# Composing a Joint Federation Object Model

*Andy Bowers*
The MITRE Corporation
USJFCOM J7, JTS Division
Suffolk, VA 23435
757.673.5718
bowersa@mitre.org

*Dannie Cutts*
AEgis Technologies Group Inc.
USJFCOM J7, JTS Division
Suffolk, VA 23435
256.922.0802 x2209
dcutts@aegistg.com

Keywords:

HLA, Object Model, FOM, BOM, Base Object Model, TENA, LROM

**ABSTRACT:** *As simulation users adopted the High Level Architecture (HLA) to promote interoperability, composability, and reuseability, Federation Object Model (FOM) development and use necessarily grew apace. HLA federations have in many cases delivered on these promised "ilities" yet a simulation fortunate enough to be a member of multiple federations often does not realize these same benefits. Membership in multiple federations requires that the individual federate interoperate with multiple FOMs. This in turn usually equates to the federate developing multiple interfaces with limited opportunity for reuse. The Modeling and Simulation (M&S) Community has recognized this issue and sought its redress through composable object model approaches such as the Base Object Model (BOM) technology. This paper reports on work accomplished under the auspices of United States Joint Forces Command (USJFCOM) to decompose the FOMs used by the Joint Warfighting Center (JWFC), identify and eliminate redundant elements, and develop a composite Joint FOM. The effort is intended as a "proof-of-principle" on the basis of which USJFCOM might solicit broader community support in developing an object model library and process for composing FOMs for use by the Joint and Multinational M&S community.*

## 1 Introduction

Interoperability has been at the heart of the High Level Architecture (HLA) since its inception. The authors believe that the failure of the HLA to achieve the level of interoperability originally envisioned has largely been with the Federation Object Model (FOM). HLA specifies the format for documenting object models but for reasons of flexibility does not specify the contents of object models. Prior experiences with fixed object model representations such as DIS, led the developers of HLA to choose to allow federation developers to develop their own object model representations for their particular needs. Many times these object models were developed with little or no regard to consistency across object models. The end result has been a lack of interoperability across federations.

The consequences of disparately developed federation object models are semantic mismatches between federations and significant duplication of effort for any one federate participating in multiple federations.

This paper describes ongoing work at JFCOM to develop best practices in support of M&S reuse through object model interoperability and composability. The application of Base Object Modeling methodologies to this problem is discussed.

## 2 Multiple FOM Interoperability

Over the past decade, multiple joint federations have been developed with object models designed to meet the unique needs of each federation with little or no coordination between federation developers. The end result for Joint Forces Command (JFCOM) has been divergent object models. As a consequence, simulation developers wishing to participate in multiple federations must expend limited resources to modify their simulations to work with different object models, or depend on gateway translators to bridge the differences. To cite just one example, the Joint Conflict and Tactical Simulation (JCATS) participates in five different federations, only two of which use the same FOM. Hence the JCATS developer uses four different gateways to exchange data with the RunTime Infrastructure (RTI).

A number of mapping strategies such as gateways and agile FOM interfaces have been employed to reduce the effort of moving from one FOM to another. (Yao, 2006), (Granowetter, 2005), and (Cutts, 2007) provide additional insight into the use of gateways and agile interfaces, and their impact on interoperability and reuse.

# 3 Composable Object Models

(Davis, 2003) describes composability as "The ability to select and assemble components in various combinations to satisfy specific user requirements meaningfully. A defining characteristic of composability is the ability to combine and recombine components into different systems for different purposes." Thus reuse is feasible only when the assets to be reused are interoperable and composable.

As pointed out earlier, interoperability, reusability and composability of M&S assets is at the heart of the HLA with the FOM at the foundation of HLA Interoperability. The HLA Object Model Template (OMT) specification defines an inheritance strategy for representing objects. While inheritance is a powerful method for representing generalization / specialization relationships, it is inadequate for representing large complex systems and the "part-of" relationships between components of those systems. The authors feel that in addition to inheritance, the ability to compose object models would offer an improvement in representational capabilities. This project is focusing on composability of HLA Object Models to allow federations to share "common" object model components but support the flexibility of including unique object model components to meet the particular requirements of the individual federations. The initial task is focused on Joint Federation Object Models (JFOM), but the longer term intent is to analyze additional FOMs as well as the Test and Training Enabling Architecture (TENA) Joint Logical Range Object Model (LROM). The project is making extensive use of the Base Object Model (BOM) approach (SISO, 2006) and the BOMworks™ tool from SimVentions. In addition, there is an effort within the SISO to develop a standard for Modular FOMs. Although we are not incorporating modular FOM work into this project, (Moeller, 2007) has shown that modular FOMs complement the BOM concept. A brief discussion of each follows.

