Phase 3: Ontology-based operation
- Currently focused on getting Phase 1 draft to ballot April 2008

NATO MSG Technical Activity 048

- MSG Exploratory Team 016 considered the potential of BML to facilitate C2-Simulation interoperability for coalitions
  - Recommended three-year technical activity to experimentally evaluate use of SISO C-BML
  - Wrapped up with FR-US demonstration indicating relative ease (and potential pitfalls) of technique
    - Presented in MSG Symposium last year
- Technical Activity MSG-048 is undertaking experimental evaluation 2006-2008
- More detail later in this presentation

Genealogy of BML



US JBML

JBML Phase 1

- Develop an initial Joint BML capability for
  - Ground BML
  - Air BML
  - Maritime BML

  in one common language
- Define this common Language with several (Service specific) interoperating domains
  - Common components for shared information
  - Service-specific components for unshared information
  - Shared common controlled vocabulary (based on JC3IEDM definitions)
  - A Lexical grammar informing the schema
- Make the results available to SISO C-BML for standards development and to MSG-048 as US component of experimental evaluation
  - Key component: open source Web services

JBML Architecture

JBML Layered Services

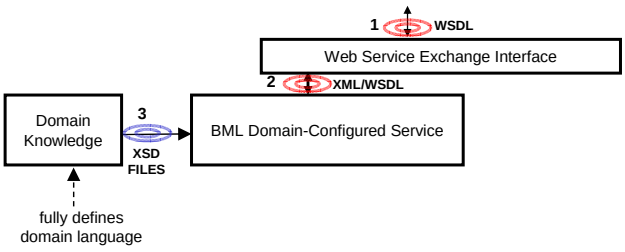
- Top layer: Domain-Configured Service
  - XML-encoded “language”
  - Characterized by XML schema
  - Implemented as Web service
- Middle layer: BML Base Service
  - Composite “building block” transactions applicable to multiple domains
  - Characterized by mappings from XML schema to JC3IEDM entities (database tables)
  - Implemented as software service (optionally accessible as Web service)
- Bottom layer: Common Data Access Service
  - Software service providing wrapper around SQL
  - Optionally accessible as Web service

JBML Domain-Configured Service (DCS)

- Purpose:
  - Provide a high-level, semantically consistent, XML-based language definition
  - Modular and readily extensible
  - Structure based on C2 Lexical grammar of Hieb & Schade
    - Gives high confidence the language will meet BML needs
    - No actual grammar processing yet; just a way to structure JBML

<task> (verb)	<tasker-who>
<taskee-who>	<affected-who>
<what> (action)	<where>
<start-when>	<end-when>
<why>	<label>
<modifier>	

JBML Service Architecture Top Layer



JBML DCS OrderType

```
<xsd:annotation>
    <xsd:documentation>Provides basic information that applies to all Tasks in the order
     </xsd:documentation>
</xsd:annotation>
<xsd:complexType name="OrderType">
    <xsd:sequence>
       <xsd:element name="OrderMode" type="OrderModeType"
                                default="SINGLE“ minOccurs="0"/>
       <xsd:element name="TaskersIntent" type="FreeTextType" minOccurs="0"/>
       <xsd:element name="Task" type="TaskType" maxOccurs="unbounded"/>
       <xsd:element name="OrderIssuedWhen" type="WhenType"/>
       <xsd:element name="OrderID" type="OrderIDType"/>
       <xsd:element name="TaskerWho" type="WhoType"/>
       <xsd:element name="TaskOrganization" type="msdl:TaskOrgType“
                                 minOccurs="0"/>
       <xsd:element name="EnemyTaskOrg" type="msdl:TaskOrgType"
                                minOccurs="0"/>
       <xsd:element name="ControlMeasures" type="MultipleControlMeasuresType"
                                minOccurs="0"/>
       <xsd:element name="TargetList" type="TargetListType" minOccurs="0"/>
    </xsd:sequence>
</xsd:complexType>
```


JBML DCS Joint TaskType

```
<xsd:annotation>
    <xsd:documentation>
          Defines the domain of warfare associated with the task
    </xsd:documentation>
</xsd:annotation>
<xsd:complexType name=“TaskType">
    <xsd:choice>
          <xsd:element name="GroundTask" type="GroundTaskType"
                              minOccurs="0"  maxOccurs="unbounded"/>
          <xsd:element name="AirTask" type="AirTaskType"
                              minOccurs="0" maxOccurs="unbounded"/>
          <xsd:element name="MaritimeTask" type="MaritimeTaskType"
                              minOccurs="0" maxOccurs="unbounded"/>
    </xsd:choice>
</xsd:complexType>
```

