Web Enabling HLA Compliant Simulations to Support Network Centric Applications

Katherine L. Morse, Ph.D.
David L. Drake
Ryan P.Z. Brunton
SAIC
10260 Campus Point Drive, MS C3
San Diego, CA 92121
(858)826-6728/(858)826-3429
morsek@saic.com, drakedavid@saic.com, bruntonr@saic.com

Web Enabling HLA Compliant Simulations to Support Network Centric Applications

Katherine L. Morse, Ph.D.
David L. Drake
Ryan P.Z. Brunton
SAIC
10260 Campus Point Drive, MS C3
San Diego, CA 92121
(858)826-6728/(858)826-3429
morsek@saic.com, drakedavid@saic.com, bruntonr@saic.com

Abstract

The Extensible Modeling and Simulation Framework (XMSF) is defined as a modeling-&-simulation-tailored set of self-consistent standards, processes and practices employing a set of web-based technologies and services to enable a new generation of Internet-distributed applications to emerge, develop and interoperate. One of the earliest XMSF uses web services to web enable the Defense Modeling & Simulation Office (DMSO)/SAIC High Level Architecture (HLA) Runtime Infrastructure (RTI) enabling communication between federates in an existing federation. As a result, existing federates and federations can be web enabled rapidly, in some cases just by relinking the federate with the Web Enabled RTI libraries instead of the standard RTI libraries. The resulting federate connects to a web server over the Internet, communicating with a federation that is unaware that the federate is not local. The Web Enabled RTI is an excellent example of the application of web technologies to the problem of making modeling and simulation capabilities available in a network centric environment, supporting the operational warfighter with network centric modeling and simulation tools.

1 Introduction

The EXtensible Modeling and Simulation Framework (XMSF) [1] provides a framework which allows both Department of Defense (DoD) and non-DoD Modeling and Simulation (M&S) projects to take advantage of Web-based technologies. Such a framework aids M&S applications to interoperate, as well as enables M&S development. In the report "Extensible Modeling and Simulation Framework (XMSF): Challenges for Web-Based Modeling and Simulation," a number of suggestions were put forth to address the requirements of developing a web-enabled collection of simulations.

In this paper we describe the web technology standards, data standards, and process used to produce the Web Enabled RTI, and our successes with employing it in federations. We also describe our ongoing efforts to add security mechanisms to the Web Enabled RTI. Security mechanisms were not an original feature of the HLA, although significant analysis of the problem was performed. Our current work with the Web Enabled RTI includes the addition of identification and authentication mechanisms based on web standard technology. Integration of these mechanisms

will allow users to connect to unclassified federations over the Internet via a web browser or lightweight client.

The remainder of the paper is organized as follows. Section 2 provides some background on XMSF as a whole. A description of the Web Enabled RTI (WE RTI) is provided in section 3. Section 4describes the application of the WE RTI to the development of a lightweight C2 viewer. Future work is outlined in section 5.

2 XMSF Background

XMSF[1] is defined as a modeling-&-simulation-tailored set of self-consistent standards, processes and practices employing a set of web-based technologies and services to enable a new generation of internet-distributed applications to emerge, develop and interoperate.

The precepts of XMSF are:

- Web-based technologies applied within an extensible framework will enable a new generation of modeling & simulation (M&S) applications to emerge, develop and interoperate.
- Support for operational tactical systems is a missing but essential requirement for such M&S applications frameworks.
- An extensible framework employing Extensible Markup Language (XML)-based languages can provide a bridge between forthcoming M&S requirements and open/commercial web standards, while continuing to support existing M&S technologies.
- Compatible and complementary technical approaches are now possible for model definition, simulation execution, network-based education and training, network scalability, and 2D/3D graphics presentations.
- Web approaches for technology, software tools, content production and broad use provides best business cases from an enterprise-wide (i.e. world wide) perspective.

Because XMSF is a framework rather than an architecture, it will be comprised of profiles rather than specifications, where a profile is defined as:

[1] This work was funded by DMSO under GSA contract GS-35F-4461G delivery order T0902BH0769.

