Proceedings of the 2003 Winter Simulation Conference
S. Chick, P. J. Sánchez, D. Ferrin, and D. J. Morrice, eds.

JTLS-JCATS FEDERATION SUPPORT OF EMERGENCY RESPONSE TRAINING

Francis A. Bowers, III

The MITRE Corporation
USJFCOM/JWFC
116 LakeView Parkway
Suffolk, VA 23435-2697, U.S.A.

David L. Prochnow

The MITRE Corporation
7515 Colshire Drive
McLean, VA 22102, U.S.A.

ABSTRACT

The Joint Warfighting Center (JWFC) supports Combatant Commander exercise programs with several simulation suites. Ten years ago simulated scenarios involved combat. Increasingly, however, scenarios depict crises requiring humanitarian assistance, disaster relief, or similar emergency response (ER). JWFC responded to the change in scenario requirements by developing a simulation suite using existing Joint Simulations and the High Level Architecture (HLA). This paper briefly introduces JWFC's concept of simulation-based exercise support and recommends its application to training and exercising members of an Emergency Operations Center (EOC) or other ER management staff. The bulk of the paper describes federating the Joint Theater Level Simulation (JTLS) with the Joint Conflict and Tactical Simulation (JCATS). The paper presents a notional scenario involving the simulated detonation of a chemical weapon and articulates how the decisions made by the training audience, members of an EOC, result in simulated actions and events taken to mitigate casualties.

1 INTRODUCTION

In his opening remarks during the March 2003, Modeling and Simulation (M&S) for Emergency Response (ER) Workshop, Dr. Dale Hall challenged participants to identify existing simulations, programs, and an integration framework to meet ER requirements (Hall 2003). This paper responds to that challenge by recommending the Department of Defense (DoD) exercise model to meet ER staff-level training requirements and the High Level Architecture (HLA) as an integration framework linking simulations to support that training. We support the latter recommendation with an extended description of an existing project integrating two disparate simulations, the Joint Theater Level Simulation (JTLS) and the Joint Conflict and Tactical Simulation (JCATS), to meet documented DoD training requirements. Moreover, we describe how such a federation could be used to support ER training by describing a notional ER scenario, its simulation by the JTLS-JCATS federation, and its use in an exercise support role.

We begin by describing the use of simulations to support exercises by the Joint Warfighting Center (JWFC). We then provide background information on JTLS and JCATS as well as explain the rationale for integrating the two simulations using the HLA. In section two, we discuss the federation development effort focusing on how HLA features such as ownership transfer at the attribute level, time management, and other services enabled the rapid integration of the simulations. In section three, we describe how the JTLS-JCATS federation could support a notional ER scenario. We conclude with recommending the reuse, where possible, of existing architectures, methodologies, and programs to minimize development costs.

1.1 JWFC Use of Simulations for Exercise Support

The Joint Warfighting Center (JWFC) is one of many Department of Defense (DoD) organizations using computer simulations to support staff-level training. JWFC has supported staff-level training for more than ten years. JWFC has developed an extensive body of literature to document their methodology for conducting simulation-supported exercises and their lessons learned. We believe the ER community should leverage this knowledge in designing simulation supported exercises for ER staff training.

JWFC's exercise support methodology is a good candidate model for ER staff training. It relies on an accepted pedagogic practice to maximize training transfer; the transfer of learning from classroom to job. Exercises maximize training transfer by assessing staff performance while "on the job." During exercises, staffs are engaged solving realistic problems posed by notional scenarios. The support methodology used by JWFC to conduct exercises relies on a structured process to ensure quality training. Implicit in this process are these principles:

- Identify the training audience and the training objectives.

- Develop the exercise support infrastructure to support the training, to include plans for performance assessment.
- Conduct the exercise and gather performance assessment data.
- Conduct during and post exercise performance reviews to improve performance.

1.1.1 Training Audience

The training audience that will benefit from the DoD staff-level exercise model is an Emergency Operations Center (EOC), or other emergency management staff. The model applies to staffs whose function it is to manage current crisis response while planning and coordinating future activities. The model applies to staffs using office tools: radio, telephone, e-mail, video teleconferencing (VTC), and so on, to receive information and issue instructions about an emergency, not someone using a fire axe, stethoscope, or police whistle at the scene of the crisis. The model under consideration requires the staff to use these office tools to receive information and issue instructions about a simulated emergency. The staff trains using the tools of their trade to perform their duties as they would during a real crisis because the simulation represents the crisis well enough to generate information of sufficient type and detail to form the basis for actions and decisions. Significant effort is made, in fact, to make the use of simulation(s) transparent to the training audience; they shouldn't be able to tell a real emergency from a simulated one.