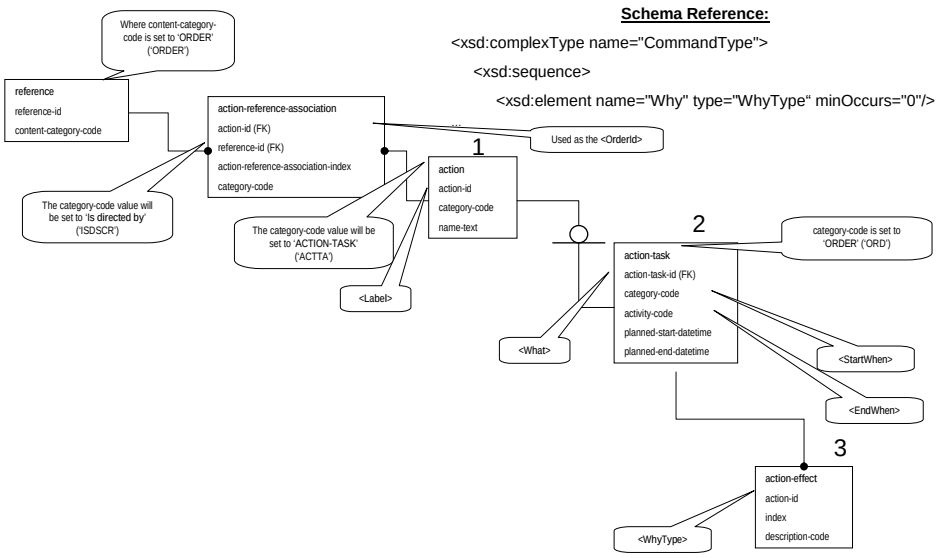
JBML DCS GroundTaskType

```
<xsd:complexType name="GroundTaskType">
    <xsd:sequence>
        <xsd:element  name="TaskeeWho" type="WhoType"/>
        <xsd:element name="What" type="GroundWhatType"/>
        <xsd:element name="Where" type="WhereType"/>
        <xsd:element name="StartWhen“ type="WhenType"/>
        <xsd:element name="EndWhen" type="WhenType"
                        minOccurs="0"/>
        <xsd:element name="AffectedWho“ type="WhoType”
                        minOccurs="0"/>
        <xsd:element name="Why" type="GroundWhyType"
                        minOccurs="0"/>
        <xsd:element name="Label" type="LabelType"/>
    </xsd:sequence>
</xsd:complexType>
```

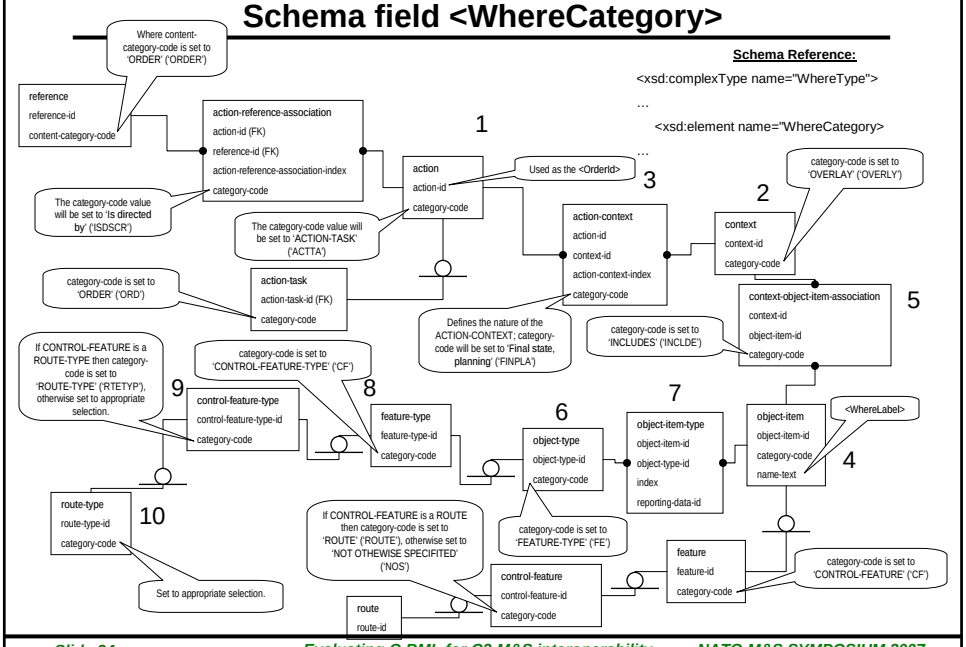

JBML BML Base Service (BBS)

