Modeling and Simulation (M&S)  
Phase Two  
Next Generation M&S Architecture  

Michael Egnor  
DD J7 Synchronization & Integration  
JOSE Environment Development  
15 Sept 2011
Purpose

• Information brief to describe the Next Generation Modeling and Simulation Architecture
Challenges with Current M&S Approach

• Federation becoming increasingly large and unmanageable
• Costs increasing to upgrade existing and new federate additions
• Resource-intensive integration of new federates and support
• Tightly-coupled simulations
• Bridges and gateways often required to integrate new and emerging technologies
• Many redundancies and inconsistencies in data representation
Next-Generation M&S Capability

• Rapidly utilize real-world source data
  – Cloud-based Rapid Data Generation (RDG) and distribution capability
  – Support operational COA development
  – Support Concept Development with Training

• Make simulations smarter / more capable
  – Navigate terrain / interpret physical environment and effects
  – Attempt to apply orders more effectively
  – Allow faster than real-time simulation without intervention
  – Dynamic Intelligent Control Agent (scalable in cloud)
  – Reduce role-player dependency

• Abstract User Interfaces to lower the barrier to use
  – Invoke only services needed by user
  – On demand access to scenario repositories
  – Users interact with common terms and symbols (Military & Civilian systems)
  – Interact with C2 systems
Fundamental Precepts

• Open Architecture Principles
• Yearly deliveries through JLVC increments with the ability to roll back to a previous year’s increment
• Composable simulations
• Central Management
• Synergy and Efficiencies Gained By Aligning Current Efforts
  – Service Oriented Architecture (SOA) Prototype (JCW Joint Simulations)
  – KORCOM project
  – Rapid Data Generation (RDG) project
  – NATO Training Federation (NTF) Objectives (MSG-105)
• JCW in collaboration with the Service stakeholders define the architecture and standards associated with the single constructive architecture

• JCW in collaboration with the Service and COCOM stakeholders lead the Serviced Oriented Architecture governance body

• JCW within resources constraints incentivizes the Services to modify their flagship simulation to adopt federation provide services
The Future of M&S -- Service Oriented Architecture (SOA)

Design to support Joint Training and Experimentation for the Joint and Coalition Warfighter

Seamless Integration
Automatic code generation based on metadata

Maximize Composability & Reusability

Design/Prototype

Phase 1: FY12 – 13 JLVC 6 & 7
HYBRID SOA ARCHITECTURE
- HLA IEEE 1516-2010 (RTI SELECTION)
- NEW MODULAR FED OBJ MODEL (FOM)
- SOA XML/WEB SERVICES
- INITIATE SOA GOVERNANCE
- INITIAL DECOUPLING OF DATA / APPS
- DEVELOP SOA SERVICES
- TEST & EVALUATE

Phase 2: FY14 – 15 JLVC 8 & 9
COMPOSABLE SOA
- ESTABLISH SOA GOVERNANCE
- RAPID DATA GEN (RDG)
- COMPOSABLE SIMS & MODELS
- DEVELOP SOA SERVICES
- WEB ONTOLOGY
- DISTRIBUTION METHODS
- ANALYZE AND REFINE SOA

Phase 3: FY16 – 17 JLVC 10
COMPOSABLE JOINT ENVIRONMENT FRAMEWORK
- SERVICE BASED M&S ARCHITECTURE
- INTELLIGENT SERVICES
- COMPOSABLE SIMS / MODELS
- SIMULTANEOUS, MULTIPLE ON-DEMAND CAP
- IMMERSIVE TRAINING
- ADAPTIVE LEARNING

CURRENT STATE: TIGHTLY COUPLED RTI; COMPLEX; EXPENSIVE INTEGRATION & EXECUTION
# M&S Future – Service Oriented Architecture (SOA) Implementation

## PHASE 1
**HYBRID SOA ARCHITECTURE JLVC 6 & 7**
- Joint Theater Level Simulation
- Joint Conflict and Tactical Simulation
- JCATS Low Overhead Driver
- Joint Exercise Control Station (JECS)