## 3.1 Goals for Object Model Composability

There are a number of "objectives" or goals that support object model composability: (Davis, 2003)

### 3.1.1 Use of Standards

Development teams must use accepted standards, tools, and methodologies to represent object model components in order to achieve long-term benefits. Moreover, teams should employ tools and methodologies common across the M&S community or across the commercial marketplace. Common tools and methodologies allow teams to develop and share object model components. The BOM standard offers such a standard.

### 3.1.2 Conceptual Modeling

A conceptual model provides a description of "what" the object model component represents independent of the object model implementation. Moreover, it can capture assumptions or limitations of those abstractions. It also provides other information to assist users in understanding the model in an implementation-independent manner. Conceptual models are critical to object model composition. BOMs provide support for conceptual modeling. Figure 3.3-1 shows the major components of a BOM and illustrate how the Conceptual Model fits into the BOM structure.

### 3.1.3 M&S Ontology / Lexicon

An established lexicon and ontology for M&S object models supports a shared understanding of the object models. These efforts facilitate both interoperability and composability of object models.

### 3.1.4 Object Model Repositories / Directories

As object model components emerge into common usage, the community needs to develop a standard approach to storing components so federation developers can easily locate and access them. In addition to the object model, the repository will also hold and/or the directory must provide access to the metadata, conceptual model, and use cases for the object model. This data should include V&V findings for the object model and use histories. As shown in Figure 3.3-1, BOMs support metadata and use case documentation.

### 3.1.5 Reduced Maintenance Costs

There will be additional start-up costs to define and document object model components. However, in the longer term, cost savings, cost avoidance, and increased responsiveness will offset the initial expenses. Costs can be minimized in the near term if initial efforts implement short-cycle, spiral development approaches with in-phase validation of modular segments focused on critical, mission-oriented capabilities. Feasible savings accrue because the long-term costs of maintaining the object models should be significantly less than the expense of maintaining multiple versions of federates and/or gateways for multiple federations.

### 3.1.6 Reduced Duplication of Effort

Currently, each federation maintains its own unique object model. The goal is to identify the common object model components between the federations and allow the federations to focus on those object model components unique to their needs.

3.1.7 Improved Interoperability & Reuse

The use of common object models with well-understood semantics will support a higher level of interoperability and reuse. As sets of common object models emerge, the community will capture their semantics. This meta-data facilitates a common understanding of the use of the object model.

3.1.8 Object Model Convergence

Over the long term, the community should strive toward convergence of object modeling across architectures, protocols, and standards. In the Joint arena, these would include HLA, TENA, DIS, Joint Command, Control, and Consultation Information Exchange Data Model (JC3IEDM), and Battle Management Language (BML). While initial efforts are focused on developing mission-critical capabilities, the intent is to reconcile Joint federation object models as a step toward a longer-term goal of improved interoperability and reuse across the Joint M&S community and beyond.

## 3.2 Base Object Model (BOM) as a unifying approach to object modeling

The BOM concept provides a flexible component approach that, based upon our analysis and experience, can be applied for resolving the issue of divergent HLA object models. It is an ideal candidate because it is specifically intended to encourage "composability". A BOM is defined as "a piece part of a conceptual model, simulation object model, or federation object model, which can be used as a building block in the development and/or extension of a simulation or federation." (BOM Template, 2006)

The modularity offered by BOMs, provides a critical step toward improved interoperability and reuse. According to (Davis, 2003), "Modularity is necessary when dealing with complex systems, and some degree of composability is surely possible and desirable." He shares that creating a simulation requires breaking the problem into parts that can be addressed separately. In our case those parts can be codified using BOMs. As illustrated in Figure 3.3-1, a BOM can be made up of four major structure elements: Model Identification, Conceptual Model Definition, Model Mapping and the underlying Object Model Definition. (BOM Template, 2006), provides a more comprehensive examination of Base Object Models while (Cutts, 2007) discusses their use in object model reconciliation.