- Purpose:
  - Provide basic building blocks that can be used by multiple domains
    - push/pull API and push/pull Web service
  - who/what-when/where/why/controlMeasures etc.
  - Logical transactions that are “atomic” in the Computer Science sense
    - Must be committed to database all-or-nothing
  - Avoids need to recode the building blocks for every new domain
    - “Where” touches up to 14 tables
    - And requires 373 lines of code, including comments/whitespace
    - To be interoperable, this must be done right – why do it over for every system that is interfaced?

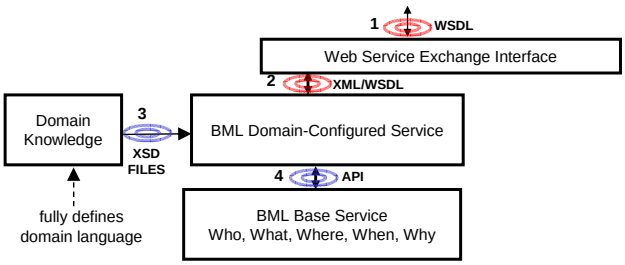
JBML mapping to JC3IEDM
Schema field <Why>



JBML mapping to JC3IEDM
Schema field <WhereCategory>



JBML Service Architecture Top Two Layers



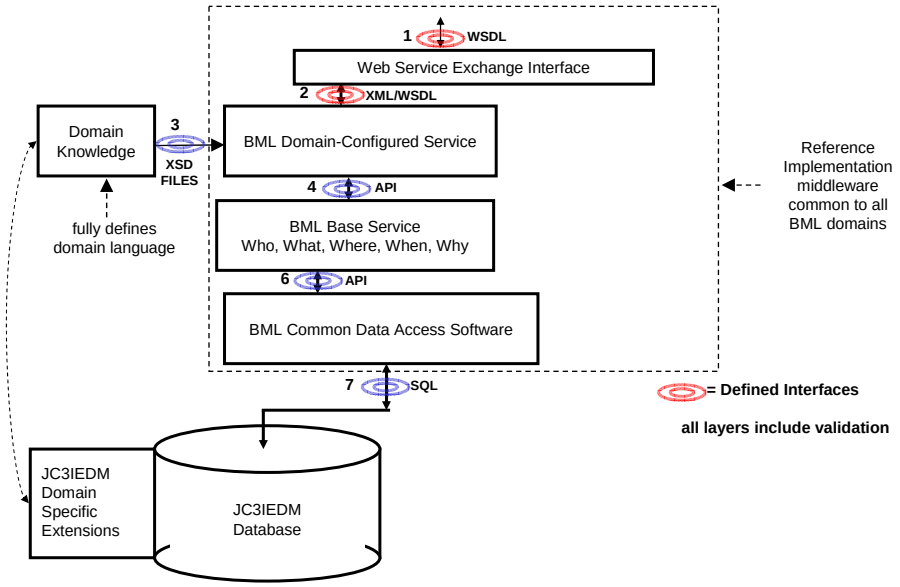
JBML BBS WhyType

```
<xsd:annotation>
    <xsd:documentation>Defines Why a Task is undertaken</xsd:documentation>
</xsd:annotation>
<xsd:simpleType name="WhyTypeEffectDescriptionCode">
    <xsd:restriction base="xsd:string">
            <xsd:enumeration value="DSTRYK"/>
            <xsd:enumeration value="FKIL"/>
            <xsd:enumeration value="IDNT"/>
            <xsd:enumeration value="ILLUMN"/>
            <xsd:enumeration value="INTREC"/>
            <xsd:enumeration value="KILL"/>
            <xsd:enumeration value="LDAM"/>
            <xsd:enumeration value="LGTRST"/>
            <xsd:enumeration value="MKIL"/>
            <xsd:enumeration value="MODDAM"/>
            <xsd:enumeration value="NORSTN"/>
            <xsd:enumeration value="NOS"/>
            <xsd:enumeration value="SDAM"/>
            <xsd:enumeration value="SUPRSD"/>
    </xsd:restriction>
</xsd:simpleType>
```