## PHASE 2
**COMPOSABLE MODELS JLVC 8 & 9**
- Proprietary and Sensitive Information
- Synergy and Efficiencies Gained Through Aligning SOA Prototype, KORCOM Project, Rapid Data Generation (RDG) project & NATO Training Federation (NTF) Objectives (MSG-105)

## PHASE 3
**COMPOSABLE JOINT ENVIRONMENT JLVC 10**
- Autonomous Intelligence Services
- JLVC SOA Framework
- New Services Added

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Initiate SOA Governance</td>
<td>Decompose M&amp;S HLA 1516-2010</td>
<td>Virtual Services</td>
<td>Cloud Computing</td>
<td>KORCOM IOC</td>
<td>JLVC FOC</td>
</tr>
<tr>
<td>RTI &amp; FOM</td>
<td>*</td>
<td>Virtual Interfaces</td>
<td>Virtual Interfaces</td>
<td>USAF Service Model</td>
<td>Live Interfaces</td>
</tr>
<tr>
<td>*</td>
<td>*</td>
<td>USAF NATO NTM V2.0</td>
<td>Composable SOA Model</td>
<td>Live Interaces</td>
<td>Rapid Data Generation (RDG)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>USAF Service Model</td>
<td>USA Service Methodology</td>
<td>USA Service Model</td>
<td>Autonomous Intel Services</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>USA Service Methodology</td>
<td>USA Service Model</td>
<td>KORCOM IOC</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Live Interaces</td>
<td>FOC</td>
</tr>
</tbody>
</table>

### Test, Evaluate & Refine
- Dynamic Services Creation
- New Services Added
- Test, Evaluate & Refine
Joint Staff
Joint & Coalition Warfighting
M&S - Future

- Integrated training and experimentation
- Open Architecture
- Service oriented architecture
- Full Interagency & Coalition integration
- Learn from results of simulation
- Seamless multi-level training
- Rapid scenario generation
- Cost effective business model
- Routinely accessible from anywhere

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>JLYC 5.0 UBMIC integrated</td>
<td>JTYN in the National Capital Region</td>
<td>Persistent Coalition training network</td>
<td>Simulation for Course of Action Development</td>
<td>On Demand Multi-Level Training (Integrated Gaming)</td>
<td>Wireless Simulation Distribution</td>
<td>DOTMLPF Recommendations</td>
<td>Prioritized Joint Needs &amp; Capabilities</td>
</tr>
</tbody>
</table>

Joint Operational Systems Environment (JOSE)
Questions?
BACKUP
JLVC Project Timeline (IOC)

Joint Activity
- JLVC Version 4.0
- JLVC Version 5.0
- JLVC Version 6.0
- JLVC Modular FOM Design
- JLVC Version 7.0 (1516 Evolved)
- JLVC Aggregate Combat Service (JLCCCTC ENG)
- JLVC-MRF Bridge (LANL)
- Terrain Services Prototype (Cloud Computing)
- Dynamic JELC - Process Model Experiments
- RDG Prototype
- Joint Model Web-Service Experiments (1516 HLA Evolved)
- Joint Model Development - Simulation of Multiple Timescales

Prototyping Activity
- MTWS JLVC Integration
- MTWS Data Alignment
- MTWS 1516 Dev
- USMC IOC
- USMC FOC
- MRF-W Implementation
- WARSIM Scalability Enhancements
- JLCCCTC JLVC Data Alignment
- AFMSTT Data Alignment
- AFMSTT 1516 Implementation
- Releasable JSAF-REL ROK/NATO
- JSAF KBSC Delivery Early look
- JLVC-NCTB Data alignment
- JSAF Naval Mines
- NATO/Coalition HLA specification design (1516)
- Coalition HLA Sustainment
- Cross Domain Guard & HLA Rule Set (1516)
- JLVC KSIMS Integration (JLVC-MRF Bridge)

Coalition Activity
- KORCOM IOC software delivered Sep 13 after UFG 13
- MRF-W FOC UFG13

Baseline

Key
- ▲ Release/Delivery
- ★ Milestone
- ◈ Decision Point
- □ NET/Fielding
JLVC Project Timeline (FOC)