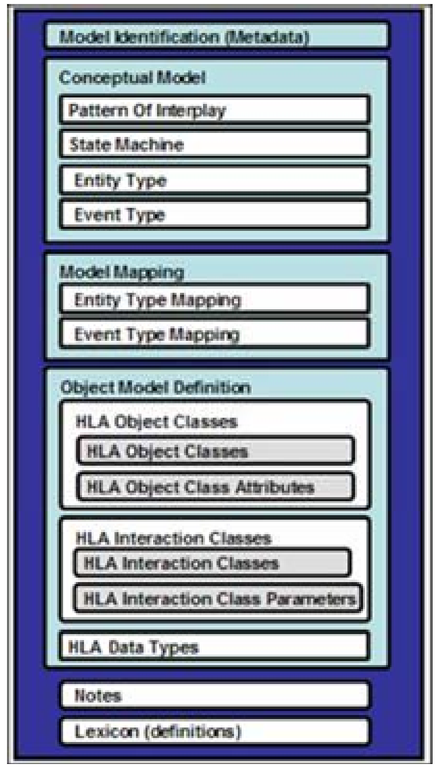


Figure 3.3-1 – BOM Elements

The Model Identification element identified in Figure 3.3-1 is used for providing the essential metadata for documenting a BOM. Figure 3.3-2 provides a view of the metadata attributes found within the Model Identification structure.

A primary purpose of the metadata is to support ease of archiving, browsing, discovery and improved understanding of object models & components.

Figure 3.3-1 identifies the Conceptual Model as the next element comprising the BOM. This element is further described in Figure 3.3-3 by identifying the elements of the conceptual model as well as the relationships between those elements. Conceptual model information is often the sparsest information available for a model and yet, can be the most useful for understanding the model's purpose. By enabling conceptual model documentation, the BOM enables effective use of that model for different federations and different architectures (e.g. HLA, TENA).