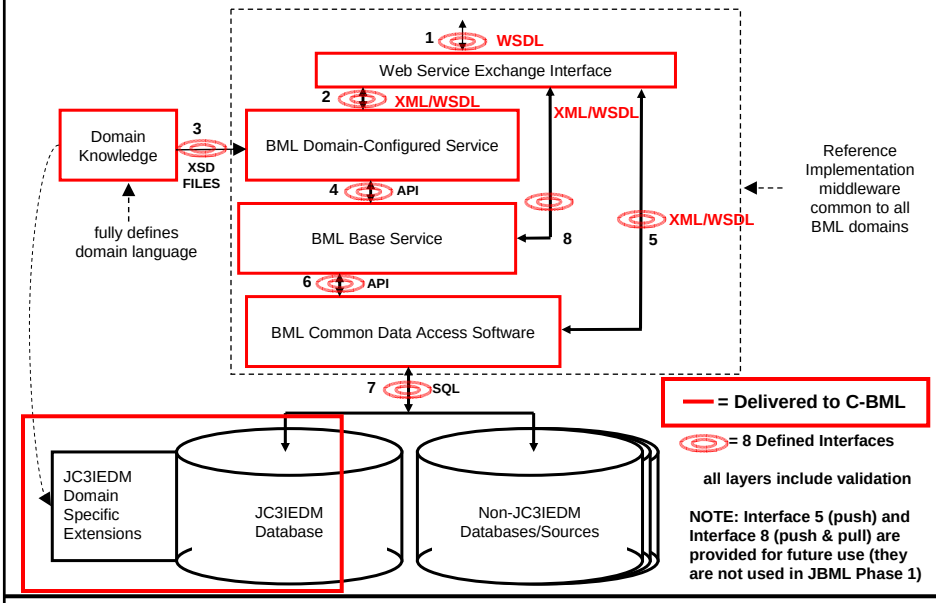
JBML Common Data Access Software (CDAS)

- Purpose:
  - Access the JC3IEDM database (push/pull API and WS)
  - Since the database is SQL-based, this only needs to be a wrapper
    - With validation that only the intended tables are accessed
  - We run this in pull-only mode
  - If it is intended to push BML input directly through JC3IEDM, the CDAS push service needs much better validation of data values
  - We didn't build that sort of service because it lacks the control needed for multiple users to update
    - All tables for a transaction must be updated together with no intervening access by other users
    - We keep this turned off

JBML Service Architecture Three Layers



JBML Service Architecture Future



JBML Web Services Available

Available at http://netlab.gmu.edu/JBML:

- Descriptive documents
  - Architecture overview
    - Domain Configured Service (DCS); BML Base Service (BBS); Common Data Access Service (CDAS)
  - Code documentation (narrative & Javadoc)
  - XSD Web service schema
  - Supporting JC3IEDM mappings
  - SQL database schema
- Open source code
  - Latest version of all Web services
    - CDAS and BBS made possible by open source bootstrap of VMASC Atomic and Composite services
  - GUI to inspect JCDIEDM database using CDAS

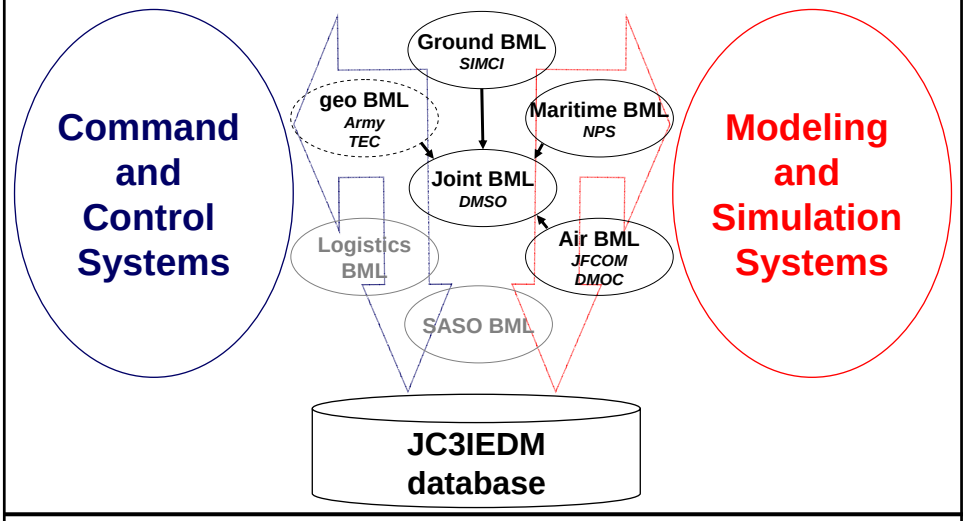
JBML XML Schema Definition (xsd) on JBML website