- A tailoring of the set of selected standards (e.g. tailoring of network protocol standards)
- Data and metadata standards
- Recommendations and guidelines for implementation (e.g. composability guidelines, recommended technologies, technology application guidelines, and recommended hardware configuration)

Our first exemplar employs the Simple Object Access Protocol (SOAP) [2] and the Blocks Extensible Exchange Protocol (BEEP) [3] protocol standards, a remapping of the HLA RTI API calls to XML (potentially a data standard), and a process implementation that is consistent with the DMSO/SAIC RTI.

3 The Web Enabled RTI

The goal of the WE RTI is to enable a simulation to communicate with an HLA RTI [4] through web-based services. The long-term goal is to be able to have multiple federates[2] that are able to reside as web services on a Wide Area Network (WAN), permitting an end-user to compose a federation[3] from a browser. To be able to meet this long-term goal, the supporting federates will need to be configurable, instantiated, and monitored by the end-user. This will require capabilities that go outside the normal operation of an RTI, and will require an additional web service layer that controls these administrative activities.

Initially we built a prototype HLA federation using XMSF compliant Web Services, in this case SOAP and BEEP, for communication between two HLA-compliant federates. We created SOAP-formatted RTI interfaces employing the BEEP communication layer. These RTI interfaces are consistent with the Java bindings for the DMSO/SAIC RTI. BEEP allows bi-directional calls through the interface, enabling Federate Ambassador call backs. By relinking one federate with these interfaces, we have taken the initial steps to making this federate callable as a Web Service. This approach also enables encapsulation of non-reentrant RTI libraries, permitting multiple instances of federates as Web Services.

3.1 Applying SOAP and BEEP

SOAP is an XML-based protocol for exchanging structured data and remote procedure calls. SOAP provides a framework describing the content and processing directives for a message, a set of conventions for making remote procedure calls and encoding their responses, and an extensible encoding scheme that allows for the encoding of user-defined data types. A typical SOAP message consists of an Envelope, which encapsulates a single SOAP packet, an optional Head, which contains processing and routing instructions, and a Body, which contains the actual message content. The advantages of using SOAP as our protocol for remote procedure invocation are many. First, it is a truly cross-platform and cross-language solution. By using SOAP we have not tied any

---

[2] A federate is a member of a HLA federation. All applications participating in a federation are called federates. In reality, this may include federate managers, data collectors, live entity surrogates simulations, or passive viewers. [5]

[3] A federation is a named set of interacting federates, a common federation object model, and supporting RTI, that are used as a whole to achieve some specific objective. [5]

component to a particular operating system or programming language. Second, it is a human readable data format, making development and debugging simpler. Finally, it allows us to set and release a standard XML schema for the HLA mapping so that third parties can develop fully compatible libraries.

BEEP is an application layer protocol for designing other application layer protocols. The BEEP core libraries provide a mapping of the basic packet structure onto TCP, a set of security protocols such as Simple Authentication and Security Layer (SASL) and Transport Layer Security (TLS), and an extensible architecture for defining their own protocol on top of BEEP. The communication pattern is simple and lightweight. A client connects to a server process and requests one or more Profiles, where a Profile is a user-defined application protocol. If the server recognizes at least one of the Profiles, a Session, roughly equivalent to a connection, is instantiated. Each Session can then support multiple Channels, where a Channel encapsulates all communications for a given Profile. Using this model we were able to separate RTIAmbassador calls from FederateAmbassador callbacks by assigning each a separate Profile for which we opened a separate Channel. Figure 3-1 illustrates the architecture of our exemplar.