1.1.2 Exercise Support Infrastructure

The infrastructure to enable exercise support includes simulation(s) that adequately represent the crisis forming the basis for the exercise, communications tools and networks to support information flows, and an exercise support staff. JWFC uses a formal evaluation process to determine the adequacy of the simulation(s) capability to support the training objectives. If one simulation adequately supports only some of the training objectives, but another simulation supports the remainder, linking the two simulations may be cheaper and faster than developing missing functionality in one or the other. Integration is also required to link simulations with communications and command and control tools. In both cases, linkages are often reused for more than one exercise. HLA, one means of integration, will be addressed in section 2. Pre-exercise support staff are required for exercise planning, assessment planning, simulation database development, scripted event development, network leasing and engineering, etc.

1.1.3 Conduct Exercise

During the exercise the simulation allows the majority of the scenario to unfold in a coherent sequence. Scenario elements not amenable to simulation are introduced to the exercise using scripted events managed by a Master Scenario Events List (MSEL). Members of "response cells" introduce many of these scripted events. Response cells are small groups of people who are the interface between the simulation and the training audience. Their function is twofold, 1. To provide information to the training audience on the situation as it unfolds and what actions they are taking, and 2. Respond to instructions from the training audience by discussing with the simulation operators how best to implement training audience decisions in the simulation. Members of the response cells keep the simulation transparent to the training audience by role playing the various people with whom the training audience would communicate during a real world crisis. Other exercise support staff include an exercise control group focused on how well the infrastructure as a whole is supporting the training, an assessment group to gather data on training audience performance, and technical people for the simulation and communications networks and systems.

1.1.4 Performance Assessment

Members of the assessment team participate in planning the exercise so that they know the training objectives, know when during the scenario they will gather data on which an assessment of performance can be made, and know what type of data they will look for. Performance reviews might occur periodically during the exercise or only after completion of the exercise. Clearly, members of the assessment team must have expertise in the subject matter being trained.

1.2 JTLS Overview

The Joint Theater Level Simulation (JTLS) system is an interactive, multi-sided wargaming system that models joint and coalition force warfare at the Operational Level. JTLS models air, ground, naval, and special operations forces, including the simulation of movement, combat, logistics, and intelligence. JWFC has used JTLS to support exercises including PACOM exercises KEEN EDGE and COBRA GOLD, EUCOM exercises AGILE LION and SHARP EAGLE, CENTCOM exercise LUCKY SENTINAL and INTERNAL LOOK, and SOUTHCOM exercise BLUE ADVANCE. JTLS is also used by other organizations to support analysis of operational plans and is the simulation of choice for NATO. The JTLS system includes its Combat Events Processor (CEP) and software for scenario preparation, order inputs, and both graphical and textual situational output. Rolands and Associates

(R&A) developed JTLS, while its graphical user interface, the Graphical Input and Aggregate Control (GIAC) system, was developed by Los Alamos National Laboratory (LANL). Figure 1 shows the JTLS system together with commonly used federates (Rolands and Associates, 2003).

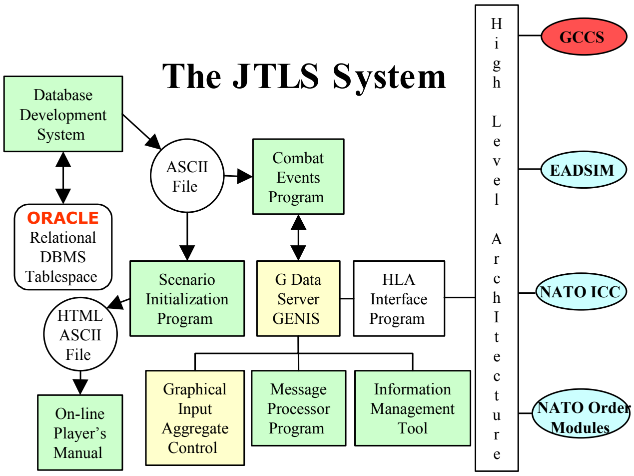

Figure 1: JTLS Diagram

1.3 JCATS Overview

The Joint Conflict and Tactical Simulation (JCATS) is a multi-service, multi-sided, interactive, entity-level simulation. Although JWFC sponsors JCATS development primarily for training, the simulation is being used in Millennium Challenge 2002 (MC02) to support experimentation, supports several analytic users, and has been used to support mission planning and mission rehearsal. JCATS is capable of simulating small group tactics, with explicit modeling of urban terrain, including detailed building features. It is also capable of supporting larger scenarios, simulating about 28,000 of the 30,000 entities in MC02.