- DCSOrderPush, DCSOrderPull
  - Top-level definition of Joint order C2 and Simulation services
- OrderTypes
  - Defines Joint operations order consisting of multiple Air, Ground, and Maritime commands
- AirTypes, GroundTypes, MaritimeTypes
  - Define domain-specific information
- FiveWTypes
  - Defines common Who / What-When / Where / Why etc.
- msdlTypes
  - Describes reusable MSDL schema from their webpage
  - This approach implements SISO guidance to make MSDL and C-BML interoperable

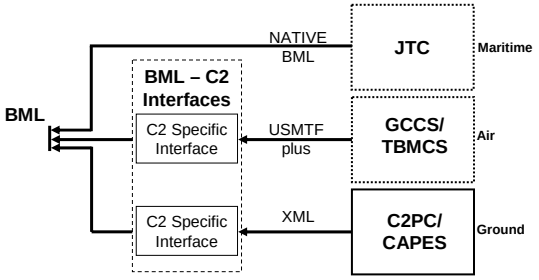
JBML Demonstration 3 May 2007

Joint BML OV-1

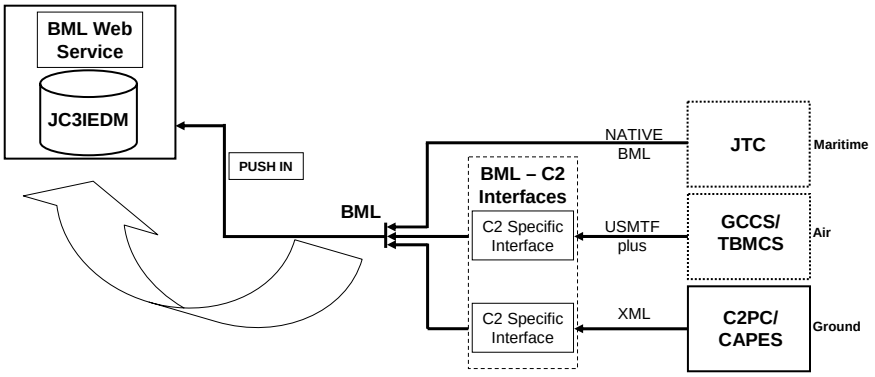
C2 Domain Language(s)



