Modeling and Simulation Coordination Office (M&S CO) High Level Task (HLT) S-C-2
Live-Virtual-Constructive Architecture Roadmap Implementation (LVCAR-I)
FY09-Funded Key Accomplishments and Deliverables

This document presents the key accomplishments and deliverables produced with Fiscal Year 2009 (FY09) funding under the Live-Virtual-Constructive Architecture Roadmap Implementation (LVCAR-I) tasking by the Johns Hopkins University Applied Physics Laboratory (JHU/APL) for the Joint Training Integration and Evaluation Center (JTIEC). Each of the three sections, Common Capabilities, Gateways and Bridges, and Convergence has four components:

1. Objective and Major Subtasks
2. Key FY09 Funded Accomplishments
3. FY09 Funded Deliverables
4. Consolidated Executive Summaries

While this synopsis is no substitute for examining the actual referenced documents the user can get an overview of the topics covered and then quickly identify the product that is of most interest. All documents are available to DoD personnel through the internet at the Modeling and Simulation Coordination Office resource repository (http://www.msco.mil/).

In the course of fulfilling the project's goals we periodically ask for and benefit from input by member organizations of the DoD M&S Community. When requested your assistance in helping us meet our objectives will be greatly appreciated.

Common Capabilities

Objective and Major Subtasks

Objective: Develop a set of products that facilitate cross-architecture communication and collaboration, and reduce divergence and duplication among LVC user communities

Major Subtasks:

- Common Systems Engineering Process
  - Develop a common systems engineering process model for mixed-architecture LVC simulation environment development and execution
- Common Federation Agreements Template
  - Develop an architecture-independent template for establishing federation agreements that can be supplemented by potential architecture-specific extensions
- Reusable Development Tools & Common Data Storage Formats
  - Examine the various options associated with efficient, effective sharing of tool resources for LVC applications, and recommend the most beneficial approach or approaches, including benefits and barriers
- Asset Reuse Mechanisms
  - Examine existing infrastructure capabilities for M&S reuse, compare these capabilities to the required mechanisms for sharing and reuse described in the LVC Architecture Roadmap, and

develop and implement a plan to ensure the appropriate discovery and distribution mechanisms are available in the future

Key FY09-Funded Accomplishments

- Identified 41 multi-architecture LVC systems engineering (SE) issues and developed recommended actions to address each in draft SE Process [Ref. 1]
- Identified 58 applicable federation agreements in eight categories, and described each in draft Multi-Architecture Federation Agreements Template [Refs. 2, 3]
- Identified 266 existing LVC-related tools, and categorized by architecture, function, and step of the Distributed Simulation Engineering and Execution Process (DSEEP) [Ref. 4]
- Described eight candidate business models for reusable LVC tools [Ref. 4]
- Identified 179 applicable data storage formats and characterized each within nine major categories [Ref. 5]
- Developed 22 use cases for asset reuse, examined 13 existing M&S catalogs, repositories, and registries; and delineated three complementary approaches (transactional, social marketing, and process-based) for improving LVC asset reuse [Ref. 6]

Developed implementation plans for reusable tools, common data storage formats, and asset reuse mechanisms, with prioritized actions [Refs. 4, 5, 6]

FY09-Funded Deliverables

[1] Guide for Multi-Architecture Live-Virtual-Constructive Environment Engineering and Execution (June 2010) [JHU/APL NSAD-R-2010-044]

[2] Federation Agreements Template Schema and Programmer's Reference, June 2010 [DVD]

[3] Federation Agreements Template User's Guide (June 2010) [JHU/APL NSAD-R-2010-052]

[4] Reusable Tools Implementation Plan (March 2010) [JHU/APL NSAD-R-2010-019]

[5] Common Data Storage Formats Implementation Plan (May 2010) [JHU/APL NSAD-R-2010-043]

[6] Asset Reuse Mechanisms Implementation Plan (April 2010) [JHU/APL NSAD-R-2010-023]

Consolidated Executive Summaries

Guide for Multi-Architecture Live-Virtual-Constructive Environment Engineering and Execution (June 2010) [JHU/APL NSAD-R-2010-044]

Robust, well-defined systems engineering (SE) processes are a key element of any successful development project. In the distributed simulation community, there are several such processes in wide use today, each aligned with specific simulation architecture such as Distributed Interactive Simulation (DIS), High Level Architecture (HLA), and Test and Training Enabling Architecture (TENA). However, there are an increasing number of distributed simulation applications within the Department of Defense (DoD) that require the selection of simulations whose external interfaces are aligned with more than one simulation architecture. This is what is known as a *multi-architecture simulation environment*.

Many technical issues arise when multi-architecture simulation environments are being developed and executed. These issues tend to increase program costs and can increase technical risk and impact schedules if not resolved adequately. The Live-Virtual-Constructive Architecture Roadmap (LVCAR) was initiated in 2007 to define the differences among the major simulation architectures from technical, business, and standards perspectives and to develop a time-phased set of actions to improve interoperability within multi-architecture simulation environments in the future.

One of the barriers to interoperability identified in the LVCAR Phase I Report was driven by a community-wide recognition that when user communities, aligned with the different simulation architectures, are brought together to develop a multi-architecture distributed simulation environment, the differences in the development processes native to each user community adversely affected the ability to collaborate effectively. To address this problem, a recommendation was made to establish a common cross-community SE process for the development and execution of multi-architecture simulation environments. However, rather than develop an entirely new process, it was recognized that an existing process standard should be leveraged and extended to address multi-architecture concerns. The process framework that was chosen is an emerging Institute of Electrical and Electronics Engineers (IEEE) standard called the Distributed Simulation Engineering and Execution Process (DSEEP). The DSEEP tailors widely recognized and accepted SE practices to the modeling and simulation domain and, more specifically, to the development and execution of distributed simulation environments. The strategy implemented in this case was to augment the major DSEEP steps and activities with the additional tasks that are needed to address the issues that are unique to (or at least exacerbated by) multi-architecture development. These tasks collectively define a "how to" guide for developing and executing multi-architecture simulation environments, based on recognized best practices.

