2004

Extensible Modeling and Simulation Framework (XMSF) Exemplars in Analytic Combat Modeling

Blais, Curtis
Extensible Modeling and Simulation Framework (XMSF) Exemplars in Analytic Combat Modeling

Curtis Blais
The MOVES Institute
Naval Postgraduate School
700 Dyer Road Rm366
Monterey, CA 94943
831-656-3215
clblais@nps.navy.mil

Keywords:
XMSF, NSS, FAST, XML, Data Interchange, Profiles, Simkit, CombatXXI

ABSTRACT: The Extensible Modeling and Simulation Framework (XMSF) is a composable set of standards, profiles and recommended practices for Web-based modeling & simulation (M&S) (http://www.movesinstitute.org/xmsf/xmsf.html). A number of exemplar prototypes are in progress to demonstrate the ability of XML-based markup languages, Internet technologies and Web Services to enable a new generation of distributed M&S applications to emerge, develop and interoperate, including interoperability with command and control systems. This paper describes work being performed to enhance existing analytical models and tools while serving as proof of concept for XMSF ideas. Key applications include integration of the Naval Simulation System (NSS) with the Simkit open source Discrete Event Simulation library and integration of a variety of models and tools in the Flexible Asymmetric Simulation Technologies (FAST) program. The paper describes work completed, work in progress, and future opportunities.

1. Introduction

The Information Technology community continues to find itself in the midst of rapid technological change, today seen most clearly in the exciting pace of evolution of the World Wide Web and its underlying technologies toward a Service Oriented Architecture offering opportunity for diverse heterogeneous systems to interoperate.

“The Open Services Architecture and the design principles based on Web-services, computational component models, and other service components…will help systems engineers, software providers, and city designers and architects lay the foundations for tomorrow’s information cities, where businesses, consumers, municipalities, schools, hospitals, and other basic service providers are connected to millions of users. The boundaries of any information city over the Internet are potentially limitless…” [1]

The Department of Defense (DoD) is engaged in warfighting and institutional transformation for the new millennium. In parallel, the DoD Modeling & Simulation (M&S) community is working to identify and adopt transformational technologies providing direct tactical relevance to warfighters [2]. In 2002, the Defense Modeling and Simulation Office (DMSO) initiated the Extensible Modeling and Simulation Framework (XMSF) program to encourage application of Web technologies and open standards by military M&S planners, managers, developers, and users. Software systems that composably scale to worldwide scope utilize the World Wide Web, making it evident that an extensible Web-based framework offers great promise to scale up the capabilities of M&S systems to meet the needs of training, analysis, acquisition, and the operational warfighter. By embracing commercial Web technologies as a shared-communications platform and a ubiquitous-delivery framework, DoD M&S can fully leverage mainstream practices for enterprise-wide software development.

1.1 The New Analytic Agenda

One of the principal uses of M&S in the military is to support combat analysis across the spectrum from acquisition to operations. OPNAV N81, the analysis branch of the Department of the Navy, faces a new challenge for analytical combat modeling to address the New Analytic Agenda promoted by the Office of the Secretary of Defense (OSD). The New Analytic
Agenda seeks to transform the way DoD applies Modeling and Simulation (M&S), requiring the services to transition from the traditional Cold War two-front outlook to a multi-polar environment whose key characteristic is uncertainty. Primary focus areas of the Analytic Agenda are:

- Service/Agency Program Objective Memorandum (POM) Development
- Studies, including: Program Budget Review; Defense Planning Guidance (DPG)-directed; Analysis of Alternatives
- Capabilities-Based Future Force Planning
- Future Requirements
- Uncertainty of Potential Threat

The 2001 Quadrennial Defense Review (QDR) defined Capabilities-Based Planning, stating it “shifts the focus of U.S. force planning from optimizing for conflicts in two particular regions – Northeast and Southwest Asia – to building a portfolio of capabilities that is robust across the spectrum of possible force requirements, both functional and geographical” [3]. To support this concept, tools are needed that enable analysts to examine a wide range of variability in Red, Blue, and Green factors, in order to achieve a broad portfolio of military capabilities that will perform robustly in an uncertain future environment and that are linked to Joint Concepts of Operations.