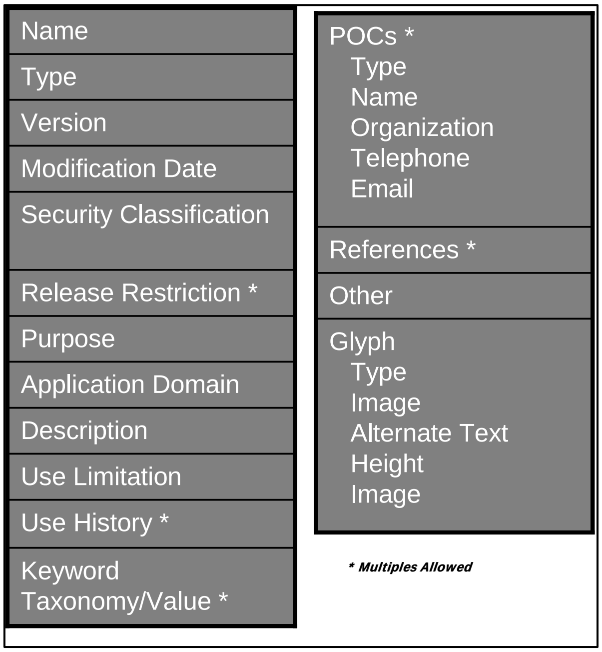


Figure 3.3-2 – BOM Metadata Elements

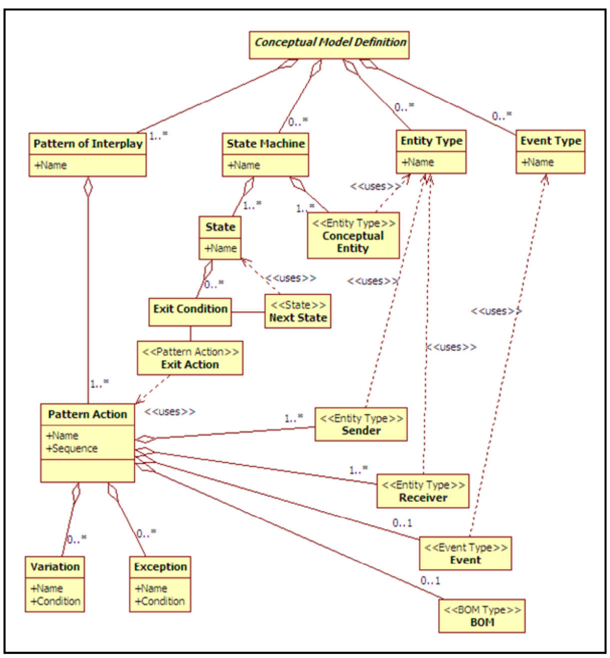


Figure 3.3-3 – BOM Conceptual Model Elements

The Object Model Definition element identified in Figure 3.3-1 is used to identify the core class structures intended to be represented by the system, simulation or model. While these elements of the BOM specification are borrowed from the HLA Object Model Template (OMT), it is important to note that this aspect of a BOM is not limited to HLA. The HLA OMT merely provides a common mechanism for describing classes that are understood by the wider M&S community.

The Model Mapping element identified in Figure 3.3-1 may be defined in one BOM and provide linkage to other BOMs. For instance, the Conceptual Model Definition and Model Mapping might exist in one BOM, whereas the specific class structure that can be used to support the Entity Types and Event Types defined in the Conceptual Model Definition may exist in one or more other BOMs.

This ability for BOMs to be loosely coupled, allowing entity types to link externally with specific class structures, is only one aspect of composability offered by the BOM. It is also possible to take a collection of BOMs that describe various patterns of interplay and state machines that are to be exhibited and aggregate them together to constitute a federation object model. This capability is illustrated in Figure 3.3-4. The capability to collect and stitch BOMs in this fashion and produce a BOM Assembly provides a useful mechanism for supporting multiple architectures such as HLA or TENA. In fact, as evidenced by Chase and Gustavson in their paper "From FOMs to BOMs and Back Again" (Chase, 2006), existing object models can be decomposed into more reusable object models, which are defined as BOMs, and then coupled to reformulate the capabilities that were initially offered in the original FOM prior to its decomposition. One advantage of this modular approach is that an individual BOM could be changed or swapped with another, without requiring a major editing change to the entire FOM or LROM and with minimal code impact to a system or simulation that uses such a FOM or LROM.

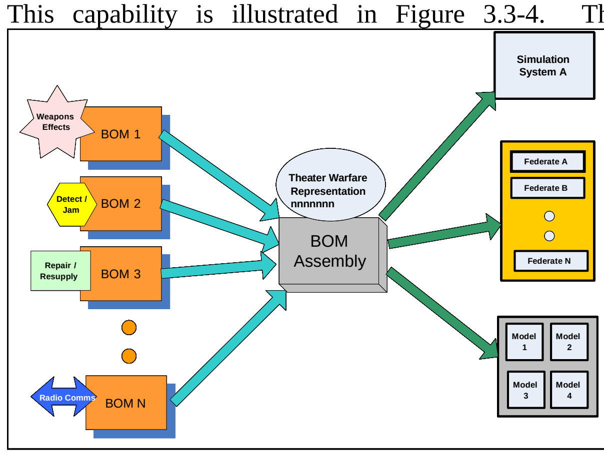


Figure 3.3-4 Composability through BOM Assembly

A well-defined BOM can be used within and across the multiple federations and architectures such as HLA and TENA. Consider the decomposition, reconstitution, and modular exchange capability offered by BOMs, and the ability for a BOM Assembly to serve up compatible HLA FOMs and TENA LROMs. Based on these capabilities, it is sufficient to say that BOMs provide an enabling capability for supporting improved interoperability between and reuse of object models.

### 3.3 BOMworks™ tool

We are using the BOMworks tool from SimVentions to decompose HLA FOMs, generate BOMs corresponding to HLA FOM classes and assemble those BOMs. The tool is available from the SimVentions website free of charge and supports the decomposition of HLA FOMs as well as the building and assembly of BOMs.

### 3.4 SISO Modular FOM Standard

Work is underway within the SISO to define a standard for modular FOMs. As the name implies, the effort promotes FOM decomposition into modules to obtain many of the same advantages as those sited in section 3.1. As previously mentioned, (Moeller 2007) anticipates synergy between BOMs and modular FOMs with the former documenting conceptual models while the latter addresses modules necessary to a FOM instance, e.g. federation management issues (synchronization, for example), Management Object Model use, etc. One might also envision groups of oft-used BOMs comprising a FOM module.