This document defines a total of 40 multi-architecture related issues, based on an extensive literature search. Each of these issues is aligned with the activity in the DSEEP for which the issue first becomes relevant. Each issue comes with both a description and a recommended action(s) to best address the issue. A set of inputs, outcomes, and recommended tasks is also provided for each DSEEP activity to address the resolution of the multi-architecture issues. This information is provided as an overlay to corresponding information already provided in the DSEEP document for single-architecture development.

An appendix to this document identifies a tailoring of the guidance provided in the main document to individual architecture communities. For each of three major simulation architectures, a mapping is provided to indicate the relevance of each Issue–Recommended Action pair to developers and users of that simulation architecture. Together with the guidance provided in the main text, it is believed that this document will provide the guidance needed to improve cross-community collaboration and thus reduce costs and technical risk in future multi-architecture developments.

Federation Agreements Template Schema and Programmer's Reference, June 2010 [DVD]

There is no executive summary associated with this deliverable. The DVD contains a machine readable product.

Federation Agreements Template User's Guide (June 2010) [JHU/APL NSAD-R-2010-052]

Federation agreements are critical to the successful design, execution, and reuse of federation assets. However, inconsistent formats and use across federations have made it difficult to capture and compare agreements between federations. This lack of a consistent approach to documenting federation

agreements makes reuse and understanding more difficult. Lack of consistent format also prevents tool development and automation. The federation agreements template is intended to provide a standardized format for recording federation agreements to increase their usability and reuse. The template is an eXtensible Markup Language (XML) schema from which compliant XML-based federation agreement documents can be created. XML was chosen for encoding agreements documents because it is both human and machine-readable and has wide tool support. Creating the template as an XML schema allows XML-enabled tools to both validate conformant documents, and edit and exchange agreements documents without introducing incompatibilities. Wherever possible, the team that prepared the template leveraged existing, authoritative schemas for the representation of elements in this schema including:

- Modeling and Simulation (M&S) Community of Interest—Discovery Metadata Specification (MSC-DMS)
- XML Linking Language (XLink)
- XML Metadata Interchange (XMI)
- Common Platform Enumeration (CPE)
- Intelligence Community Information Security Marking (IC-ISM)
- eXtensible Configuration Checklist Description Format (XCCDF)
- Geography Markup Language (GML)

The federation agreements are decomposed into eight categories:

1. Metadata—Information about the federation agreements document itself.

2. Design—Agreements about the basic purpose and design of the federation.

3. Execution—Technical and process agreements affecting execution of the federation.

4. Management—Systems/software engineering and project management agreements.

5. Data—Agreements about structure, values, and semantics of data to be exchanged during federation execution.

6. Infrastructure—Technical agreements about hardware, software, and network protocols, and processes for implementing the infrastructure to support federation execution.

7. Modeling—Agreements to be implemented in the member applications that semantically affect the current execution of the federation.

8. Variances—Exceptions to the federation agreements deemed necessary during integration and testing. This document also provides brief examples of mapping an existing federation agreements document to a document conformant with the XML schema.

Reusable Tools Implementation Plan (March 2010) [JHU/APL NSAD-R-2010-019]

The Live-Virtual-Constructive (LVC) Architecture Roadmap (LVCAR) Study (Comparative Analysis of the Architectures and Comparative Analysis of Business Models) identified two significant impediments to sharing and reuse of event development tools across programs and communities. The first is the existence of a wide range of tools utilizing a correspondingly wide range of business models. The second impediment is the current environment where different formats are used by the different architectures to

store like event data. The purpose of the LVCAR Implementation Reusable Tools task is to identify the most beneficial approach that will facilitate tool sharing across architectures based on a structured analysis of the current state.

A key deliverable in the LVC Implementation task is plan for implementing a library of reusable LVC tools. The activities described in this, the Reusable Tools Implementation Plan, build the content for the planned library, and the activities described in the LVC Implementation Asset Reuse Implementation Plan will provide the mechanisms and form the basis of repository for tools access. The analyses found in this document inform the decisions for adoption of the best methodology for the acquisition and distribution of LVC tools that will be found in the repository defined and implemented in the LVC Implementation Asset Reuse task.

The study team began this analysis by concurrently establishing a set of candidate business models and developing an inventory of existing LVC tools. These business models were drawn from prior LVC Architecture Roadmap research and an assessment of current trends in software licensing:

1. Department of Defense (DoD) Wholly-owned and Centrally Distributed - A DoD organization owns the rights to the tool(s) and provides them to interested DoD organizations. This business model, also known as government-off-the-shelf (GOTS), is implemented by the organizations that support the Test and Training Enabling Architecture (TENA) and the Common Training Instrumentation Architecture (CTIA).

2. Laissez-faire - Each user organization or "buyer" decides what to adopt. Those adoption options can include purchasing commercial-off-the-shelf (COTS) software, adopting DoD wholly-owned tools, or adoption of an open source tool. Since DoD wholly-owned and open source are assessed as separate business models and described in subsequent subsections, the evaluation of the Laissez-faire business will focus on the purchase of COTS tools.

3. Open source - DoD fosters open source tools by contributing to their development and sustainment. This is the Open Technology Development approach recommended in the LVCAR Comparative Analysis of Business Models [Reference (a)]1

1 A list of References may be found in Appendix A.

4. Laissez-faire with Increased Visibility - This is essentially the same as Laissez-faire, but DoD puts an emphasis on making programs' choices, and more importantly, their experiences, more visible. The visibility mechanisms are addressed by the LVCAR.

5. Standards-focused - In this approach, DoD does not take an active role to try to foster or force tool reuse but rather obtains the benefits of interoperability by focusing in on standards for tool data interchange. This is another variant on Laissez-faire, but tool compliance with the standards is used to "limit" the market.

6. Preferred Provider List - DoD publishes a list of preferred tools. Programs are "encouraged" to buy from this list or justify why not.

7. Central License Negotiation - DoD central purchasing organization receives tool requirements from users, makes selections on tools, and negotiates terms and licensing fee(s) for DoD users. Unlike the DoD wholly-owned, the individual organizations procure the tools with their own funds.