The New Analytic Agenda demands that the M&S community move beyond the scenarios that have been long used as the foundation for decision-making and into a much more complex battlespace influenced by factors that have traditionally been ignored. This requires a new class of M&S capabilities, moving away from monolithic, closed system designs to open, M&S frameworks that permit modular, loosely coupled components to be rapidly integrated to create agile analytical capabilities that can address the variety of missions conducted by today’s warfighters. These tools must be flexible, extensible, scalable to a variety of levels of resolution, re-usable, executable in a desktop/laptop environment, convenient to use, able to exploit the best methods (functionality) available in various domains, not bound to traditional approaches to combat modeling, and able to model future concepts by providing a framework for introducing wholly new concepts of warfare.

To address this challenge, OPNAV N81 recently initiated the World-Class Modeling (WCM) program consisting of a number of complementary studies and development efforts. Among these efforts, N81 tasked the Modeling, Virtual Environments and Simulation (MOVES) Institute of the Naval Postgraduate School (NPS), Monterey, California to investigate application of XMSF concepts to improve Naval analytical modeling capabilities. The goal is to develop and demonstrate a modeling framework using Web services to enable two disparate models, the Naval Simulation System (NSS) and the Army/Marine Corps CombatXXI, to interoperate through a common discrete event simulation engine (Simkit). Application of the integrated models to specific analysis problems posed by N81 will demonstrate that a broader set of scenarios can be represented and assessed more effectively than operating the models in isolation.

Upon completion of the development and experimentation, the source code, framework standards, scenarios and results will be made available to DoD agencies to use and build upon. The goal is to begin creation of an analytical modeling “marketplace of capabilities” where analysts can benefit from selection and application of specific desired capabilities rather than being necessarily bound to packaged, non-extensible functionality that has to be “worked around” to accomplish analyses of interest. Early efforts in the N81 project are described later in this paper to illustrate application of XMSF principles to enhance existing model interoperability.

1.2 Analytical Modeling for Military Operations

A second XMSF exemplar discussed in this paper is DMSO’s Flexible Asymmetric Simulation Technologies (FAST) program and development of the Operations Other Than War (OOTW) Toolbox (see [4] and [5]). Whereas the OPNAV N81 WCM project focus is largely on the acquisition end of the military analysis spectrum, the FAST project represents the operations end of the military analysis spectrum. The goal of the FAST project is to provide an integrated set of combat simulations, databases, and computational tools to military analysts deploying to theater or supporting operations from a reach-back center, with primary focus on tools supporting OOTW mission planning and assessment.

Taken together, the N81 World-Class Modeling and FAST programs serve as a proving ground and early exemplars of the application of XMSF concepts to military analytical modeling efforts. A brief introduction to XMSF and exploration of approaches to defining XMSF Profiles are given in sections 2 and 3, respectively, followed by a discussion of the two projects in sections 4 and 5.
2. Extensible Modeling and Simulation Framework (XMSF)

XMSF is defined as a composable set of standards, profiles and recommended practices for Web-based modeling and simulation [2]. The goal is to enable simulations to interact directly and scalably over a highly distributed network, achieved through compatibility between a web framework and networking technologies. XMSF must be equally usable by human and software agents and must support composable, reusable model components.

We distinguish some of the key XMSF terminology as follows [6]:

- **Web technologies** – the collection of standards and applied methods related to Internet-based computation, such as Hyper-Text Transfer Protocol (HTTP), Extensible Markup Language (XML), Simple Object Access Protocol (SOAP), Transmission Control Protocol (TCP), User Datagram Protocol (UDP), Internet Protocol (IP), and many more.
- **Web-enabled application** – an application that can be executed in the Internet environment and that can make use of Web technologies supporting its execution.
- **Web-based application** – an application that was designed to be executed in the Internet environment and that makes explicit use of Web technologies by design.
- **Web service based application** – a Web-based application that uses Web services as its main means of information exchange, which includes the use of all four protocols XML, SOAP, Web Services Description Language (WSDL), and Universal Description, Discovery and Integration (UDDI).