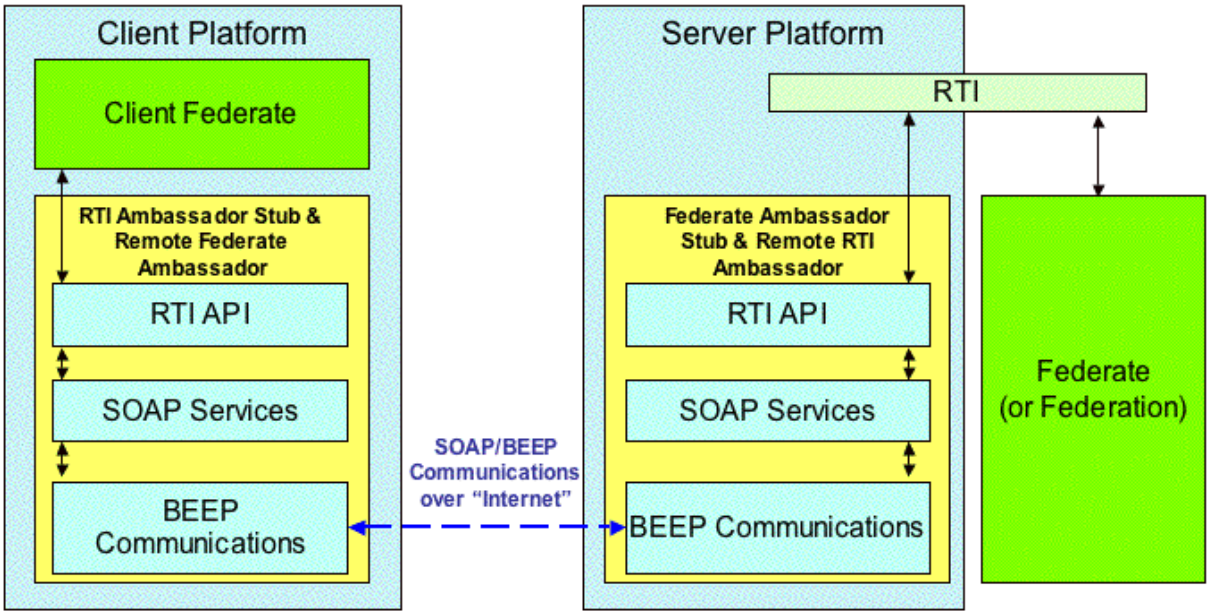

Figure 3-1. Web-Enabled RTI Architecture

The process model for one possible federation using the WE RTI is as follows:

1. The user initiates the server side federate
    a. The server side federate starts the Federation
2. The user initiates the client side federate
    a. The client side federate spawns an RTI Ambassador Stub
    b. The RTI Ambassador Stub spawns a Federate Ambassador
    c. The Federate Ambassador spawns a Federate Ambassador Stub

d. The stubs makes remote SOAP calls to initialize the remote RTI and Federate Ambassadors

3. The client side federate RTI calls go through the RTI Ambassador Stub, out to remote RTI Ambassador via SOAP, which passes the calls to the RTI
4. Communications from the RTI pass to the Remote Federate Ambassador, to the Federate Ambassador Stub via SOAP, and pass to the client side federate.

3.2 What's Important About This Exemplar

The Web Enabled RTI has already been applied in three federations:

- The Defense Threat Reduction Agency's (DTRA) existing Weapons of Mass Destruction Operational Analysis federation
- An integrated HLA-Advanced Distributed Learning (ADL) circuit design training application
- The XMSF Distributed Continuous Experimentation Environment (DCEE) Viewer that participated in JFCOM J9's initial DCEE integration event

As has been demonstrated in the past, all of these federations can be run on a LAN. However, the implications of running federations via web services are enormous. A legacy simulation may be made available without moving its dedicated hardware or trying to create a new installation on potentially rare hardware[4], both very expensive propositions. The simulation can stay home based with its technical support and configuration management. There's no switching between supporting different federations at different times. The positive impact on lifecycle costs and availability can be significant.

This exemplar demonstrates that we are capable of implementing bi-directional communication initiation over the Web using SOAP and BEEP. This approach is superior to http's uni-directional initiation that makes it unsuitable for supporting simulation communication patterns. This technology enables existing HLA compliant federates to be integrated easily over the Internet, including through most firewalls with minimal reconfiguration! The concept of placing the simulation in the DMZ[5] also aids in making it a web service. Furthermore, it demonstrates web service wrapping of existing architectures, which means that this same approach can also be applied to DIS, ALSP, etc.