## 4 Current Work

Work to date has focused on decomposing and analyzing the major JFCOM Federation Object Models. We are performing an initial analysis on the Joint Live Virtual Constructive (JLVC) Federation and the Joint Multi-Resolution Model (JMRM) Federation Object Models. The decomposition and analysis process is discussed in the following sections. An example of a comparison follows that discussion.

### 4.1 Decomposition

The BOMWorks™ tool from SimVentions was used to decompose the HLA Object Models and produce BOMs corresponding to individual classes in each of the FOMs. The individual BOMs form the basis for analysis. Although the tool automates the extraction of classes from the FOM, the overall process is still largely manual. Each class in the FOM had to be manually selected and a BOM generated corresponding to each HLA Class. Each BOM corresponds to a stand-alone class in the HLA FOM, that is, with no inherited attributes. This class-by-class decomposition allows object classes to be analyzed and compared at the "atomic" level. Future work will define assemblies to compose these base-level BOMs into higher order BOMs representing entities and actions within the simulation space.

### 4.2 Analysis

After decomposition, the next step was to perform an analysis of the decomposed object models. This phase is ongoing. Analysis remains a manual process with the objective being to define "Measures of Similarity" between classes. There are several potential aspects of analysis that should be considered:

- Morphological Analysis- An understanding of word forms (e.g. understanding that Aircraft, Air_Vehicle, and UAV are related)
- Grammatical Analysis – An understanding of the parts of speech (e.g. The use of *Target* as a verb in an operations order vs. *Target* as a noun pertaining to something being targeted).
- Semantic Analysis – An understanding of the semantics behind the use of a class or attribute. That is, the purpose of a class or attribute and how it is used in a federation. (Tolk, 2003) discusses Levels of Conceptual Interoperability Models (LCIM).

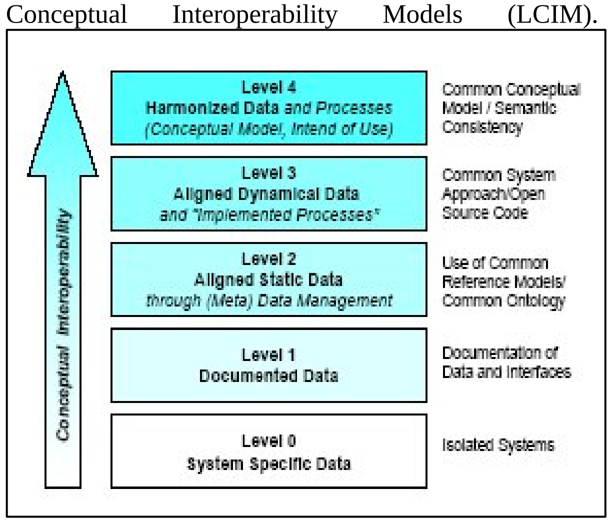


Figure 4.2-1 – Levels of Conceptual Interoperability

Semantic Analysis would result in a Level 4 conceptual interoperability level. The current level of conceptual interoperability between the existing FOMs is around a Level 1 (documented data). An important ingredient of semantic analysis is to build conceptual models for each of the object model components. Figure 4.2-1 depicts the four levels of the LCIM.

The analysis strategy used on this project for "Measures of Similarity is as follows:

- **Class name similarity:** If two classes have the same (or nearly the same) name, an analysis was performed to determine if they represented the same thing in the simulation space
- **Attribute name similarity:** If two attributes have the same (or nearly the same) name, an analysis was performed to determine if they represented the same characteristic of a class
- **Semantic/Usage similarity:** If two classes are used the same way in a federation, an analysis was performed to determine if they are functionally the same or similar.

### 4.3 Example

(BOM Guidance, 2006) documents two approaches for developing BOMs. One approach addresses building a BOM from scratch while the second describes reusing parts of existing object models. The latter approach was adopted to provide two examples of work completed to date. This section first compares object class representation of aircraft in the JMRM and JLVC FOMs. Aircraft are represented, in some form, in many object models. For example, service FOMs, including both the Army Constructive Training Federation (ACTF) and the Marine Corps Federation (MCFED) FOMs, include object class(es) representing aircraft. By starting with a common object we 1. reinforce the assertion that a BOM library has community applicability, 2. demonstrate reuse promised by the BOM literature and 3. encourage subsequent reuse of BOM representations described herein. Different interactions with the same purpose from the JMRM and JLVC FOMs are then described with intent to illustrate how adapting one might serve the purpose of both while improving interoperability.