8. Software as a Service - Rather than purchasing licenses for installation, DoD negotiates with service providers. Delivery mechanisms vary across vendors. The software may be downloaded at run-time from the provider's server, may be provided via a services oriented architecture (SOA) interface, may be provided via a web client, or may be locally installed for a negotiated period of time.

A set of comparative factors was established by the study team to characterize business models.

In order to compare business models quantitatively, the study team identified a set of benefits and barriers by reviewing prior literature, as well as through the experience of the study team participating in previous LVC events. This analysis was performed using a quantitative method based on subject matter expertise from the study team's background and knowledge because observed data does not exist for an objective, quantitative analysis. In the absence of this subjective, but quantitative approach, the only recourse given the absence of observed data would be a purely subjective analysis.

The weights were developed by consensus of our expert team and reviewed by doing pair-wise comparisons of "is 'a' more important than 'b'?" As the study team did this consensus review of weights and likelihoods, some values change without dramatic change to the results; the top approach remained on top. Some of the lower rated approaches, where the ratings were close, shifted in order. For the purpose of this report, LVC tools include software used throughout the lifecycle of development and execution of an LVC event, with the exclusion of simulations, bridges and gateways, and middleware. The second and third items are addressed through other efforts. This inventory provided the analytic underpinnings key to understanding the market in which the development tools are developed and distributed. Currently, tools are developed and distributed through one of the first three of the business models listed above: laissez-fare, DoD wholly-owned, and open source, leading to the wide range of business models noted in the LVCAR Study (Comparative Analysis of the Architectures and Comparative Analysis of Business Models) as an impediment to reuse.

In order to compare business models objectively, the study team identified a set of benefits and barriers by reviewing prior literature, as well as through the experience of the study team participating in previous LVC events. Benefits and barriers were included only if they had a positive or negative impact on the Department of Defense (DoD). These benefits and barriers have been assigned weights, reflecting their relative impact on DoD. Each business model was then assessed for the likelihood of encountering those benefits and barriers.

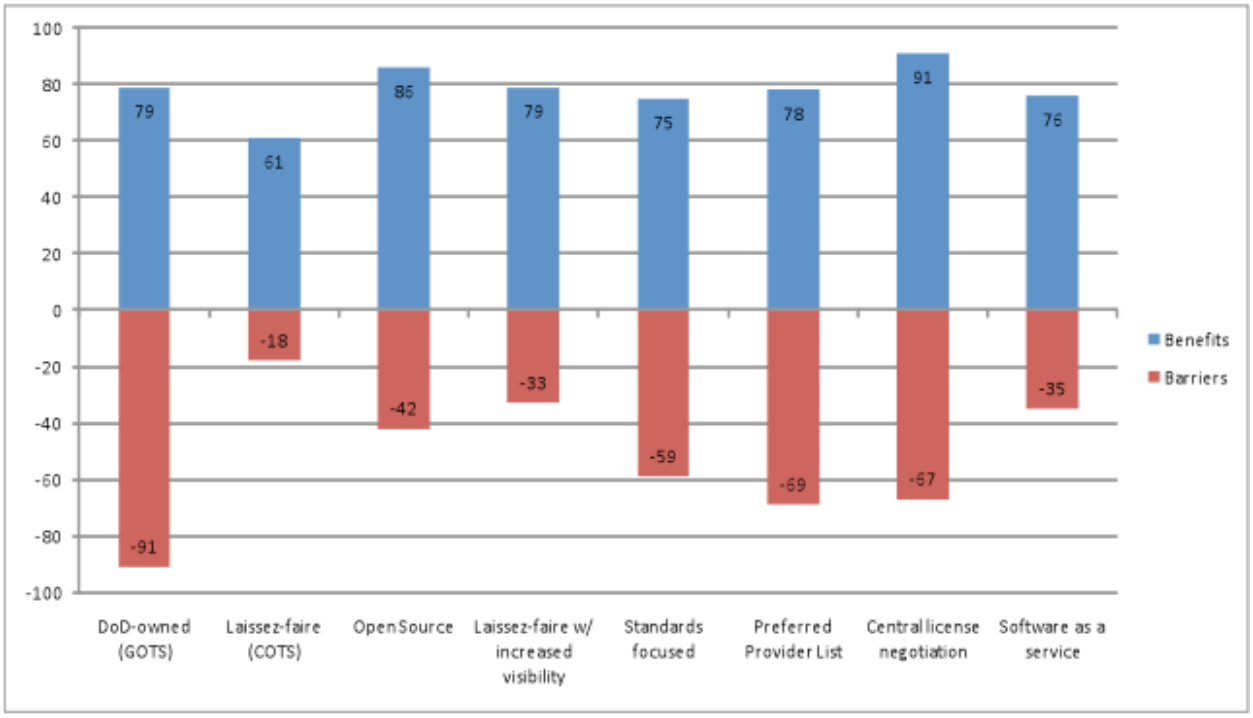

Figure ES-1: Relative Benefits and Barriers of All Business Models

The benefit and barrier scores for each of the business models found in Figure ES-1 are calculated by multiplying the weighted relative impact by the numerical value for the likelihood of encountering the benefit or barrier. Based on the magnitude of the relative benefits and barriers, as illustrated in Figure ES-1, combined with experience from participation in DoD LVC events, the following have been identified as key findings of this analysis:

1. A change to any of the other business models has more benefit to DoD when compared to staying with the current Laissez-faire model.

2. The DoD-owned business model compares well with other business models with respect to benefits but has significant barriers.

3. Central license negotiation appears to have the largest benefit but has significant barriers.

4. Open source appears to have the largest single benefit but does not have to be implemented in isolation.

5. Standards have a significant benefit and can be used in conjunction with other business models. Although there are a significant number of barriers, the barriers likely to be encountered are gauged to have low impact.

6. It is possible to reap the benefits of open source and standards-focused in concert, resulting in software available to DoD at little to no cost (providing the benefits of DoD-owned solutions).

7. On the surface, software as a service appears to be more beneficial to DoD than procuring software licenses. However, this technique has not been exploited in DoD and experiences are needed to validate this finding.

8. Central license negotiation appears to be clearly superior to preferred provider lists.

9. Increased visibility shows significant benefits that significantly outweigh barriers associated with implementing this business model.