Interestingly, these definitions do not necessitate that the application, to align with XMSF principles, must be a distributed application or even have a network connection. What they do signify is the set of standards and techniques that must be used in the construction and operation of the software, whether on a single machine or operating on a network. The point is to encourage (if not require) use of those technologies that are enabling rapid advancement of the Internet and the World Wide Web while also creating extended opportunities for interoperability of the application with others. To do so, however, requires clear description of the XMSF “standards, profiles, and recommended practices” so that the M&S community can effectively and efficiently adopt and exploit those capabilities. Before discussing the two exemplar projects, the N81 WCM and FAST OOTW Toolbox, we first consider the challenges of defining practical XMSF Profiles that can inform and enlighten the variety of stakeholders across the military M&S community. This will enable us to later discuss these analytical modeling efforts in light of possible XMSF Profile approaches.

3. XMSF Profiles

The XMSF concept promotes interoperability through the use of open standards, specifically those associated with the World Wide Web, and established practices. To create practical understanding of the application of XMSF precepts to real products, SISO established an XMSF Profiles Study Group in September 2003. The Study Group is working to determine the required scope for XMSF Profiles and to define their structure and application. The Study Group Terms of Reference document [7] states that the specification of XMSF will be in the form of a collection of profiles detailing how to interoperate with XMSF compliant systems. These profiles will enable inter- and intra-domain interoperability. The Study Group has established that at a macro level a profile will consist of:

- Applicable Web technologies and protocol standards
- Applicable data and metadata standards, including a tailoring of the set of selected standards (e.g., tailoring of authentication standards)
- Recommendations and guidelines for implementation
  - Composability guidelines
  - Technology application guidance
  - Hardware configuration recommendations, requirements, and constraints; e.g., network bandwidth, minimum processing capability
  - Software configuration recommendations, requirements, and constraints; e.g., browser support for specific applications
  - Specialization of design methodologies

XMSF Profiles will become formal technical specifications for application of interoperable Web-based technologies enabling composable and reusable modeling and simulation, and facilitating enterprise integration. Furthermore, the Study Group has established the following objectives for XMSF Profiles:
• Provide unambiguous specification of the interfaces and functionality of components of the framework.
• Ensure interoperability between existing and new Web-enabled technologies, both within M&S and in related domains.
• Provide the necessary metadata to facilitate composability and reuse of components across multiple M&S application domains.
• Facilitate development of new applications and services that are functionally interchangeable with existing applications and services.
• Enable development of new applications and services that readily extend functionality for continuous evolution of capabilities.

These translate to the following questions with respect to applications defined by a profile:
• What can I expect it to do?
• How do I physically integrate with it?
• How do I semantically integrate with it?
• How can I build another one?
• How can I build a better one?

The Study Group is proceeding with development of a Concept of Operations describing how each XMSF stakeholder develops, finds, and uses profiles. Stakeholders include Profile Developers, Profile Community/Working Group (XMSF Profile Study Group for now), Profile Users (model developers and integrators), Profile Certifying Authority (not yet established), and End Users (possibly unaware of the use of profiles), Profile Manager, and possibly a Profile Verification and Validation (V&V) Agent. This effort is helping the Study Group participants come to grips with the nature and purpose of XMSF Profiles. To further inform the activity of the group, specific exemplars are needed—much can be learned by trying to describe the profile for a particular application, even before the Study Group has fully specified what a profile consists of.

It is well beyond the scope of this paper to derive a set of XMSF Profiles—the Study Group continues to labor at that effort. What we can do, however, is identify many of the core Web technologies that are established and emerging, and attempt to create a basis for profiling the characteristics of particular applications. This will address part of the definition of XMSF profiles; namely, that a profile consists of: (1) applicable Web technologies and protocol standards and (2) applicable data and metadata standards. At its simplest, then, an XMSF Profile is an identification of Web technologies, data, and metadata standards employed in an application. More fully, profiles define common capability levels needed for user requirements and application support, including specification of mandatory (and optional) standards and recommended practices, recommendations and guidelines for implementation (e.g., composability requirements, recommended technologies, application guidelines, and recommended hardware configuration), and implementation and evaluation metrics to measure conformance and capabilities. Association of profiles with actual applications helps us distinguish features of the applications that support greater levels of interoperability, providing both an appraisal of what an application can do now and an assessment of how it can be modified to achieve higher levels of interoperability in the future, as may be required.