Figure 2 shows an example of the modeling detail in JCATS. JCATS simulates chemical and biological weapon use, beam weapons, and some human characteristics. JCATS was developed by Conflict Simulation Laboratory (CSL) at Lawrence Livermore National Laboratory (LLNL) and continues to be enhanced, maintained, and supported by the JWFC (Joint Warfighting Center, 2003).

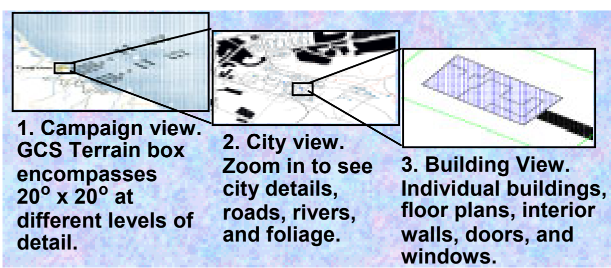

Figure 2: Depiction of JCATS Level of Detail

2 JTLS-JCATS, AN EXAMPLE HLA FEDERATION

The High Level Architecture (HLA) was designed to facilitate interoperability among simulations. In doing so, HLA also promotes reuse of existing simulations and tools. DoD experience has shown reuse can lead to tremendous savings in resources including funding, development time, training time, etc. We acknowledge that HLA is one of several means of enabling interoperability among simulations. We recommend that the ER community consider existing interoperability solutions for the same reason we recommend HLA; to avoid the resource costs associated with designing and developing a new solution.

HLA is supported by four IEEE standards:

- 1516-2000 - Framework and Rules.
- 1516.1-2000 - Federate Interface Specification.
- 1516.2-2000 - Object Model Template (OMT) Specification.
- 1516.3-2003 - Federation Development and Execution Process (FEDEP).

Rather than discussing each of these standards in turn, we will address the use of HLA by discussing an example HLA Federation, the JTLS-JCATS Federation. We do so, not to necessarily recommend JTLS-JCATS, but to indicate the effectiveness of HLA in enabling integration of very different simulations.

2.1 Federation Development

The Federation Development and Execution Process (FEDEP) is the newest HLA IEEE standard, but the start point for developing an HLA Federation. The FEDEP is essentially a systems engineering process for developing an HLA federation. Again, with reusability in mind, we recommend that the ER community consider the advantages of utilizing an existing, domain specific, systems engineering process to integrate simulations and/or other information systems. As the Integration Breakout Group at the M&S for ER Workshop found there is an extensive number and variety of integration issues; we recommend an existing process for addressing them.

We utilized the FEDEP and recommend the process to other federation developers. We produced formal deliverables, available on request, as shown in Figure 3:

- Initial (Yellow highlight).
- Federation Requirements (Blue highlight).
- Federation Development Plan (Purple highlight).
- On-Going (Red dotted).
  - Federation Object Model.
  - Federation Agreements.
  - Integration Test Plans and Summary Reports.

The JTLS-JCATS Federation Objectives included the use of the federation to support Joint Force training at mul-

Define Federation Objectives	Develop Federation Conceptual Model	Design Federation	Develop Federation	Integrate and Test Federation	Execute Federation and Prepare Results
Identify Needs	Develop Scenario	Select Federates	Develop FOM	Plan Execution	Execute Federation
Develop Objectives	Perform Conceptual Analysis	Allocate Functionality	Establish Federation Agreements	Integrate Federation	Process Output
	Develop Federation Requirements	Prepare Plan	Implement Federate Modifications	Test Federation	Prepare Results

Figure 3: FEDEP Steps and Component Activities

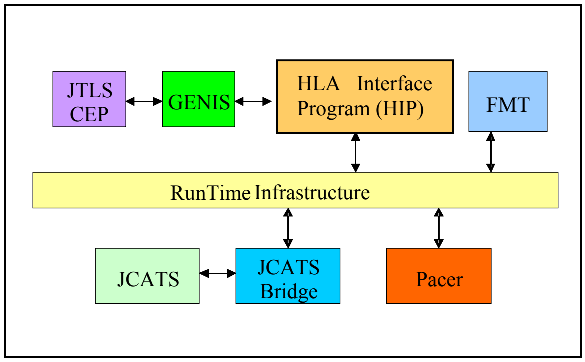

Figure 4: JTLS-JCATS Federation Architecture

tiple echelons, the requirement to address multi-resolution modeling (MRM), and manage resource constraints.