FY 14

Q1: Oct – Dec
Q2: Jan - Mar
Q3: Apr - Jun
Q4: Jul - Sep

FY 15

Q1: Oct – Dec
Q2: Jan - Mar
Q3: Apr - Jun
Q4: Jul - Sep

FY 16

Q1: Oct – Dec
Q2: Jan - Mar
Q3: Apr - Jun
Q4: Jul - Sep

FY 17

Q1: Oct – Dec
Q2: Jan - Mar
Q3: Apr - Jun
Q4: Jul - Sep

Joint Activity

Version 8.0

JLVC Version 9.0

JLVC Version 10.0

JLVC-MRF Bridge Modified for final KSIMS Architecture

JLVC Continued Web-services Development

Service Activity

AFMSTT Bridge

WARSIM JLVC Integration

MRF-W Capability Sustainment

Service Activity

JSAF/NCTE KBSC Enhancements

Coalition Activity

Coalition HLA Sustainment

KSIMS Led Federation Design

Baseline

Key

Release/Delivery

Milestone

Decision Point

NET/Fielding

UNCLASSIFIED
Enhancements (JLVC Version 6.0—Not delivered to KORCOM)

- WARSIM/WIM XML Ingestion/FED ID to MRF
- JLCCTC JLVC Data Alignment and Configuration Management
- WIM Integration with JLVC—Classified/Standalone
- Consistent Logistics Play (Convoys, Supply, Med)
- Consistent Fixed Targeting Play
- ROK/NATO Releasable JSAF
- NCTE & AFMSTT Data Alignment with JLVC and Configuration Management
- JLVC design modular 1516 HLA Evolved Architecture
- Joint Models, JBUS, and GIAC 1516 Evolved Experiment
**Enhancements (Delivered in JLVC Version 7.0-Sep 2013)**

- Configuration Management of WARSIM/WIM/ONESAF Data
- Implement scenario XML Ingestion for KSIMS products
- Production JLVC-MRF Bridge and Federation Agreements
- AFMSTT connect to MRF using JLVC Bridge solution IOC
- JLVC design modular 1516 Architecture implemented
- Cross Domain Guard and HLA Rule Set implemented
- JSAF Mine Enhancements
- MTWS & AFMSTT Implement 1516 HLA Evolved
Enhancements (Delivered in JLVC Version 8.0) – Reduced R&D Funding

- WARSIM-JLVC Bridge Development (3 year effort)
- JDLM and remaining HLA Constructive federates implement 1516 Evolved
- KSIMS data alignment
- Design for new KSIMS Architecture & Federation Agreements
- AFMSTT connect to MRF using JLVC Bridge solution FOC
- Production Aggregate Combat Service
- Enhanced AAR Collection
- JSAF KBSC Requested User Enhancements
- Prototype Sensor Services
Enhancements (Delivered in JLVC Version 10.0)

- Implement Korean Hierarchical Federation
- Modified JLVC Bridge to support Korean Hier FOM
- KBSC Requested Mods
- Enhanced Interoperability with Korean Models
- WARSIM-JLVC Bridge
- MRF-W Option for Time Managed Policies
• **Dynamic Link Compatibility** – That means federates can switch which RTI they use without recompiling/relinking their application.

• **Modular FOMs** – Modular FOMs allow federation developers to break up their object model into useful parts (called FOM Modules). Then each federate only needs to know about the FOM Modules it uses.

• **Update Rate Reduction** – Allows federates to tell the RTI they can only handle data updates below a certain rate. This allows update rate constrained federates to participate in busy federations without bogging down.

• **Better Fault Tolerance** – HLA now has a mechanism for notifying Federates when another Federate becomes disconnected from the network. It means when something goes wrong, everyone will understand what went wrong quickly.

• **WEB Services API (WSDL)** - The HLA Standard now defines a Web Service Description Language (WSDL) “binding”. This is similar to the C++, or Java bindings, but for the Web.
  - Based on the WSDL code, it is possible to generate code in different programming languages. It is transparent to a federation whether a specific federate uses the C++, Java or WSDL API.
  - Major benefits of the WSDL API include the support for numerous newer and older languages as well as the ease of deployment across wide area networks.
  - The major long term benefit of the WSDL API is that it promotes and supports the concept of simulations as readily available services provided within and between enterprises.