For profiles to successfully enable interoperability their initial content and structure must be agreed upon. As the underlying technologies and standards evolve the profiles and their implementations will need to be upgraded in an iterative fashion to maintain interoperability. Knowing what those technologies are and how they interrelate facilitates evolution of the applications as underlying technologies evolve.

3.1 Levels of Conceptual Interoperability

One of the fundamental defining characteristics of an application employing XMSF concepts is the level of interoperability intended in the design of the application. To this end, Tolk and Muguira introduced a Levels of Conceptual Interoperability Model (LCIM) [8], arguing that “meaningful interoperability on the implementation level requires composability on the conceptual level.” Based on work by Hofmann [9], the model was refined to introduce a pragmatic level above the semantic level to reflect the idea that the receiver of the information needs to understand what to do with the information (pragmatic level) in addition to understanding meaning (semantic level). As currently formulated [10], the LCIM describes the following levels of conceptual interoperability:

• On level 0, no connection is established at all (i.e., no interoperability is intended in the design and implementation of the application).
• On level 1, the technical level, physical connectivity is established allowing bits and bytes to be exchanged. The technical level involves physical connections and network layers.
• On level 2, the syntactical level, data can be exchanged in standardized formats; i.e., the same protocols and formats are supported. Examples
include the High Level Architecture Object Model Template (HLA OMT), Common Object Request Broker Agent (CORBA) Interface Design Language (IDL), Distributed Interactive Simulation (DIS) Protocol Data Unit (PDU), XML Metadata Interchange (XMI), and Web Services Description Language (WSDL).

- On level 3, the **semantic** level, not only data but also its context (i.e., **information**) can be exchanged. The unambiguous meaning of data is defined by common reference models, such as the Command and Control Information Exchange Data Model (C2IEDM).

- On level 4, the **pragmatic/dynamic** level, information and its use and applicability (i.e., **knowledge**) can be exchanged. The applicability of information is here defined in an unambiguous form through such approaches as reference models of processes, Unified Modeling Language (UML), and Discrete Event System (DEVS) specifications.

- On level 5, the **conceptual** level, a common view of the world is established through a system-of-systems wide conceptual model — i.e., an **epistemology** (formalization of the knowledge about a domain) — and the place of the model within the view can be exchanged. This is manifested in common conceptual models using engineering based standards such as UML and DEVS.

The LCIM provides a foundation for distinguishing XMSF applications and can be used to define one dimension of a profile “space.” For example, we can build profiles on the basis of the levels of interoperability; e.g., for the XMSF Technical (Level 1) Profile, we identify the Web technologies, practices, and standards appropriate for connectivity; for the XMSF Syntactical (Level 2) Profile, we identify the technologies, practices, and standards appropriate for exchanging data in standardized formats. Characterizing an application against the LCIM “yardstick” creates an initial understanding of the current (intended) scope of interoperability of that application. Such understanding can guide users in employing the application, integrators in interfacing other systems to the application, and development managers in identifying capabilities and effort required to move the application to higher levels of interoperability, if so desired. A hierarchy of XMSF Profiles must ensure alignment of the various levels of interoperability to complement each other and show how moving up in levels adds to capabilities of the application to interoperate with other applications.

XMSF Profiles on one level are likely to contribute to an interoperability solution for a particular application or set of applications, but they are not likely to be the full solution. A family of XMSF Profiles covering several levels is more likely to be needed to provide the solution for a particular interoperability requirement.