Conceptual model development was critical in meeting the above federation objectives and in subsequent design decisions. Paraphrasing the FEDEP, the conceptual model is based on a scenario(s) depicting the real world domain the federation is designed to represent. The JWFC Chief of M&S Development selected the Battle of Khafji from among six alternative scenarios as the basis for the conceptual model. The resultant model envisioned a JTLS playbox encompassing the theater area of operations within which JCATS simulated tactical operations in Khafji. The JCATS playbox is, therefore, a much smaller polygonal area overlaying a portion of the larger JTLS playbox. Our implementation of this conceptual model allows selected MRM, relevant to Joint training requirements, while avoiding solving irrelevant, and problematic, MRM challenges. A more complete description of the conceptual model development is contained in Bowers, Prochnow and Roberts (2002).

2.2 Federation Architecture

The JTLS-JCATS federation architecture is depicted in Figure 4. The JTLS Combat Events Program (CEP) serves as the wargame engine for JTLS and communicates with the GENIS, a component of the Graphical Input Aggregate Control (GIAC) System. The HLA Interface Program (HIP) connects to the GENIS and sends and receives wargame data to and from the federation through the RunTime Infrastructure (RTI). JCATS employs a program referred to as the JCATS Bridge that serves as their gateway between JCATS and the RTI. As with all HLA systems, the Runtime Infrastructure (RTI) provides the backbone for the federation. Currently, the federation employs RTI version 1.3NGv6.4.1.

In addition to the two simulations, the federation contains the Pacer federate, which allows the user to set the desired game rate. The federation also includes the Federation Management Tool (FMT), which is used for observing the state of the federate processes.

As a testament to use of a standard infrastructure for linking systems, the JTLS-JCATS federation reused elements from the prior HLA experiences of both JTLS and JCATS. In the past, JTLS had federated with the Enhanced Air Defense Simulation (EADSIM), the Global Command and Control System (GCCS), and with NATO C2 systems (Prochnow, Furness, and Roberts 2000). JCATS had participated in Millennium Challenge 2002 (MC02), a large federation effort developed at the Joint Forces Command (JFCOM). Based on these experiences, both JTLS and JCATS reused their HLA interfaces. In addition, the majority of the Federation Object Model (FOM), which defines the data to be passed within the federation, was also exploited from earlier efforts.

2.3 Federation Functionality

This section covers the basic capabilities that HLA enables in the JTLS-JCATS federation.. As HLA is supported by the Runtime Infrastructure (RTI), which is the software that adheres to the HLA specifications, a quick overview of the RTI services is given here. The RTI features six categories of services: (1) federation management for federation-wide operations such as creating and joining federations, (2) declaration management, for announcement of intent to send or receive specific classes of data, (3) object management, for the sending and receiving of persistent data, as well as transmission of events, (4) time management, for ensuring causality between systems that maintain an internal simulation time, (5) ownership management, for transferring attributes from one federate to another, and (6) data distribution management, for optimal filtering of data beyond what is offered in the declaration management services. All HLA federates must use the federation management services, and virtually all use the declaration management and object management services.

The JTLS-JCATS federation makes use of all categories of RTI services except the data distribution management services. The section highlights the federation's usage of the object management, ownership management and time management services.

2.3.1 Object Management

JTLS and JCATS both represent air, land, and sea objects. While the representation of air and sea objects is at similar levels, the simulations represent ground units at different levels of resolution. JTLS represents ground units as aggregate objects while JCATS represents these units at the entity (individual soldier) level.

An early design issue for the JTLS-JCATS team was whether to pass data across the RTI at the aggregate level or at the entity level, or both. Passing data at the JTLS level would allow use of the previously existing Federation Object Model (FOM) associated with JTLS but would require that any aggregation or disaggregation of objects occur within JCATS. Passing data at the JCATS level (i.e. entity level) would require extensive FOM changes and would force JTLS to disaggregate its objects prior to sending updates. A third option was to publicize object data at both aggregate and entity levels. This would have the advantage that another federate could subscribe to whatever level it was interested. However, having the same simulation data represented multiple times, albeit in different ways, would add a level of complexity that was beyond the scope of our work.