4 The XMSF DCEE Viewer

The Distributed Continuous Experimentation Environment (DCEE), managed by the J9, U.S. Joint Forces Command (JFCOM), has established a framework of common terminology for the information to be exchanged between components using an enhancement of the Real-time Platform Reference (RPR) Federation Object Model (FOM). Although the DCEE actually uses HLA as the

---

[4] Consider just the cost of moving hardware and people to support Millennium Challenge 02.

[5] The demilitarized zone (DMZ) of a network is one of the connections to a corporate firewall. Computers with controlled external access that will be connected to the Internet, e.g. web servers, are typically placed in the DMZ.

technical backbone, the concept is open for extensions to emerging solutions. Use of XML, standardized tag-sets, web services, and web technology is part of the general concept of DCEE.

The XMSF Partners have successfully demonstrated the benefits of XMSF in the DCEE with the XMSF DCEE Viewer (XDV)[6]. XDV is a web-based, low-cost viewer for DCEE based events. Our concept for XDV is simple: every eligible stakeholder interested in observing execution of the ongoing experiment can log into the federation and use the viewer software to follow the actual experiment. The necessary software was installed on COTS PCs connected via Internet protocols, allowing eligible stakeholders to follow the experiment from wherever they were located. XDV was demonstrated both within the DCEE, and between the DCEE and the Virginia Modeling, Analysis & Simulation Center (VMASC) Battle Lab.

4.1 Objectives

The concept for a web-based, low-cost viewer is simple: every eligible stakeholder interested in observing execution of the ongoing experiment can log into the federation and use the viewer software to follow the actual experiment. The necessary software must execute on a COTS PC over the Internet, Defense Research Engineering Network, or classified DoD SIPRNET.

The task is similar to creating the Common Relevant Operational Picture (CROP) during a joint operation, in which the various information pieces of the C4I systems of the services have to be unified. However, there are two key differences between the challenges in creating a CROP and in displaying the experiment during execution:

- There is no need for data fusion in experiment monitors. Other than in the C4I domain, the data from the participating simulations are "ground truth" data, and variances are caused by aggregation/disaggregation procedures or variances within the resolution of the models, i.e., variations are reproducible and of interest to those monitoring the experiment.
- Dynamic aspects are more important in experiment monitoring than in the CROP.

The XMSF team successfully demonstrated that, via rapid prototype development, it was possible under XMSF to develop the viewer in a two-month period. The viewer supports both 2D unit level and 3D unit level views. The XDV has the following properties:

- The XDV is unclassified Internet/web protocol-based software, although it may be used to view classified data.
- An eligible stakeholder can download the software to his COTS PC or COE compliant system and execute it after installation.
- The XDV can display all information exchanged within the DCEE, i.e., all information elements comprised in the enhanced Millennium Challenge 02 FOM are mapped to at least one element of the XDV Graphical User Interface (GUI).
- The XDV is individually configurable, i.e., the user can decide what predefined entities to view.

---

[6] This work was performed in conjunction with Old Dominion University, Naval Postgraduate School and General Dynamics Information Systems under contract N00140-01-C-H001, ERS0206.

- Because the XDV is a passive subscriber, its execution does not impact the overall execution of the experiment.

The time delay between the occurrence of an event in the experimentation and its display on the XDV used by the stakeholder was demonstrated to be sufficiently small as to be negligible for human users.

4.2 Design and Architecture

Figure 4-1 illustrates the logical architecture of the XDV.