In contrast, the approach taken by the Web3D Consortium in defining profiles for the emerging Extensible 3D Graphics (X3D) standard is identification of various levels and layers of functionality [11]. In our case, we are not describing functionality, per se, but various layers of standards, practices, and techniques that are considered relevant to XMSF applications. For example, the use of XML in applications can vary from simplistic use of tagged data in input/output files, to design of XML schemas and use of the schema to drive XML document reading/writing with automatic validation of document contents, customized XML messaging for run-time transactions, standardized wrapping of XML messages (i.e., SOAP), functional exposure through XML (e.g., XML-RPC), standardized discovery and interactions (e.g., XML/soap/WSDL/UDDI Web services stack), and increasingly sophisticated levels of semantic representation and reasoning; e.g., using Resource Description Framework (RDF), RDF-Schema, Defense Advanced Research Projects Agency Agent Markup Language (DAML) Ontology Inference Layer (OIL), Web Ontology Language (OWL) or OWL-Services (OWL-S). Applications can be built that employ several of these technologies/techniques. XMSF profiles must identify what standards are used, but also how they are used to provide a more complete basis for understanding capabilities of an application for interoperability and Web-enabled to Web-based deployment.

### 3.2 XMSF Profile Space

What characterizes an application as an “XMSF” application? That is, what makes an application an XMSF application as opposed to any other type of application? Is a Web page in HTML with or without Javascript an XMSF application? Why or why not? What if it uses XHTML instead of HTML, is it then an XMSF application? Does the application need to employ Web-based technologies such as XML, Extensible Stylesheet Language for Transformations (XSLT), and others? Certainly at a minimum, association of an XMSF Profile to an application needs to clearly identify these distinguishing characteristics. The following discussion explores various perspectives on characterization of XMSF Profiles. Since the precise
definition of XMSF Profiles is a work-in-progress in the XMSF Profile Study Group, the following is not intended to be exhaustive, but merely enlightening to the various considerations possible.

From the 2002 XMSF Report [2], characteristics of an XMSF application include some or all of the following which may suggest certain aspects of profiles:

- **Use of Standards**:
  - Identifiable set of standards, profiles, and recommended practices used in the application
  - Use of repositories across different levels of abstraction (e.g., 3D models, portable computational models, software agent templates with requested capabilities, stream-specific adaptors/components, exercise simulation management, operational recording of simulated or actual interactions, order of battle)
- **Composable and Extensible**:
  - Composably scales and extensible, using reusable model components
  - Kernel plug-ins supporting extensions and modifications to framework layers
- **Web Technologies**: Commercial web technologies as a shared communications platform and a ubiquitous delivery framework
- **Distributed**:
  - Enabled to interact directly and scalably over a highly distributed network
  - Diverse network channels and transport mechanisms
- **Non-proprietary**: Unconstrained by proprietary technology or legally encumbering patents which might discourage the free, open, *ad hoc* development of interconnected tactical models and simulations
- **Usable by human and software agents**
- **Defined Vocabularies**:
  - Use of XML for representation of data structures
  - XML Schema and XML Namespaces employed for defining and referring to precise vocabularies
- **Engineering Methodology**:
  - Employ object-oriented paradigms and validatable structured data in a language-independent and object-system-independent manner
  - Design patterns map representations and component models from root XML schemas to multiple programming languages and Application Program Interface (API) bindings
  - Software component functionality and interactions documented using the UML
  - Graphical User Interface (GUI) description in language and platform independent manner
- **Security**:
  - Offer utilities that include one or more default encryption algorithms
  - Standard for signing messages and documents
- **Grouping**:
  - Mechanism for defining groups and group membership (dynamic)
  - Group definitions must be able to apply to a single service or span multiple services

Some possible dimensions of the profile space emerge from this characterization of XMSF applications, as identified in the grouping of characteristics above. However, this characterization does not cleanly map to the various Web and Internet technologies that can be employed in implementation of an application. From the discussion of the LCIM and definition of profiles, we can alternatively consider a number of different aspects of profiles based on their use of Web technologies. For example, the following is a partial identification of ways those technologies are employed in applications:

- **Data exchange**:
  - Data expressed in XML with exposed structure in an accompanying Document Type Definition (DTD) or Schema; data transformation using XSLT
  - Messaging:
    - Data interchanged during execution as messages/transactions encoded in XML and parsed/interpreted via XML mechanisms
    - XML data marshaling/serialization
    - XML data compression/decompression
    - XML payload embedded in an XML envelop (e.g., SOAP, Jabber)
  - Data storage:
    - Data stored as XML in documents with an accompanying DTD or Schema
    - Data stored in relational or object-oriented databases with query/response in XML
    - Data stored in native XML database (e.g., Xindice, Tamino)
  - **Internal architecture**:
    - Application of Web services for procedure/method calls
    - Interchangeable execution as single process, multi-process, or distributed application
  - **Presentation/GUI**: 
For a particular level of interoperability (LCIM), this additional information is needed to characterize how that level of interoperability is implemented so that users and developers can be more fully informed of aspects of the application that relate to representation and implementation of data and functionality.

While some characteristics of the applications relate to levels of interoperability in terms of information and operation exchange, security characteristics (e.g., authentication, authorization, encryption/privacy) can quite independently allow or prevent interoperability at any level regardless of the technical, syntactical, semantic, pragmatic/dynamic, or conceptual interoperability that may be technically feasible in the implementation of an application.

Another approach to identification of an XMSF profiling “space” considers the following dimensions:

- **Computation:** e.g., single-process, single-processor; multi-process, single-processor; distributed processing, parallel computation; grid computation; service-oriented; composable
- **Networking:** e.g., communications protocol such as HTTP, Simple Mail Transfer Protocol (SMTP), or File Transfer Protocol (FTP); messaging protocol (UDP, TCP/IP); broadcast or multicast; reliable or unreliable; Quality of Service; local area or wide area network; mobile, wireless
- **Presentation:** e.g., Internet browser; custom GUI
- **Modeling (and Data Representation):** e.g., XML, Namespaces, DTD, XML Schema; static file interchange (XSLT); dynamic data exchange (see Messaging below)
- **Messaging:** XML content; SOAP
- **Semantics:** e.g., RDF, RDF Schema, DAML/OIL, OWL
- **Security:** e.g., Security Assertion markup Language (SAML), Extensible Access Control Markup Language (XACML), Extensible Rights Markup Language (XrML), XML Encryption, XML Digital Signature
- **Management:** e.g., UML, Model Driven Architecture (MDA), XML repositories

Clearly, a number of these dimensions, such as Computation and Networking, are not limited to what we are considering Web-based or Web-enabled applications.

As Web technologies mature and evolve, several characterizations from the community are helpful to our discovery of profiling approaches. These characterizations, presented in the following sub-sections, can also help to provide a more explicit characterization of an application, combining several of the above dimensions into a single layered ‘stack’ description.

### 3.2.1 Web Services Stack

From the World Wide Web Consortium (W3C) Web Services Architecture Working Group:

A Web service is a software system designed to support interoperable machine-to-machine interaction over a network. It has an interface described in a machine-processable format (specifically WSDL). Other systems interact with the Web service in a manner prescribed by its description using SOAP messages, typically conveyed using HTTP with an XML serialization in conjunction with other Web-related standards.” [12]

Web services are being supported and adopted by many businesses as a way to securely integrate heterogeneous applications over the Internet [1]. As such, it is a primary strategy in XMSF (see [2], [13], and [14]).

[Web services and service-oriented architectures] ...are going to fundamentally change the way we build our internal systems – the information systems that support our organizations – and how our internal systems interact with external systems... We are on the cusp of building “plug-compatible” software components that will reduce the costs of our software systems at the same time increasing the capabilities of the systems. A service-oriented architecture is essentially a collection of services. Connections among services are Web services. A service is a function that is well-defined, self-contained and does not depend on the context or state of other services. [15]

Figure 3.1 shows the Web Services Stack consisting of:

- Communications protocols,
- Messages, using SOAP [16],
- Descriptions, using WSDL [17],
- Processes for: discovery, such as UDDI [18]; aggregation through Business Process Execution Language for Web Services (BPEL4WS) [19] and
other emerging techniques; and choreography [20].

Other similar descriptions of the Web Services Stack are found in [2] derived from [21].