Ultimately, the federation team decided to pass data across the RTI at the aggregate level. This maximized reuse and minimized the traffic that had to flow over the RTI. This put the onus on JCATS to perform disaggregation and to re-aggregate data when updating the disaggregated entities. Fortunately, JCATS was already able to represent units as aggregates. In support of another project, JCATS was modified to maintain the unit hierarchy during disaggregation and reaggregation. With this change, a player can deaggregate a unit in JCATS when there is need for a greater level of detail and reaggregate it when complete.

Because data passed over the RTI for most surface objects is at the JTLS level, JCATS must update that object at the same aggregation level after it takes ownership of an object's attributes even if internally it disaggregates the object. If the JCATS user transfers the object back to JTLS, it must obviously be transferred back at the same level.

In addition to the creation and updates of objects, the JTLS-JCATS federation also employs HLA *interactions*, or events that occur between objects. The federation utilizes interactions for combat engagements, supply transfers, and other ephemeral events.

2.3.2 Ownership Transfer

The strength of the JTLS-JCATS federation is the capability to pass control of a simulated object from one simulation to another, allowing a military unit to be modeled at either an aggregate or detailed level of resolution. The multi-resolution modeling is made possible by the attribute ownership services within HLA. By exploiting these specialized RTI services, JTLS objects can be transferred to JCATS when more detailed modeling of an object is needed, and similarly, objects can be transferred from JCATS to JTLS when a more aggregate representation of the object is appropriate.

To be more precise, using HLA, it is actually the ownership of attributes (the defined components of an object) that is transferred from one simulation to another. Because of different model representations in JTLS and JCATS, it does not make sense to transfer an object attribute that has no meaning in the other simulation. Therefore, in many cases, it is a subset of the attributes of an object instance that are transferred.

HLA allows for both *push* and *pull* methods of transferring attributes. In addition, an attribute push may be *unconditional* or *negotiated*. The JTLS-JCATS federation implements ownership transfers using an unconditional push. One challenge of using this mechanism is ensuring that object attributes do not become disowned for an extended period of time. Obviously, attributes will not be owned by any federate for a very brief time during the attribute transfer process, and that is a tolerable condition. However, if a problem occurs in the transfer process causing an object's attributes to remain unowned, then this is a very serious problem. In this case, neither federate takes control of the object, and it become a sort of zombie. Although the RTI has services for querying attribute ownership, the JTLS-JCATS federation does not invoke them, and it is unclear that use of these services would be practical. Therefore, every effort is made to prevent the situation of permanently unowned attributes.

2.3.3 Time Management

Both JTLS and JCATS invoke HLA time management services to coordinate the advancement of time. In HLA terms, JTLS and JCATS are both time regulating and time constrained. That is, neither JTLS nor JCATS will advance time until the other simulation has also requested time advancement to or beyond the other simulation's current time. Users can also manage time using the Pacer federate, developed by R&A, which provides a graphical user interface to pause, resume, or advance time at a user-specified ratio of simulation time to real time.

HLA's value, and more particularly the value of the RTI, is underscored by the relative ease with which time was synchronized between these two disparate simulations. JTLS and JCATS typically operate with different time advancement mechanisms. Although most JTLS CEP events can occur at any point in time, the game time and certain events (e.g. combat assessments, supply reorders, some types of ground movement, etc.) are updated at preset intervals, typically one-minute and one-hour respectively. JCATS, on the other hand, is entirely event-stepped with events that can occur on the order of seconds or less. In addition, the HIP uses event-based HLA services while the JCATS bridge uses time-step HLA services. Despite these differences, and the primary concern that whatever is done in the federation does not adversely affect the training audience's perception of the passage of time, the time management implementation was relatively simple.

2.4 Multi-Resolution Modeling (MRM)

Multi-resolution Modeling (MRM) is necessary in the JTLS-JCATS Federation to account for differences in the two simulations. Two examples will be presented herein, MRM of objects, in 2.4.1, and MRM of supply functionality, in 2.4.2. While MRM is necessary, it also promises to be the federation's greatest feature. In fact, as was indicated in the federation objectives, JTLS-JCATS was developed in part to gain insight into MRM.

The value of MRM, in the case of JTLS-JCATS, is that the integrated system will be capable of supporting multi-level exercises integrating large-scale theater operations with small unit and individual combat system actions. This will improve JWFC's capability to support such a wide range of training audience needs without significantly increasing exercise support costs. As previously discussed, JTLS supports operational-level training. Adding JCATS-level fidelity to JTLS would be costly in terms of both development and use. In a Joint Force exercise, correct tactical employment of component forces has to be an assumption for the operational staff. Yet to a tactical training audience, the tactical employment of certain types of soldiers makes a difference in both results and outcomes. In these cases, using JCATS is critical because its fidelity is capable of representing tactical employment. In short, we developed the federation to use each simulation in the manner for which it was designed and to capitalize on the inherent strengths in each simulation, when each is needed.