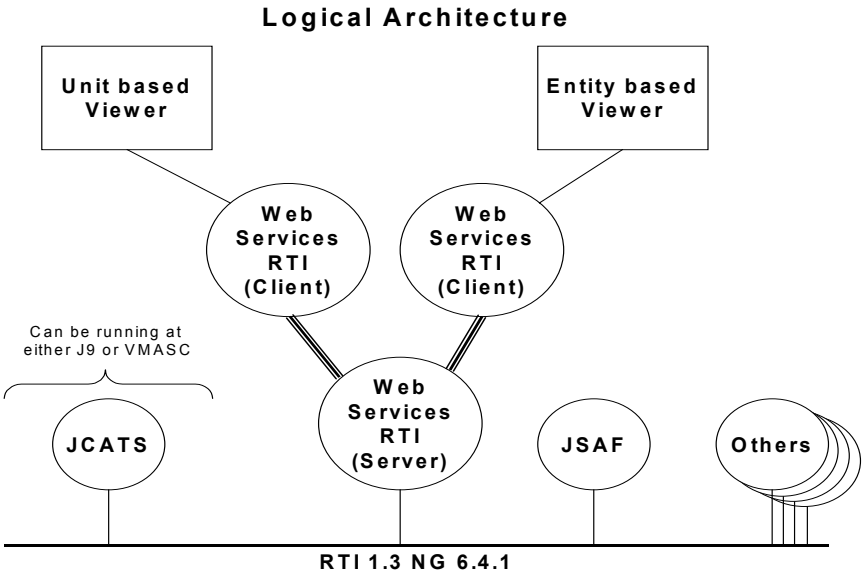

Figure 4-1. XDV Logical Architecture

Figure 4-2 illustrates the physical software architectures in which the XDV was tested. Note that the XDV was tested in all four permutations of installation of the viewer and server in the DCEE and in the VMASC Battle Lab.

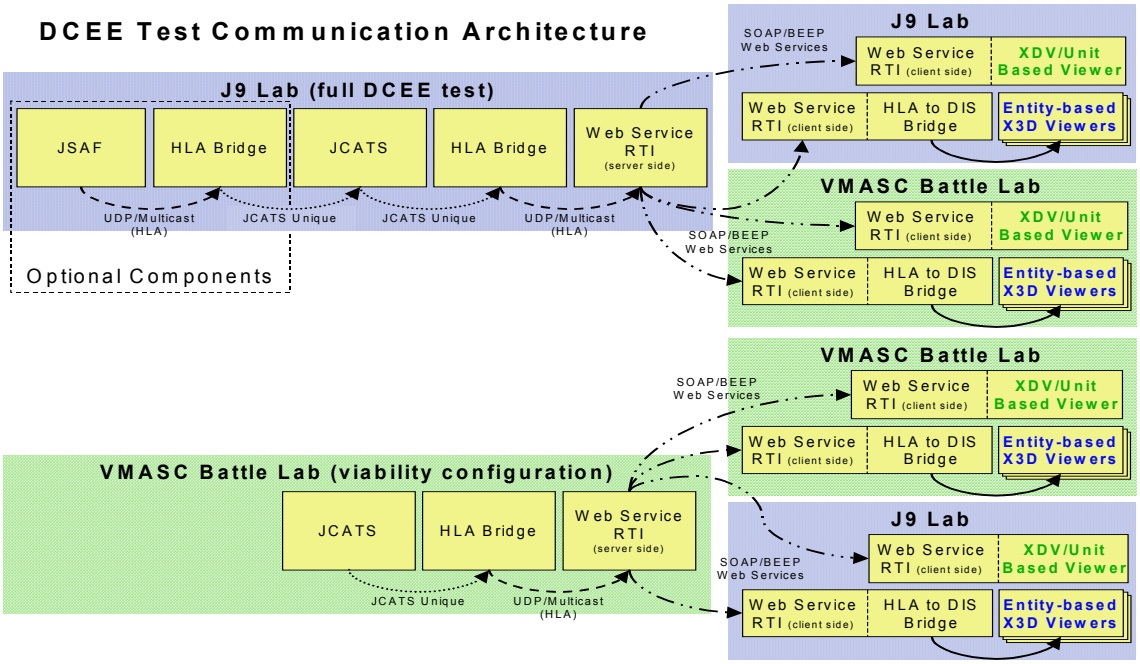

Figure 4-2. XDV Deployment Architecture

Although the viewer was designed to support only 500 entities, it was able to participate in a federation run at VMASC with 800 entities, and another federation run at DCEE with over 19,000 entities.