Based on these findings, the study team developed both short- and long-term recommendations for implementation. The short-term recommendations are intended to provide risk reduction before making more broadly sweeping long-term changes. They consist of pilots and initiatives that will determine feasibility and provide lessons learned.

1. Execute a set of pilots to attempt to move some tool categories (especially DoDowned) to business models with higher benefit to DoD.

2. Explore the barriers to central license negotiation to see if any can be ameliorated.

3. Conduct a pilot that transitions a currently DoD-owned tool to open source, while basing the effort on open standards.

4. In addition to the open source pilot effort, foster the establishment of open standards.

5. Gather information from industry to 1) inform industry that DoD is interested in the software as a service business model, and 2) determine if there are industry providers who would be interested in offering software as services.

6. Regardless of the business model selected for a tool category, take actions to increase visibility of tools in use, and gather user feedback. Improved visibility across the board will enable decision-makers under all business models to make more informed decisions, particularly using feedback from other users' experiences with tools under consideration. This recommendation is supported by the results of the LVCAR Implementation Asset Reuse task.

Implementation plans for the recommended pilots have been developed. Assuming the short-term pilots and explorations produce the desired outcomes, a set of long-term recommendations can be implemented. Figure ES-2 illustrates the relationship between the recommended changes from current practices to a set of actions that both change business model focus for reusable tools and take other actions beneficial to DoD's reuse goals.

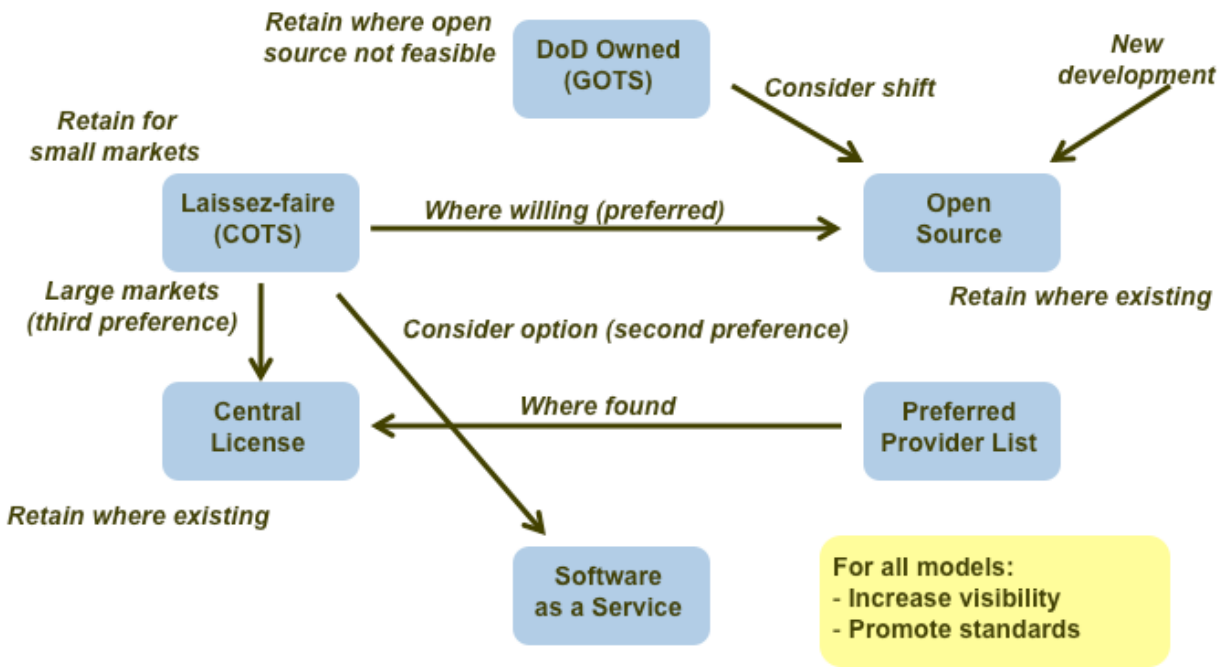

Figure ES-2: Relationship of Long Term Recommendations

Individual long-term recommendations based on the analysis represented in this table are as follows:

1. For legacy DoD-owned tools, consider a shift to open source, to reduce DoD costs and foster potential innovations. Use the experiences from the short-term recommendation #3 open source pilot to decide if this should be done, and if so, what considerations exist for LVC tools.

2. For new tools, where there is a desire to provide DoD influence but to defray ownership costs, use an open source model also informed by the open source pilot.

3. Where small numbers of licenses are purchased from industry, do not make a change.

4. Where a large number of licenses have been and continue to be procured from industry, take the following actions in the order presented until a viable option is identified:

a. Shift to open source. This assumes that vendors are willing and the open source pilot experience described in short-term recommendation pilot #3 indicates there is benefit to DoD.

b. Shift to software as a service. This assumes that vendors are willing and the experiences from the short-term recommendation #5, software as a service pilot, shows benefit to DoD exists.

c. Attempt to negotiate DoD-wide discounted licenses.

5. For current open source efforts, make no changes.

6. If preferred provider lists have been established, attempt to establish DoD-wide discounted licenses, using the experiences gained from the short-term recommendation #2 pilot.

7. For existing centrally-negotiated licenses, do not make a shift.

8. The study team is unaware of any existing “software as a service” arrangement for LVC tools, so no recommendations in terms of a shift from current practices are made.

9. For all business models, increase the visibility of what tools are currently used, and take steps to increase the visibility of user experiences as indicated by the LVC Software/Asset Reuse Mechanism task.

10. Consistent with DoD policy, use open standards as a basis for tool procurements, and participate in standards development activities to ensure DoD’s needs are met.

Common Data Storage Formats Implementation Plan (May 2010) [JHU/APL NSAD-R-2010-043]

The Live-Virtual-Constructive Architecture Roadmap (LVCAR) report recommended actions to promote the sharing of tools, data, and information across the Enterprise and to foster actions to promote the sharing of tools, data, and information across the Enterprise and to foster common formats and policy goals to promote interoperability and the use of common Modeling and Simulation (M&S) capabilities. One of the recommended actions was to examine different data storage formats used across the various architectures to determine the feasibility of creating a set of architecture-independent formats. Such formats would be used for storage of classes of data in order to mitigate the cost and schedule impacts of database conversion, minimize conversion errors, and improve consistency across Live-Virtual-Constructive (LVC) architectures. The focus of this effort is limited to data interchange formats and applicable standards where the data is persistent, e.g., in stored datasets.