2.4.1 Object Class MRM (Unit Example)

Objects in the ground domain, in particular ground units, afford ample opportunity for MRM because ground unit representation is so different in JTLS vice JCATS.

Firstly, JTLS ground units are either Aggregate Resolution Units (ARU), for example an Army Artillery Battalion or a Marine Rifle Company, or High Resolution Units (HRU), such as Special Operations Force (SOF) teams. In both cases, JTLS maintains a list of the systems comprising the unit. In JTLS version 2.5, the current version, a combat system cannot be separated from its parent unit and represented as a distinguishable object to which the user can give commands. In JCATS, units are comprised of systems that can be separated into distinguishable objects and given commands.

Secondly, algorithmic differences in conflict adjudication are solely in the ground domain. JTLS adjudicates combat between opposing ARUs using Lanchesterian equations. JCATS adjudicates direct fire combat between all individual systems using probability of hit (Ph) and probability of kill (Pk). JTLS uses Ph/Pk for weapons fired by air and naval forces, so the two simulations handle adjudication similarly in these domains.

The practical result of the differences is that JTLS ground units are comprised of generic system types in comparison to more detailed types found in JCATS units.

Table 1 shows a data mapping of an HRU, Operational Detachment Alpha (ODA) 745, an Army Special Forces A-Team. JTLS lists the majority of combat systems comprising the ODA as combat troops, as one would expect, but does not distinguish between each of the combat troops. Each contributes equally to the combat power of the unit as a whole. Other systems in the unit, MG-AGL for example, would contribute a different amount to the combat power of the unit.

Table 1: High Resolution Unit MRM

JTLS SYSTEMS	#	JCATS SYSTEMS	#
ODA 745		ODA 745	
CBT-TROOPS	11	SOF CDR	1
		SOF OPS SGT	1
		SOF COM SGT	1
		SOF DEMO SGT	2
		SOF MEDIC	2
		SOF SNIPER (12.7MM)	2
		SOF SPOTTER	2
MG-AGL	1	SOF WPN SGT (M249)	1
66MMRKT-AT4	2	JTLS ONLY	0
EXPLOSIVES	3	JTLS ONLY	0

JCATS distinguishes between far more of the systems in the unit. This is because the systems may own different weapons, e.g. the sniper, which affects the Ph/Pk for that system against other systems, or possess different capabilities, e.g. the medic if casualty play is important, or a variety of other reasons. Even the JCATS systems showing multiple entries, e.g. the two demolitions sergeants, could be modeled as two distinct systems if there were a reason to distinguish between them – exercise of a personnel replacement system, for example.

When ODA745, a named unit in the JTLS database, is transferred to JCATS, the representation of ODA745 in the federation as a whole changes from the level of detail shown in the left two columns of Table 1 to that shown in the right two columns. This is MRM. JCATS controls many of the attributes of ODA745 so, for example, the unit can be disaggregated and the members placed in particular positions to take advantage of the JCATS terrain and LOS.

As shown in ODA745, systems typically represent either major end items of equipment or personnel. System attributes are represented in the FOM as complex datatypes of variable sizes. Because JTLS and JCATS do not track all of the same individual equipment types, it does not make sense to pass ownership of these attributes. Instead, JTLS always owns these attributes. The data type for the number of each system is real. Before JTLS transfers control of a surface object, it updates another attribute that represents the integer number of manned systems. JCATS uses this integer to instantiate the correct number of combat systems in the unit. When a combat system is damaged or destroyed in JCATS, JCATS sends an HLA interaction to communicate state changes of the individual equipment types that it represents. JTLS in turn makes a corresponding update to the appropriate system list.

Similarly, when a JTLS ground object is transferred to JCATS, JTLS retains ownership of attributes associated with the number of personnel in the object. For purposes of personnel accounting, infantry is explicitly counted while vehicle crews are implicitly represented. If explicitly represented personnel become casualties in a surface object, JCATS sends an interaction to JTLS to pass this information. In turn, JTLS updates the relevant personnel list, to include calculating crew casualties for vehicles that have sustained damage.