Aircraft Example. In JLVC the object class describing aircraft is termed 'Aircraft.' In JMRM there are two object classes, Air_Mission and Aircraft. The JMRM Aircraft object class was copied from the JLVC FOM together with its super classes up to and including the BaseEntity class. This was done to promote some measure of interoperability when entity representation was added to the JMRM FOM. The addition of these classes was not, however, accompanied by a concurrent simplification of the existing Air_Mission class. Thus there is some duplication that exists within the JMRM FOM itself. Comparison of the JMRM Air_Mission and JLVC Aircraft classes will identify duplicate attributes and provide a basis for simplification. In both federations these classes inherit attributes from other classes. Analysis of all relevant object classes results in the discovery of few attributes common to both FOMs even when allowing for different terminology. Of the fifty-six JMRM attributes and sixty JLVC attributes, seven have a common purpose. Table 4.3-1 displays JMRM and JLVC attributes whose common purpose is identified in the first column.

Table 4.3-1 Aircraft Attributes with Common Purpose

| Purpose | JMRM | JLVC |
|---|---|---|
| Identification | Entity_ID | EntityIdentifier |
| Type | Aircraft_Type | EntityType |
| Location | Location (2D) | Spatial |
| | Altitude | |
| Association with other object(s) | Entity_List | IsPartOf |
| Damage State | Entity_List | DamageState |
| Concealment | Entity_List | IsConcealed |
| Mission Number | Call Sign / Tail Number | Mission_Number |

What about the other attributes? Those familiar with the RPR FOM, on which the JLVC FOM is based, know that many of the Aircraft, or any BaseEntity subclass, attributes describe physical characteristics detectable by sensors or provide entity state information which affects sensor detection, for example, EngineSmokeOn and RunningLightsOn. Attributes typical of the Air_Mission class in the JMRM FOM are not necessarily discernable by external sensor, like Mission, Current_Manifest, or Next_Coordination_Location. The difference is traceable, at a high level, to federation objectives but it is useful to consider the differences through another lens, this summarized by (Klein, 2007) based on Endsley's categorization of three levels of situation awareness (SA).

- **Level-1 SA** is the perception of information. For example, it is having the awareness of where different battlefield objects (enemy and friendly) are located on the battlefield at different times.
- **Level-2 SA** is the comprehension of meaning. It addresses what the Level-1 situation awareness means currently; for example, what actions the enemy is currently capable of performing.
- **Level-3 SA** is the projection of the situation over time. It is the awareness of what could happen in the future under various contingencies.

Aircraft representation in the JLVC FOM enables excellent Level-1 SA. Aircraft attributes provide detail on physical characteristics and activity which in turn enable "detection" by a variety of sensor types and enrich reporting to the training audience. In the context of Level-1 SA, the PhysicalEntity object class is

appropriately named. On the other hand, Aircraft representation in the JLVC FOM provides few clues with which to discern Level 2 or 3 SA. Certainly this is not true in every instance; one might assess the capability of a particular aircraft on the basis of it's DamageState enumeration. Few other attributes, however, provide insight for Level 2/3 SA. Air_Mission representation in the JMRM FOM, on the other hand, provides sparse data to support Level-1 SA. As documented in table 4.3-1, aircraft type and location are known, but few other details are provided on the basis of which to enrich perception. Air_Mission representation does however enable Level-2 SA. Attributes for Mission, (Weapons) Load, and Fuel_Remaining indicate the reason for the air mission's existence and ability to perform that mission. These and other attributes enable some Level-3 SA, e.g. Movement_Plan references the current plan for future movement and on/off-load of manifest objects.
We do not mean to conclude that the "actions the enemy is currently capable of performing" depend on one aircraft, or even all current air missions (since these are class attributes). We will apply the SA lens to other object classes and evaluate whether it is useful in supporting the conceptual model portion of those BOMs. Eventually the decision to use one BOM, e.g. Aircraft, and/or another, e.g. Air_Mission should be based on the federation objectives.

Comparing Interactions. Both the JLVC and JMRM federations allow munitions fired by an object owned by one federate to affect an object(s) owned by another federate. In both federates the interactions describing such an event can be best visualized by referencing a sequence diagram such as the 'Weapons Effects Pattern of Interplay' from the BOM Template.