The study team identified nine (9) categories of data storage formats, based on expertise and feedback received at the LVC Common Capabilities Workshop at The Johns Hopkins University Applied Physics Laboratory (JHU/APL) in November 2009 and questionnaires administered in person at the Interservice/Industry Training, Simulation and Education Conference (I/ITSEC) 2009 and online. This stakeholder feedback was used to assess the priority for rationalization of data storage formats for each category. The team examined the contents of eight (8) metadata standards registries, catalogs and repositories for each category identified.

These sources included the Department of Defense (DoD) Metadata Registry, DoD Information Technology Standards and Profile Registry (DISR), the North Atlantic Treaty Organization (NATO) and DoD M&S Standards Profile, and the Acquisition Streamlining and Standardization Information System (ASSIST) database, in addition to privately maintained source materials. For each of the nine (9) format categories, a list of applicable formats was compiled and characterized in terms of currency, openness, maturity, and applicability as a source (producer), interchange (mediation) and executable (consumer) data format. This information was used to assess the difficulty of rationalizing formats within each category.

In addition, the team developed a strategy for each of the nine categories by evaluating the feasibility of moving to a state of greater reuse via a combination of: (1) reduction in the= number of formats used in each category; (2) standardization of formats in each category if no standards exist; (3) increased adoption of mediation formats to reduce translation errors; and (4) creation or engagement with category-specific communities of interest (COIs).

Using this prioritization approach, the standardized formats should be pursued in the following order:

Priority 1

- Man-made features

- Event results

Priority 2
- Geospatial

Priority 3
- Unit Order of Battle (UOB)
- Plans/scenarios

Priority 4
- Platform/Weapons performance
- Behavior

Priority 5
- Electronic Order of Battle (EOB)/network
- Logistics

The team identified the following list of general engagement strategies for improving the state of any category:

- Engage with ongoing efforts to ensure they include LVC-specific requirements,
- Establish forum/process for mediating between dissenting communities,
- Initiate activity to bring key stakeholders together to develop format where no clear viable solution currently exists,
- Accelerate promising, but slow, ongoing efforts,
- Enable management/cooperation changes, and
- Monitor until sufficient meaningful direction emerges.

Table ES-1 consolidates the recommendations, feasibility, and priority with these general strategies to provide specific strategies and timelines for each category.

(Refer to original document for associated tables)

Asset Reuse Mechanisms Implementation Plan (April 2010) [JHU/APL NSAD-R-2010-023]

The reuse of software, data, and other assets in Department of Defense (DoD) Modeling and Simulation (M&S) development is neither as frequent nor as effective as it could be, and as a consequence, the potential benefits of reuse to the DoD enterprise are not being fully realized.
Improvements in the enterprise culture and processes supporting reuse are needed to increase the frequency of reuse. Three alternative approaches to accomplishing those improvements were defined and evaluated. Enhancements to the capabilities and coordination of DoD M&S asset repositories are needed to increase the effectiveness of reuse. An assessment of multiple existing repositories using a carefully developed set of M&S-oriented evaluation criteria was conducted to identify where those enhancements are needed.

The LVC Asset Reuse Implementation study team examined thirteen (13) existing M&S catalogs, repositories, and registries of interest to the Live-Virtual-Constructive (LVC) Architecture Roadmap Implementation effort and evaluated the applicability of these and other reuse initiatives. A detailed model of LVC asset reuse mechanisms based on twenty-two (22) comprehensive reuse use cases tied to

the DoD Net-Centric Data Strategy and commercial standards for repositories was developed and used to facilitate the research and analysis conducted. Consideration of the state of these LVC asset reuse mechanisms, together with feedback from stakeholders within all communities enabled by M&S in the form of questionnaires, workshop discussions, and interaction in the government-industry profession, informed this study and recommendations.

Three complementary approaches to improve LVC Asset Reuse mechanisms were examined. The Transactional Approach focuses on enhancing the discovery and acquisition of reusable M&S assets through a set of distributed, interconnected M&S catalogs, registries, and repositories**.** The Social Marketing Approach addresses the long term improvement of behaviors that promote reuse of M&S assets**.** The Process-Based Approach encourages more frequent reuse by enhancing reuse guidance within standard DoD M&S systems engineering process models.

These three approaches were evaluated in terms of desirability, achievability, and affordability, as well as the likely barriers to their success. The Transactional Approach was rated as the most affordable due to existing investments and is roughly equivalent to the Process-Based Approach in terms of desirability. The Process- Based Approach was rated as the most easily achievable based on its compatibility with ongoing standards initiatives in M&S systems engineering processes, and also an emerging impetus towards Service-Oriented Architectures (SOA). A Social Marketing Approach was rated as the least mature in all three indices of desirability, achievability, and affordability, but it offers some unique methods to increase reuse frequency. Barriers to the success of the Social Marketing and Process-Based Approaches are rated as equal in difficulty.

Although these approaches were defined and evaluated in such a manner as to be separately executable, the study team's consensus was that all three approaches had merit, were synergistically combinable, and should be pursued together in the next phase of the LVC Architecture Roadmap Implementation Project in a twenty (20) staff-month effort.

Gateways and Bridges

Objective and Major Subtasks

Objective: Provide a highly-customizable and well-documented set of gateway products to the LVC user community

Major Subtasks:

- Survey and Characterize Existing Gateway and Bridges
- Prepare Execution Plan for Common Gateways and Bridges

Key FY09-Funded Accomplishments

- Developed detailed characterization of 11 existing gateways according to 12 functional and operational capability areas [Ref. 7]
- Identified capability gaps (or absence thereof) across all capability areas [Ref. 7]
- Delineated four possible alternative implementation strategies (Status Quo, Inform, Enhance, and Create), and recommended the Enhance strategy for implementation [Ref. 8]

FY09-Funded Deliverables

[7] Common Gateways and Bridges Characterization Report (May 2010) [JHU/APL NSAD-R-2010-031]

[8] Common Gateways and Bridges Execution Plan (June 2010) [JHU/APL NSAD-R-2010-049]

Consolidated Executive Summaries

Common Gateways and Bridges Characterization Report (May 2010) [JHU/APL NSAD-R-2010-031]

The Live-Virtual-Constructive (LVC) Architecture Roadmap (LVCAR) Study developed a vision for achieving significant interoperability improvements in LVC simulation environments. The study recommended several activities intended to reduce the time and cost required to integrate mixed-architecture events.