Other attributes of the unit objects, location for example, are owned by JCATS. Sharing ownership of object attributes enables MRM because without it repeated transfers of a unit would cause repeated changes to the number of combat systems in the unit as the number changed from real to integer and back.

2.4.2 Functional MRM

MRM is not only, however, a function of object representation. JTLS and JCATS differ functionally in many areas and these provide opportunities for implementing MRM while, more importantly, providing improved functionality to the user.

For example, JTLS and JCATS differ in their representation of supplies. Depending on the training audiences desires for supply representation, and the resulting database build, JCATS objects consume fuel and ammunition, classes III and V respectively, and carry realistic quantities of each. JCATS also provides functional use of barriers like wire or sandbagged positions, but doesn't explicitly link the use of these to consumption of the class IV supplies necessary to construct them. JCATS obviously represents major end items, like tanks, but does not typically represent the supply chain replacement of class VII. JTLS training audiences are more likely to exercise supply and resupply, albeit at the operational or theater level, so JTLS enables consumption of each of the classes of supply I through X. Although JTLS represents some supplies, e.g. classes I and II, that JCATS doesn't, JCATS typically represents far more ammunition types than does JTLS.

We capitalized on this difference in resolution with respect to supplies in the JTLS-JCATS Federation by ensuring that units originating in JTLS transferred to JCATS continue to consume all classes of supplies represented in JTLS. The means of doing so is shared object ownership.
It clearly does not make sense to pass ownership of supply attributes from JTLS to JCATS for supply types JCATS does not represent. So, as in the case of combat systems, class VII major end items, JTLS retains ownership of other supply attributes. JCATS consumes classes III and V and reports on-hand quantities using an interaction. JTLS updates the attribute values and also uses the current values and reorder levels, a JTLS parameter, to know when to start resupply of the unit. MRM is enabled by shared object ownership of supplies because the federation as a whole better represents the consumption of supplies than either simulation does alone.

3 FEDERATION SUPPORT OF ER STAFF TRAINING

The example presented in this section is designed to illustrate the use of a federation to support ER staff training. The example necessarily uses JTLS-JCATS as we are familiar with the functionality of these simulations, but our recommendation extends to use of any federation with the requisite functionality. We briefly describe a notional scenario and devote the bulk of the example to allocating the functionality between simulations, underscoring functional shortfalls in the individual simulations that are resolved by federating.

3.1 Notional ER Scenario

A terrorist group detonates a chemical weapon in a city of 200,000 on the Eastern Seaboard. The release occurs shortly before sunset to take advantage of the evening inversion layer over the city, to maximize casualties during rush hour traffic and to complicate relief efforts with the arrival of darkness.

3.2 Allocate Functionality

JCATS is used to simulate the chemical weapon detonation. Initial casualties and the spread of the plume are modeled in JCATS, based on updates from VLSTRACK. The evacuation of residents downwind from and "close", e.g. 0 to 6 hours, to the source is simulated in JCATS, but farther away (6+ hours; these times may need to be adjusted based on the affected area and corresponding population size) is simulated in JTLS. This allows people fleeing the plume close to its source to be individually represented as are the means (foot, vehicle, boat) which they use to flee, attendant traffic congestion, and opportunities for local response measures to alleviate congestion, promote orderly evacuation, render medical assistance and evacuation, etc. Evacuation further from the source can be simulated at a more aggregate level, which will be necessary given the large number of people involved. Organized mass evacuation by convoy and air is also done more easily in JTLS and, in the case of the convoys, with perhaps better representation of the net (after congestion) traffic flow. The exchanges necessary between the simulations might initially be as limited as those:

- necessary for JCATS to update JTLS on contaminated areas as the plume spreads with the wind.
- required to maintain coherency of responder personnel, vehicles and equipment.

More elaborate HLA services could be utilized and would be appropriate for particular events, e.g. groups of evacuees, in JCATS, using different modes of mass transit, in JTLS, to distance themselves from the local area.

As previously noted, local response is simulated, as much as is possible, in JCATS. While JCATS supports representation of emergency and chemical response teams, vehicles, people, and many other objects and activities, the effectiveness of many response efforts will be better judged by a white cell. The cell can subsequently direct decreases, or increases, in the severity of casualties or local incidences based on the effectiveness and comprehensiveness of the response. For example, if local law enforcement officials are prompt in initiating evacuation and managing traffic flow out of the area, one might expect fewer chemical casualties. Vehicle accidents, looting, and other incidences which disrupt the official response to alleviate catastrophe and attempts to locate the perpetrators to forestall a second detonation could be scripted and introduced, or not, depending on the training audience response and flow of the exercise.