5 Future Work

The XDV team is now extending these concepts to a viewer/controller called Extensible Command and Control Interface (XC2I). Not only will users be able to view the battlespace, but they will also be able to exercise limited control over entities modeled by the Joint Semi-Automated Forces (JSAF) simulation.

XC2I will also represent the first time we have integrated security mechanisms into an XMSF environment. The security solution we are building will not be used for classified data unless it is within an outer "system-high" network. Rather, it will be used to protect sensitive-but-unclassified data and also to partition the users administratively. Because the user's right to view certain data will be based on his/her role, interest management (IM) [6] will be integrated with access control. We will build an authentication server that takes a user name and password and returns a token identifying the user's role and privileges. The first time the viewer connects to the web server, it will return a list in XML of the IM scope rules available. These rules will be derived from the intersection of the entities available in the federation and the user's access privileges. Figure 5-1 illustrates the preliminary architecture for XC2I.

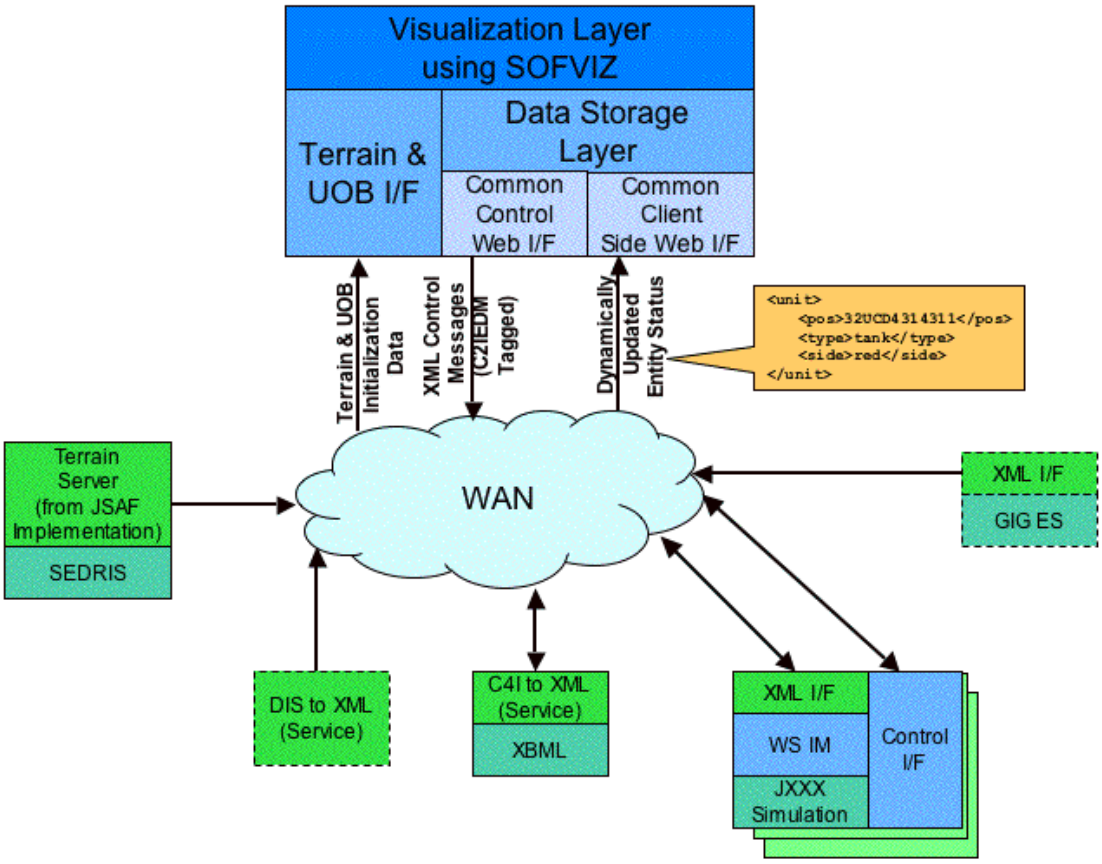

Figure 5-1. XC2I Architecture

XC2I is initially intended to support JFCOM J9's Joint Urban Operations (JUO) experiments. However, the team's goal is to use technologies and standards that will enable it to be applied more broadly as a general purpose C2 viewer/controller. For example, control messages to JSAF will be formatted in Command and Control Information Exchange Data Model (C2IEDM) [7], the data model used by the NATO Data Administration Group (NDAG) for data administration of NATO C2I and by the NATO Multilateral Interoperability Program (MIP) for information exchange between real C4I systems.

6 References

[1] Don Brutzman, Michael Zyda, J. Mark Pullen, Katherine L. Morse, "Extensible Modeling and Simulation Framework (XMSF) Challenges for Web-Based Modeling & Simulation," www.movesinstitute.org/xmsf, October 2002.

[2] IETF, Simple Object Access Protocol (SOAP), http://www.ietf.org/rfc/rfc2396.txt

[3] IETF, Blocks Extensible Exchange Protocol (BEEP), http://www.ietf.org/rfc/rfc3080.txt

[4] Defense Modeling and Simulation Office (1998) "Department of Defense High Level Architecture Interface Specification, Version 1.3," April 2 1998.

[5] Defense Modeling and Simulation Office (1998) "Department of Defense High Level Architecture Framework and Rules, Version 1.3," February 5 1998.