Three of the key LVCAR Phase I recommendations were to determine whether existing gateway and bridge applications were effective in meeting user requirements, whether improvements in gateway/bridge capabilities were necessary to address identified gaps, and how these improvements could be best implemented to maximize the Department of Defense (DoD) return on investment. The term "bridge" in this context refers to intelligent translators that link together enclaves of simulations that use the same underlying simulation architecture. A "gateway" is also an intelligent translator but is designed to link simulation enclaves that use dissimilar architectures. To address this recommendation, the LVCAR Phase II Common Gateways and Bridges task began with three major activities: performing gateway and bridge literature research, compiling the team's gateway and bridge usage and development experience, and developing formal gateway and bridge operation terminology. With this starting point, the team was able to create an initial delineation of gateway and bridge capabilities based on its members' background and experience.

At this point in the project, it became clear that the distinction between "gateway" and "bridge" was moot from a development and usage standpoint. For that reason, the term "gateway" is used for both types of applications in this document. Starting with the initial delineation of capabilities, the team compiled a Gateway Capabilities Matrix Template. This template allowed the team to create two structured questionnaires. The first was for commercial and government-funded gateway developers, for which an on-line web interface was provided. The second was for site visits to users of gateways within DoD, mainly selected from large-scale exercise coordinators for the United States (US) military services and joint operations. Although the questionnaires were written for different audiences, they were written in a parallel fashion that allowed a correlation of answers across the two audiences. The background literature research also aided the team in creating a candidate list of developers and users to fill out the questionnaires.

The characteristics investigated were in the following twelve areas: Simulation Data Exchange Model Translations, Simulation Data Exchange Model Behaviors, Architecture Translations, Architecture Behaviors, Exercise Management Behaviors, User Interface, Performance, Operation Modes, Extension Modes, Platform Support, Documentation and Support, and Maturity. Each of these characteristic types was codified in a set of terms of reference and explained to the developers and users so that their answers would strongly correlate.

In addition to the developer questionnaire and the gateway user site visits, the team also held a one-day workshop, the "LVCAR Common Gateways and Bridges Workshop," to present the findings of those questionnaires. The workshop was held at the Virginia Modeling, Analysis, and Simulation Center (VMASC) in Suffolk, VA, on March 4, 2010. This workshop not only allowed the team to verify its results with the community, but it also allowed additional feedback and data gathering. This additional information has been included in the data that is presented in this document.

At the completion of the data gathering and analysis aspects of this task, eleven gateway characteristic assessment points have been identified. Listing them in order of most agreement between developers and users to least agreement produces the following delineation.

1. A capability gap in Simulation Data Exchange Model (SDEM) Translations does not exist.

2. A capability gap in Architecture Translations does not exist.

3. A capability gap in Architecture Behaviors does not exist.

4. No capability gaps were identified for Operation Modes.

5. In regard to Support, the provided support generally meets user needs.

6. In terms of Gateway Performance, users and developers provided consistent responses. When asked to provide the critical performance characteristics, the top three characteristics for both developers and users matched perfectly.

7. Overall, regarding Exercise Management Behaviors, the match between developer supplied capabilities and user-needed capabilities is strong.

8. In regard to Platform Support, there did not appear to be any gaps, as there seem to be two primary platforms in use by the community at this time.

9. There are no identified capability gaps in the User Interface capabilities; however, further research is needed to determine if there is a capability gap for remote control of multiple gateways.

10. In regard to Extension Modes, most users expressed a need to have the capability to extend the functionality of developer-provided gateways.

11. In regard to SDEM Behaviors, there is a disconnect between the developer-indicated capability and the user expressed needs, particularly in the areas of dynamic information translation where one SDEM requires it and the other does not, and supporting differing behaviors between SDEMs. As for the latter, about half of the users indicated a need for this capability, but no developers reported providing this capability. This is either an actual gap in the capability required by the user or a misunderstanding of the capability by either the users and/or developers.

Despite the apparent lack of gateway capability gaps, there was wide agreement at the 4 March workshop that there are several potential improvements that can be made that will lower the technical and cost risks generally associated with the use of gateways. Such improvements, along with team recommendations, will be forthcoming in a later deliverable. However, the information provided in this document provides the necessary foundation for these recommendations by articulating the current state of gateway usage within the DoD.

Common Gateways and Bridges Execution Plan (June 2010) [JHU/APL NSAD-R-2010-049]

The Live-Virtual-Constructive Architecture Roadmap (LVCAR) Study developed a vision for achieving significant interoperability improvements in live, virtual, or constructive (LVC) simulation environments. The study recommended several activities intended to reduce the time, effort, and cost required to integrate multi-architecture events.

Three of the key LVCAR Phase I recommendations were to determine whether existing gateway and bridge1 applications were effective in meeting user requirements, whether improvements in gateway/bridge capabilities were necessary to address identified gaps, and how these improvements could be best implemented to maximize the Department of Defense (DoD) return on investment (ROI). To address this recommendation, the LVCAR Phase II Common Gateways and Bridges task began with three major activities: performing gateway and bridge literature research, augmenting this research with the team's own gateway and bridge usage and development experience, and developing formal gateway and bridge operation terminology. With this starting point, the team was able to create an initial delineation of gateway/bridge2 capabilities.