In addition to simulating organized mass evacuation, JTLS will be used for regional and national response including the deployment of personnel and equipment to the city and the region to assist in chemical decontamination, medical treatment, and the establishment of shelters for the displaced. Consumption of food and water, medical supplies, and personal care necessities such as bedding, tenting, etc. would also necessarily occur in JTLS. The exchanges necessary between the simulations might include transferring some personnel deploying into the region with their bulk equipment into JCATS so that they can be simulated at the individual or small team level with attendant equipment, i.e. protective clothing, masks, chemical detection equipment, decontamination equipment, etc., or medical equipment. Conversely, simulating the displaced persons camps should be done in JTLS at the aggregate level for ease of management.

3.3 Exercise Support

The brief scenario description and allocation of functionality does not portray all the detail available, of course, from simulation of the event and its consequences. Hopefully it is sufficient to envision some of the considerations, planning, and activities that an EOC staff would face in reacting to such an event. Coordination for personnel and equipment support, transportation of ER personnel into the crisis area and citizens away from the crisis, arranging support facilities for ER personnel and displaced citizens alike, and the host of other actions comprising the real world responsibilities of the EOC staff are engendered through such scenarios. Dealing with these real world types of issues are, of course, the means of training.

4 RECOMMENDATIONS

The Defense Department has developed, over an extended period of time, a staff training methodology utilizing exercises to maximize training transfer. We recommend that the ER community evaluate this methodology for use in training EOC or other ER management staffs.

The HLA is an industry-accepted, IEEE-documented, means of integrating simulations. It includes a domain specific systems engineering process and software tools for integrating simulations. It promotes reuse of existing simulations thereby minimizing or avoiding extensive software development efforts. We recommend that the ER community evaluate use of HLA for their simulation integration requirements.

The JTLS-JCATS federation development proved the value of HLA in that it allowed the reuse of existing simulations, in the manner for which each was designed, capitalizing on the inherent strengths in each simulation. Moreover, the federation provides greater combined functionality than either single simulation. HLA enabled rapid capability development, and has provided a useful, cost-effective product.

REFERENCES

Bowers, A., D. Prochnow, and J. Roberts. 2002. JTLS-JCATS: Design of a Multi-Resolution Federation for Multi-Level Training. In *Proceedings of the Fall 2002 Simulation Interoperability Workshop.* Orlando, Florida. Available online via <http://siso.sc.ist.ucf.edu/index.cfm> [accessed July 11, 2003]

Hall, D. 2003. Opening Remarks. In *Presentations of the Modeling and Simulation (M&S) for Emergency Response (ER) Workshop.* Available online to workshop attendees only via <http://www.mel.nist.gov/div826/msid/sima/simconf/mns4er.htm> [accessed July 12, 2003]

Joint Warfighting Center, US JFCOM. *Joint Conflict and Tactical Simulation (JCATS)* Available online via <http://www.jtasc.acom.mil/genpublic/jw500/jcats/> [accessed July 11, 2003]

Prochnow, D., Furness, C., Roberts, J.. 2000. The Use of the Joint Theater Level Simulation (JTLS) With the High Level Architecture (HLA) to Produce Distributed Training Environments. In *Proceedings of the Fall 2000 Simulation Interoperability Workshop*. Orlando, Florida. Available online via <http://siso.sc.ist.ucf.edu/index.cfm> [accessed July 11, 2003]

Rolands & Associates. 2003. *JTLS Executive Overview*. Available online via <http://www.rolands.com/> [accessed July 11, 2003]

AUTHOR BIOGRAPHIES

FRANCIS A. BOWERS is a Senior Modeling and Simulation Engineer at MITRE Corporation's Joint Warfighting Center site. A retired U.S. Army officer, Mr. Bowers holds a Bachelor's degree from the United States Military Academy and a M.S. in Operations Research and a M.S. in Civil Engineering from Stanford University. You can reach him by e-mail at <bowersa@mitre.org>.

DAVID L. PROCHNOW is a Lead Software Engineer in the Information Systems and Technology Division at the MITRE Corporation and is currently the technical lead on several High Level Architecture programs. At MITRE, Mr. Prochnow previously contributed to the ALSP project and to logistics simulation efforts. While at BDM International and Control Data Systems, Mr. Prochnow developed software for various corps-level and theater-level wargames. He received a B.S. in Computer Science from the University of Virginia in 1983. He can be reached by e-mail at <prochnow@mitre.org>.