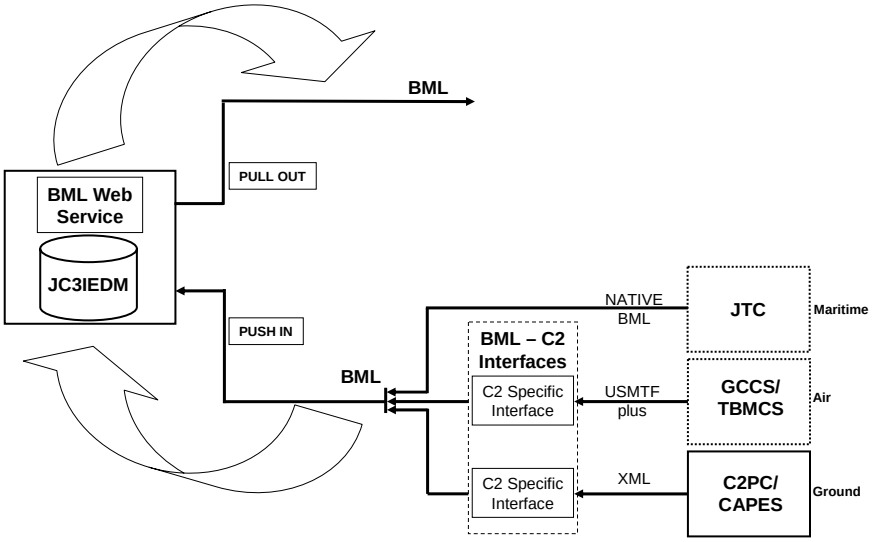
JBML Demo Environment



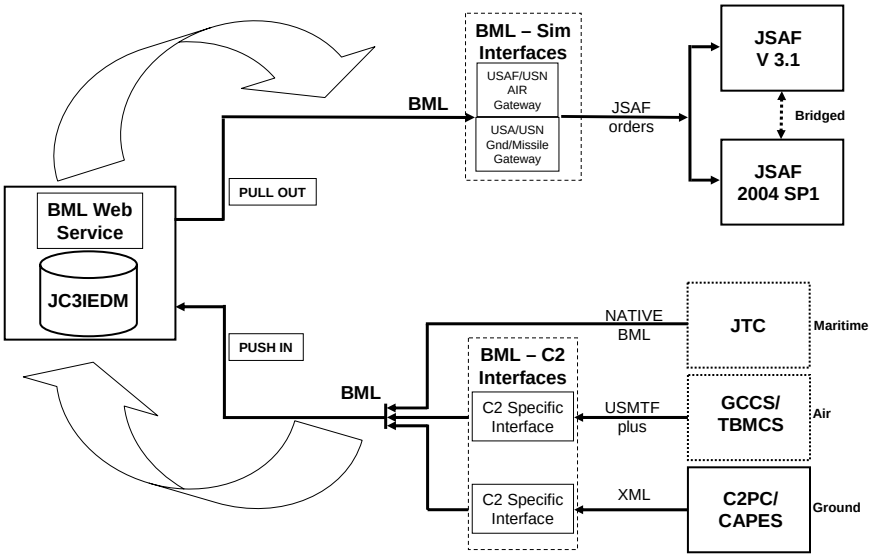
JBML Demo Environment



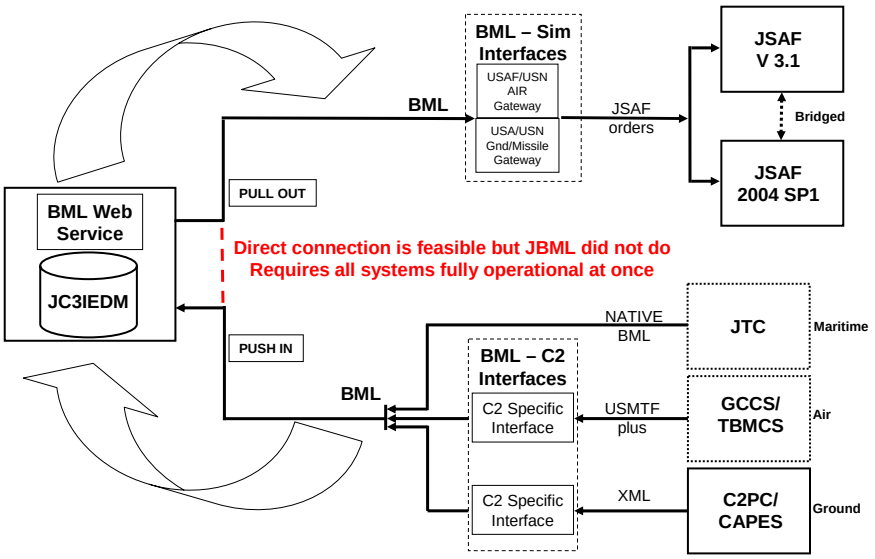
JBML Demo Environment



JBML Demo Environment

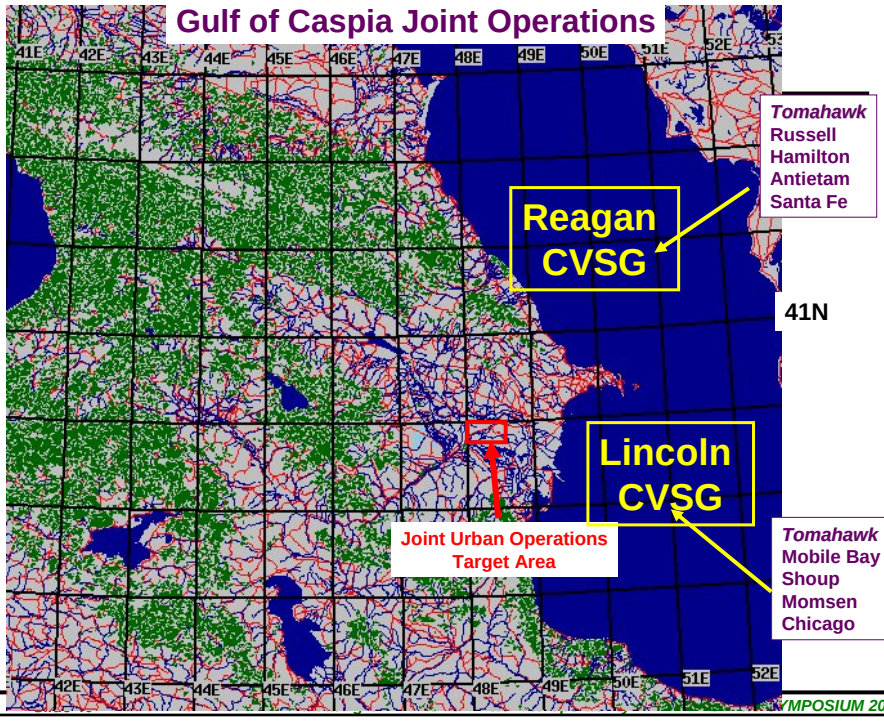


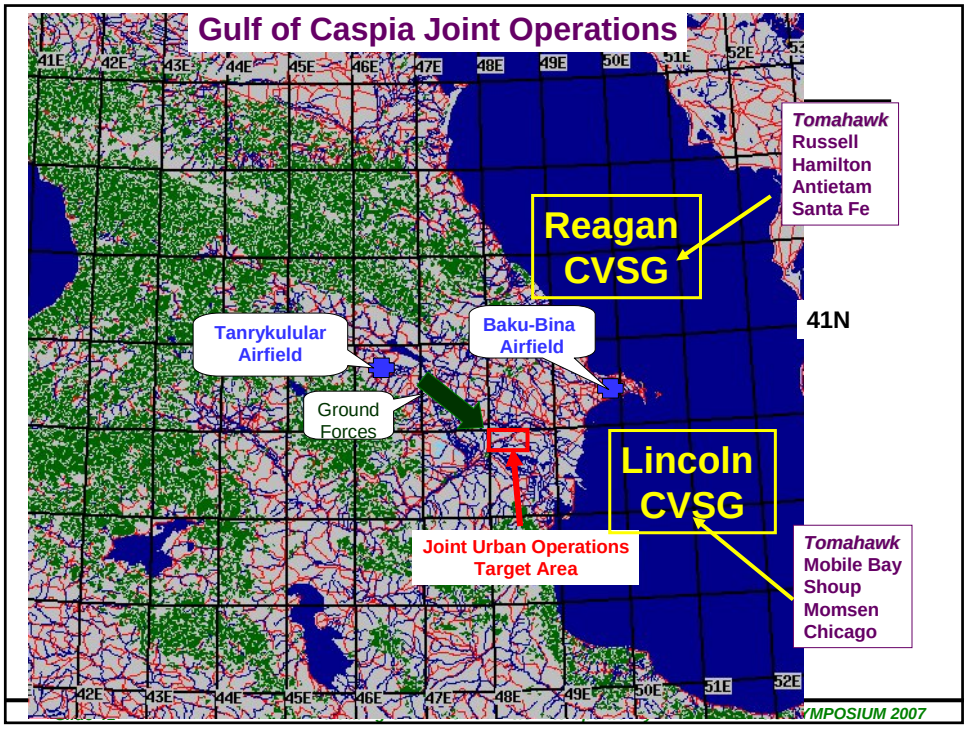
JBML Demo Environment



Demo Concept of the Operation

- **Army** units proceed through **Urban Areas**, seizing designated objectives and destroying enemy forces, in order to reestablish an international border.
- They are preceded by:
  - **Navy** and **Air Force** strikes on key C2 and communication nodes
  - Close Air Support strikes
  - Pre-planned Navy Tomahawk strikes