Starting with this initial capability description, the team compiled a Gateway Capabilities Matrix Template. This template allowed the team to create two structured questionnaires. The first was for commercial and government-funded gateway developers, for which an online web interface was provided. The second was for site visits to users of gateways within DoD, mainly selected from large-scale exercise coordinators for the United States military services and joint operations. Although the questionnaires were written for different audiences, they were written in a parallel fashion that allowed a correlation of answers across the two audiences. The background literature research also aided the team in creating a candidate list of developers and users to fill out the questionnaires. In addition to the developer questionnaire and the gateway user site visits, the team also held a one-day workshop, the "LVCAR Common Gateways and Bridges Workshop," to present the findings of those questionnaires. The workshop was held at the Virginia Modeling, Analysis, and Simulation Center (VMASC) in Suffolk, VA, on 4 March 2010.

At the completion of the data gathering and analysis aspects of this task, the gateway characteristic assessment points were identified and documented in the project's initial deliverable, the "Live-Virtual-Constructive Architecture Roadmap Implementation Common Gateways and Bridges Characterization Report."

As indicated in that report, there is wide agreement that there are several potential improvements that can be made to lower the technical and cost risks generally associated with the use of gateways. Such improvements, along with the team recommendation, are presented in this document as three strategies for execution: *informing* the community as to gateway existence and capabilities; *enhancing* the efficiency and effectiveness by which existing and future gateway products are applied; or *creating* new gateways components or systems. For completeness, the team also included a *status quo* strategy, which describes the impact on the DoD modeling and simulation communities if no action is taken. Below is a description of these strategies.

The first strategy presented involves taking a *laissez-faire* approach to the growth and maturation of the development and use of gateways in the modeling and simulation community, aptly named the "Status Quo" strategy. This strategy acts as a base case for the other strategies. It describes what the team determined as the outcome of not taking any action and allowing the current market forces to continue shaping the industry. The immediate benefit of using the Status Quo strategy is the lack of any new

requirements for DoD investment and the fact that it minimizes any potential disruption to the existing LVC community.

The second strategy ("Inform") focuses on educating the user community about existing gateway availability and capabilities. This knowledge would assist potential gateway users in making better-informed decisions when considering use of a gateway. Understanding what gateways are available and what capabilities they have could reduce proliferation of gateways by promoting reuse of existing products vice building additional one-off gateway solutions. The execution of this strategy involves a number of education options, such as workshops, tutorials, and training courses.

The third strategy ("Enhance") incorporates several of the fundamental elements defined in the previous strategy but extends the purely educational focus with several products intended to make more effective use of the gateway capabilities that exist today. Examples of products identified in this strategy include a Common Gateways Description Language (CGDL) to allow gateway capabilities to be described in a machine-readable form, a set of Gateway Performance Benchmarks (GPBs) that would provide a common way of assessing the relative ability of competing gateways to provide needed capabilities, and a Common Gateway Configuration Model (CGCM) that would provide a standard means of initializing, tailoring, and configuring gateways. Widespread adoption of these products in the LVC community will result in users making much better choices as to the gateway products they use in their applications, and will also assist users with how to best employ these products to minimize technical and cost risks to their projects.

The fourth strategy ("Create") is focused on creating new capabilities to meet the gateway needs of users. These new capabilities can range from relatively minor extensions or enhancements to existing gateways to whole new gateways (or gateway capability libraries) to address capability gaps and provide users with a common interface and common configuration processes/tools. While representing the most expensive option, it also provides the highest potential return. However, achieving this return is dependent on the degree of market penetration that is achieved with new gateway products.

Of the four strategies, the team recommended that the "Enhance" strategy be executed, given that it has the highest perceived ROI. A project plan is presented that details the duration and dependencies of the various tasks in this strategy and estimates the level of effort for execution of this strategy.

[1] The term "bridge" in this context refers to intelligent translators that link together enclaves of simulations that use the same underlying simulation architecture. A "gateway" is also an intelligent translator, but it is designed to link simulation enclaves that use dissimilar architectures.

[2] It became clear during the team's preliminary investigations that the distinction between "gateway" and "bridge" was insignificant from a development and usage standpoint. Both are used for translating between architectures; the difference between a gateway and bridge was primarily a matter of how the system was configured rather than how the system was coded. For that reason, the term "gateway" is used for both types of applications in this document. This terminology was accepted by all of the developers and users with which the team interacted.

Architecture Convergence

Objective and Major Subtasks

Objective: Examine the technical issues and risks related to architecture convergence, and develop an evolutionary strategy to achieve convergence

Major Subtasks:

- Parse Existing Architectures and Produce Reference Model
- Assess Existing Architectures, including
  - Distributed Interactive Simulation (DIS)
  - High Level Architecture (HLA)
  - Test and Training Enabling Architecture (TENA)
  - Common Training Instrumentation Architecture (CTIA)
- Investigate and Evaluate Convergence Options

Key FY09-Funded Accomplishments

- Documented characteristics of existing architectures [Ref. 9]
- Developed a design concept for an implementation of converged services, technically acceptable to team-member subject matter experts (SMEs) from each of the four existing architectures [Ref. 10]
- Delineated the services to be included in a Common Simulation Infrastructure (CSI) to support convergence [Ref. 10]
- Identified three potential courses of action (COAs) for incremental development of the converged capability, and estimated the return on investment (ROI) over time for the recommended COA [Ref. 11]

FY09-Funded Deliverables

[9] Legacy Architectures Reference Model (November 2009) [JHU/APL NSAD-R-2009-220]

[10] Convergence Defining Architecture Characteristics and Activities (February 2010) [JHU/APL NSAD-R-2010-016]

[11] Convergence Final Report (June 2010) [JHU/APL NSAD-R-2010-047]

Consolidated Executive Summaries

Legacy Architectures Reference Model (November 2009) [JHU/APL NSAD-R-2009-220]

The Live-Virtual-Constructive (LVC) Architecture Roadmap (LVCAR) Study developed a vision for achieving significant interoperability improvements in LVC simulation environments. The study recommended activities proposed to lower the time and cost required to integrate mixed architecture events by building better bridges between the legacy architectures and making the architectures more compatible. An LVCAR Convergence Team (LVCAR-CT) has explored converging the current architectures. The recommended approach evolves each architecture to meet the needs of users while favoring common techniques and solutions. Rather than make the current High Level Architecture (HLA) like the current Test and Training Enabling Architecture (TENA), the goal is to make future HLAs more like future TENAs. Subject matter experts (SMEs) from each architecture participated together on the LVCAR-CT. Each SME provided existing documentation resources and identified where in the documents to extract the key services and tools. The LVCAR-CT met to discuss these artifacts and agree on a framework of common constructs through which to view them. Before identifying the changes needed to encourage this evolutionary convergence, the LVCAR-CT needs a common understanding of the services and tools needed to meet the needs of users. This report documents that understanding.