[6] Katherine L. Morse, Lubomir Bic and Michael Dillencourt "Interest Management in Large Scale Virtual Environments," MIT PRESENCE - Teleoperators and Virtual Environments, February 2000.

[7] Andreas Tolk, "A Common Framework for Military M&S and C4I Systems," Proceedings of the 2003 Spring Simulation Interoperability Workshop, March 2003, Orlando, FL.

7 Author Biographies

Dr. Katherine L. Morse is a Chief Scientist with SAIC. She received her B.S. in mathematics (1982), B.A. in Russian (1983), M.S. in computer science (1986) from the University of Arizona, and M.S. (1995) and Ph.D. (2000) in Information & Computer Science from the University of California, Irvine. Dr. Morse has worked in the computer industry for over 20 years, specializing in the areas of simulation, computer security, compilers, operating systems, neural networks, speech recognition, image processing, and engineering process development. Her Ph.D. dissertation is on dynamic multicast grouping for Data Distribution Management, a field in which she is widely recognized as a foremost expert. With Don Brutzman (NPS) and Mark Pullen (GMU), Dr. Morse is one of the founding XMSF partners. She is also a key contributor in the area of HLA-Advanced Distributed Learning (ADL) integration.

David L. Drake is a Program Manager with SAIC. Mr. Drake has 23 years as a computer security professional in computer security design, implementation and evaluation at companies including SAIC, Computer Sciences Corporation, and the MITRE Corporation. At SAIC, he was a senior computer scientist and a manager for the Commercial Security Products Research and Development Division. While at the MITRE Corporation, he was lead developer for the Practical Verification System, a formal specification and automated verification. Mr. Drake received a Bachelors degree in mathematics from State University of New York at Buffalo and did graduate studies there in artificial intelligence. Mr. Drake received additional computer science training at Stanford Research Institute in California, Northeastern University, and Wang Institute in Massachusetts. He has published articles and given speeches on security and risk assessment topics and has a patent pending on the process for enterprise-wide intrusion detection.

Ryan P.Z. Brunton is a Software Engineer with SAIC. He received his B.S. in computer science (2001) from the University of California, San Diego. Prior to completing his B.S. he worked at the MOVES Institute at the Naval Postgraduate School. Mr. Brunton's key areas of expertise are modeling & simulation, object-oriented design and development, and the extreme programming methodology.