The interactions Weapon Fire and Munition Detonation shown in figure 4.3-1 correspond respectively to the WeaponFire and MunitionDetonation interactions in the JLVC FOM. The WeaponFire interaction, used to indicate a weapon has fired, is optional for munitions other than missiles. The MunitionDetonation interaction is always sent. The Direct_Fire_Engagment (DFE) interaction is used in the JMRM federation both to indicate a weapon has fired and to indicate the munitions detonation. Like the JLVC WeaponFire interaction, the initial DFE interaction is only sent for missiles whose time of flight is long enough to warrant separate interactions for fire and impact. Unlike the JLVC, the JMRM FOM uses a different interaction, the Area_Munition_Impact interaction, for munitions with area effects. In both federations the receiving federate updates the damage state of the object if necessary.

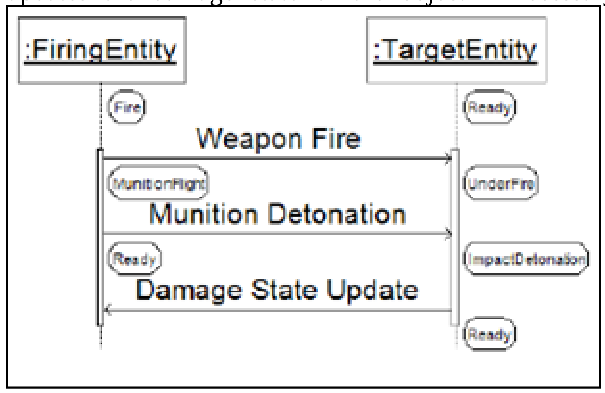


Figure 4.3-1 – Weapons Effects Pattern of Interplay

Using as an example a tank in one federate shooting a tank in another federate, we compare the MunitionDetonation interaction with 13 parameters and the DFE interaction with 11 parameters, three of which it inherits from Order. Table 4.3-2 displays the parameters with common purpose in the two interactions

Table 4.3-2 Interaction Parameters with Common Purpose

| Purpose | JMRM | JLVC |
|---|---|---|
| Firing Object | Firing_Unit | FiringObjectIdentifier |
| Point of impact | Aim_Point | DetonationLocation |
| Targeted Object | Targeted_Object | TargetObjectIdentifier |
| Munitions | Weapon_Type | MunitionObjectIdentifier |
| | | MunitionType |
| | | FuseType |
| | | WarheadType |

Figure 4.3-1 is useful for another purpose, to illustrate the use of the Model Mapping element of the BOM. Recalling the purpose of this element from Figure 3.3-1, our Air_Mission BOM, including an attribute describing the weapons load might refer to the Entity Type portion of the BOM shown in Figure 4.3-1 as the FiringEntity while the Aircraft BOM might reference the TargetEntity portion of Figure 4.3-1.

## 5 Future Work

In spite of the seemingly small amount of work accomplished thus far in analyzing the two major JFCOM Federation Object Models, we can easily envision future work that would extend the benefits of composibility. Commencing logically with additional HLA FOMs, we would subsequently include the JFCOM TENA Joint LROM and possibly BML, and the J3CIEDM model for C2 to M&S interoperability. Before embarking on such an ambitious program, we must first finish and prove the value of our initial efforts.

### 5.1 Complete Decomposition and Analysis

The current work has focused on the decomposition and analysis of the two major JFCOM FOMs. The analysis process is ongoing, but is expected to lead to a set of common object model components that can be used across both federations as well as a set of federation unique object model components.

### 5.2 OM Reconciliation

Another near term goal is to begin the process of object model reconciliation. Working with Federation Managers, we will attempt to "standardize" the common object model components as well as reconcile differences between similar object model components. Object Model components (BOMs) unique to each federation will also be identified and documented. This corresponds closely to the Level 2 interoperability discussed in (Tolk, 2003). These BOM components will be placed in the Joint Federation Engineering Library (JFEL) and will be maintained for use by other federations.

### 5.3 Conceptual Models

In addition to considering other non-HLA Object and Data Models, we will build conceptual models corresponding to the Base Object Models. Conceptual models are critical to the ability to archive, browse and assemble object models independent of the implementation of those object models. The initial effort seeks to achieve a level 2 level of conceptual interoperability (Tolk 2003) while the longer term effort will move toward levels 3 (Aligned Dynamic Data) and 4 (Harmonized Data).