While the architectures do not do the same thing, their conceptual foundations are based on common ideas. The architectures serve to connect specialized software programs together, providing some insulation between the programs to loosen their coupling. Common goals include facilitating LVC software application reuse, increasing software portability, fostering network topology independence, and off-loading common functionality. The reuse goal for each architecture is in the selection of software applications that represent the real players (in a live context) or the simulation (in a virtual or constructive context). These applications may be directly connected to military systems, particularly command and control systems, using military system interfaces. Individual architectures tailor these common constructs in several ways. They expand the scope of architectural definitions, mandating tool or scenario solutions, for example. The tradeoff between user needs for broad design freedom and interoperability is reflected in this tailoring. Providing broad design freedom requires that LVC software application designers make potentially non-interoperable choices. The other tailoring technique is the insertion of intermediate layers between the LVC software applications and the infrastructure and support services. The intermediate layers abstract the provided services to allow more generic implementations of the software applications. The architecture provides means to implement these intermediate layers, either directly or by standard interfaces, to save users effort while providing acceptable interoperability.

A common theme with each of the architectures is the addition, either by an architecture distributor, by an architecture vendor, by third parties, or developed by end users, of services and support tools. Section 4 describes the services provided by the architectures presented in the previous section. Some of the architectures have little explicit notion of a service (i.e., DIS) but service analogs are discussed. Section 5 presents an overview of the support tools available for each of the four architectures we examined.

The LVCAR-CT has established an independent view of the current architectures. The next step is to determine what actions lead to convergence. The vision is that in 2015, new versions of Common Training Instrumentation Architecture (CTIA), Distributed Interactive Simulation (DIS), HLA, and TENA will come out that incorporate the results of the Convergence Initiative. The LVCAR-CT work does not stand alone. In particular, many preconditions, which are being pursued as part of related tasks, are necessary to achieve this vision.

Convergence Defining Architecture Characteristics and Activities (February 2010) [JHU/APL NSAD-R-2010-016]

The Live-Virtual-Constructive (LVC) Architecture Roadmap (LVCAR) Study developed a vision for achieving significant interoperability improvements in LVC simulation environments. The study recommended activities proposed to lower the time and cost required to integrate multi-architecture events by building better bridges between the legacy architectures and making them more compatible with each other. An LVCAR Convergence Team (LVCAR-CT) has explored converging the current architectures. The recommended approach evolves each architecture to meet the needs of users while favoring common implementation techniques and solutions. Rather than make the current High Level Architecture (HLA) like the current Test and Training Enabling Architecture (TENA), the goal is to make future HLAs more like future TENAs. Subject matter experts (SMEs) from each architecture participated together on the LVCAR-CT. Each SME provided existing documentation resources and identified where in the documents to extract the key services and tools. The LVCAR-CT met to discuss these artifacts and agreed on a framework of common constructs through which to view them.

The LVCAR-CT has established an independent view of the current architectures. The next step was to determine what actions lead to convergence. The vision is that in 2015, new versions of the Common Training Instrumentation Architecture, Distributed Interactive Simulation, HLA, and TENA will come out that incorporate the results of the Convergence Initiative. The LVCAR-CT work does not stand alone.

In particular, many preconditions, which are being pursued as part of related tasks, are necessary to achieve this vision.

This report describes the converged architecture envisioned by the LVCAR-CT in terms of how it would execute in a multi-architecture event. This converged execution contains: (1) simulations that need not be aware that multiple architectures are in use, (2) parts of the support infrastructure of the legacy infrastructures, and (3) a common shared library for communication. The LVCAR-CT selected this concept because it requires no changes to the simulations (which are the area of greatest Department of Defense modeling and simulation investment). As a result, changes under this proposed solution impact only a few infrastructure providers and require significantly less investment to achieve convergence.

Convergence Final Report (June 2010) [JHU/APL NSAD-R-2010-047]

The Live-Virtual-Constructive Architecture Roadmap (LVCAR) Study developed a vision for achieving significant interoperability improvements in live, virtual, and constructive (LVC) simulation environments. The study recommended activities proposed to lower the time and cost required to integrate multi-architecture events by building better bridges between the legacy architectures and making them more compatible with each other. An LVCAR Convergence Team (LVCAR-CT) has explored converging the current architectures. The recommended approach evolves each architecture to meet the needs of users while favoring common implementation techniques and solutions. Rather than make the current High Level Architecture (HLA) like the current Test and Training Enabling Architecture (TENA), the goal is to make future HLAs more like future TENAs. Subject matter experts (SMEs) from each architecture participated together on the LVCAR-CT. Each SME provided existing documentation resources and identified where in the documents to extract the key services and tools. The LVCAR-CT met to discuss these artifacts and agreed on a framework of common constructs through which to view them.

The LVCAR-CT has established an independent view of the current architectures. The next step was to determine what actions lead to convergence. The vision is that in 2015, new versions of the Common Training Instrumentation Architecture, Distributed Interactive Simulation, HLA, and TENA will come out that incorporate the results of the Convergence Initiative. The LVCAR-CT work does not stand alone. In particular, many preconditions, which are being pursued as part of related tasks, are necessary to achieve this vision.

This report describes the converged architecture envisioned by the LVCAR-CT in terms of how it would execute in a multi-architecture event. This converged execution contains (1) simulations that need not be aware that multiple architectures are in use, (2) parts of the support infrastructure of the legacy infrastructures, and (3) a common shared library for communication. The LVCAR-CT selected this concept because it requires no changes to the simulations (which are the area of greatest Department of Defense modeling and simulation investment). As a result, changes under this proposed solution impact only a few infrastructure providers and require significantly less investment to achieve convergence.

Construction of software to gradually evolve legacy infrastructures and achieve convergence involves several years of effort. Based on initial return on investment calculations, substantial returns are available in the out years.