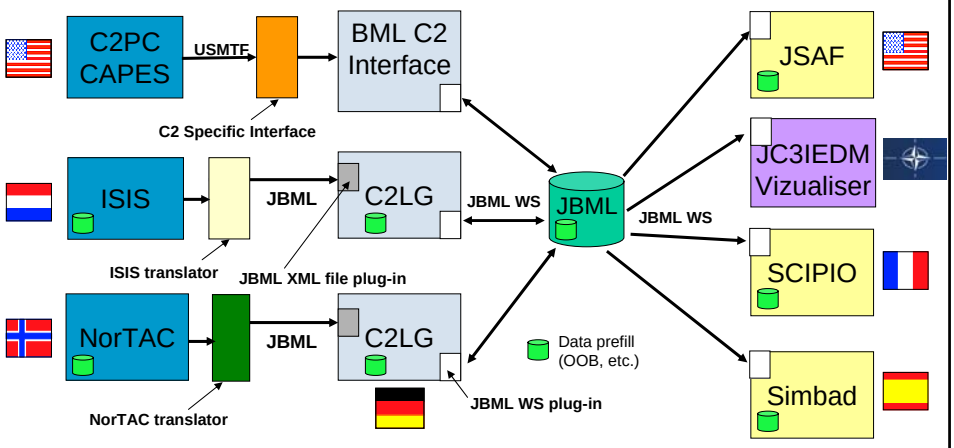


MSG-048 Activities

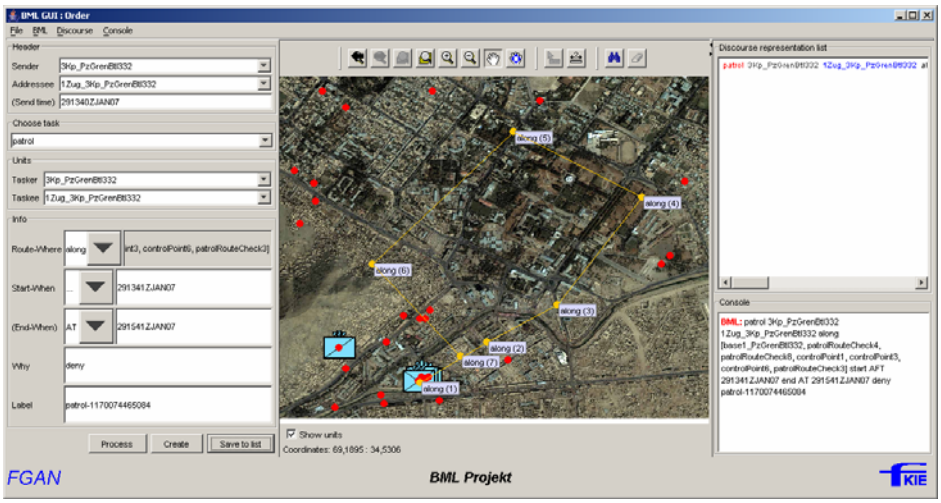
MSG-048 I/ITSEC'07 Demonstration

- First major MSG-048 capability will build on US JBML in a multinational configuration
  - NOR, NLD, DEU ,FRA, DEN, USA, SPA
  - USA: JBML configuration (with small changes) plus VMASC open-source JC3IEDM map viewer
  - NOR: NORTaC C2 System
  - NLD: ISIS C2 System
  - DEU: C2 Lexical Grammar (C2LG) GUI
  - FRA: SCIPIO simulation
  - SPA: SIMBAD simulation
- MSCO strong support to help advertising the C-BML new approach

MSG-048 I/ITSEC'07 Configuration



C2LG GUI



MSG-048 I/ITSEC'07 Scenario

- Six demonstration sessions will highlight different parts of the scenario and architecture
  - Briefers from participating nations
- Use US Caspian Sea terrain
- Context: former colony (failed state)
- NATO RTF providing security assistance
- Mission: secure airport
  - US secures bridges providing access to airfield
  - NO/NL forces secure the airfield

MSG-048 Future Plans

- Improve the current architecture and scenario with other National C2 or simulation systems
  - GBR
    - ALPS: Simulation of Army-focused C2 application; MIP-compliant
    - JSAF: Customized for UK equipment, units, behaviors
    - Implement vignette from I/ITSEC demo
    - Longer term: Operational C2 systems (*e.g.* Bowman/ComBAT and JADOCS) and Survey of UK requirements for C-BML
  - FRA: introduce C2 system SICF
  - DEN: adapt Sitaware as the commander C2 system
  - USA: improved geospatial and situational awareness
- Implement spot and intel reports
- Develop a new experiment next year

Summary/Conclusions

JBML Contributions to C-BML

- Service architecture and open source Web services provide a regular and extensible framework upon which a powerful, flexible and growing family of standards can be created
- Defined building blocks who/what/when/where/why of the BBS layer are candidates for the Phase 1 standard
- JBML demonstration showed “it can be done” for interoperable Joint C2-Simulation
  - Inspired similar activity for Coalition BML

MSG-048 Future Activity

- The I/ITSEC’07 demonstration is the first step toward C2-Simulation experiments in planning for 2008
- Initial experimentation will take place via Internet
  - Consider a range of activities enabled by national C2 and simulation capabilities
  - Incorporate expected advances such as GeoBML and Situation Report feedback from simulations
- Hope to bring the whole capability to MSG Symposium 2008
- Potential to support a NATO exercise or experiment