### 5.4 BOM Assemblies

BOM assemblies will be used to assemble Federation Object models for the JLVC and JMRM federations using the BOMs archived in the JEFL. Assemblies for HLA, TENA, DIS, and possibly other architectures or protocols will be produced.

## 6 Conclusions / Summary

As noted above, the current work involves only JFCOM FOMs. The longer term plan is to incorporate non-JFCOM object models for coalition partners and to open the work up for a wider community involvement. We do not believe that a single FOM (sometimes referred to as the "Mother of all FOMs") is feasible. We do believe however, that there is a great deal of commonality across independently developed FOMs as well as object models for TENA and other applications such as command and control and scenario definition languages. A composable approach such as that offered by using BOMs will allow unique object model representation to be built from a common set of object models while supporting the unique requirements of individual users. It will also allow object models to be constructed for multiple uses such as HLA, TENA and DIS.

## 7 References


(Yao, 2006) *Agile Data Logging and Analysis*, Ke-Thia Yao, Gene Wagenbreth and Craig Ward, *Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC) 2006*

(Granowetter, 2005), Solving the FOM Independence Problem, http://www.mak.com/pdfs/wp_fomagility.pdf

(Cutts, 2007), Cutts, Dannie; Andy Bowers, Kevin Brandt, *"Advancing a Joint Federation Object Model"*, 07S-SIW-096, April 2007

(Davis, 2003), Davis, Paul K and Robert H. Anderson, *"Improving the Composability of Department of Defense Models and Simulations"*, Rand Monograph MG101, 2003, http://www.rand.org/pubs/monographs/MG101/index.html

(Cutts, 2006), Cutts, Dannie; Paul Gustavson, John Ashe, *"LVC Interoperability via Application of the Base Object Model (BOM)"*, IITSEC 2006, December 2006

(BOM Template, 2006), *Base Object Model (BOM) Template Specification,-SISO-STD-003-2006*, 8 May 2006, http://www.sisostds.org

(BOM Guidance, 2006), *Guide for Base Object Model (BOM) Use and Implementation-SISO-STD-003.1-2006*, 8 May 2006, http://www.sisostds.org

(Tolk, 2003), Tolk, Andreas; James A. Muguira, *"The Levels of Conceptual Interoperability Model"*, Simulation Interoperability Workshop, September 2003, 03F-SIW-007

(Chase, 2006), Chase,Tram; Paul Gustavson and Larry Root, *"From FOMs to BOMs and Back Again"*, 06S-SIW-116, April 2006

(Klein, Gary L., Leonard Adelman, Alexander Kott, *„An Integrated Theory of Collaboration"*, Under Review: CCRP International C2 Journal

(Moeller, 2007), Moeller, Bjorn, Bjorn Löfstrand, Mikael Karlsson, *"An Overview of the HLA Evolved Modular FOMs"*, Simulation Interoperability Workshop, Spring 2007, 07S-SIW-108

## Author Biographies

**ANDY BOWERS** is a Modeling and Simulation Engineer with MITRE supporting the Joint Forces Command (JFCOM) Joint Technology and Simulation Division (JTSD) in Suffolk, VA. A retired U.S. Army officer, Mr. Bowers has been involved in simulation-supported training for over fifteen years. Mr. Bowers has supported several federation development efforts and is currently the FOM manager for the JMRM and MTWS-JCATS federations. At USJFCOM he supports the JTSD Systems Engineering Branch.

**DANNIE CUTTS** is a Senior Computer Scientist with the AEgis Technologies Group Inc. supporting the U.S. Joint Forces Command (USJFCOM) Joint Technology and Simulation Division (JTSD) in Suffolk, VA. He has over 20 years experience in M&S for NASA and the DoD and has been involved with the High Level Architecture (HLA) since 1995 serving on the Interface Specification and Time Management Working Groups. He has provided HLA Training, Cadre support for DMSO, and currently serves on the IEEE Drafting Group for the HLA IEEE 1516 standard. Mr. Cutts has supported numerous federation development efforts as well as projects bringing legacy and new simulations to HLA Compliance. At USJFCOM he is involved in efforts to improve interoperability between Live, Virtual, and Constructive assets for Joint Training.