Web Services Stack

![Web Services Stack Diagram]

Figure 3.1. The Web Services Stack is built upon service description, messaging, discovery, and process standards with over-arching Security and Management considerations. From [12].

Also see [10] for a comparable description of the Web Services Stack showing a perspective on the Security and Management layers. That paper and its referents provide considerable coverage of the aspects that can be considered part of the Management layer (or even, an eventual Management Profile). For example, the Model Driven Architecture “is about using modeling languages as programming languages rather than merely as design languages” [22]:

One of the most important improvements that MDA brings to this picture is an architecture for managing metadata in an integrated fashion, even when the metadata is expressed in widely varying languages … the Meta Object Facility (MOF) uses model-based technology to support the managing of disparate metadata in a coordinated way. The idea is to define a formal model of each language.

Application of such techniques clearly apply to a characterization of XMSF applications, while applying much more broadly as well. For purposes of this paper, we will not look into the Management aspect further. We describe a breakout of the Security layer in paragraph 3.2.4.

3.2.2 Semantic Web Stack

The Semantic Web is “an extension of the current Web in which information is given well-defined meaning, better enabling computers and people to work in cooperation” [23]. The Semantic Web is a systematic approach for creating “smart data” through the following stages [24]:

- Text and databases (pre-XML)
- XML documents for a single domain, where data achieves application independence within a specific domain
- Taxonomies and documents with mixed vocabularies, enabling data to be composed from multiple domains and accurately classified in a hierarchical taxonomy
- Ontologies and rules, where data can be inferred from existing data by following logical rules

Figure 3.2 illustrates the Semantic Web Stack, extending the ideas of “smart data” even further to achieve levels of trust across applications. This model again identifies cross-cutting security layers that act as “gate-keepers” across any level of the stack.

Semantic Web Stack

![Semantic Web Stack Diagram]

Figure 3.2. The Semantic Web Stack adds knowledge description and reasoning specifications onto the basic Web data description layers. From [25]; see also [24], p105.

3.2.3 Semantic Web Services Stack

As might be expected, the parallel conceptualizations of Web Services and the Semantic Web have led researchers to create various descriptions of “Semantic Web Services” (see [26] and [27]). Semantic Web Services are characterized by elements shown in Figure 3.3. As suggested earlier in the discussion of the
LCIM, Web services without semantic representations only support achievement of Level 2 (Syntactical) interoperability. Applications may be able to discover existing services, but automated interpretation of the service descriptions and establishment of logical interactions across applications require interoperability at higher levels (semantic and above). All levels of the LCIM are readily associated with layers of the Semantic Web Services stack as it builds from the basic transport mechanisms (supporting technical interoperability) up through the semantic, pragmatic/dynamic, and conceptual layers.

The Semantic Web Services Stack is perhaps the most comprehensive (currently) structure for Web-based applications in the literature. Recent work envisions integration of the Web-scale distribution capabilities of the Semantic Web with high-speed data and computation scaling via Grid Computing, the “flexible, secure, coordinated resource sharing among dynamic collections of individuals, institutions, and resources” [28]. The emerging concept is variously termed the Semantic Grid and Semantic Grid Services (see [14], [29], and [30]). We will not delve into this further in this study, but recognize that the work will provide identifiable components for the XMSF Profile Study Group to consider for the previously identified Computation dimension of the XMSF Profile space.

3.2.4 Web Services Security Stack

Finally, the Web Services Security Stack identifies various layers of security that are being developed and promoted to apply security concerns across individual service invocations and end-to-end sequences of Web service interactions:

- Advances in Web services security and emerging standards (such as WS-Security and WS-Trust) (see [1]) will provide a trusted and secure environment for small and mid-size businesses to access and link to information cities and reassure wary consumers their transactions are secure over the Internet.

Web services security considerations include [24]:

- Authentication – a means of validating identity.
- Authorization – verifying user permissions.
- Single Sign-On – enabling authentication and authorization across sites and services based on a single interchange.
- Confidentiality – protecting information exchanged.
- Integrity – ensuring information is not altered.
- Nonrepudiation – legally proving a particular user has performed a transaction.

### Semantic Web Services Stack

<table>
<thead>
<tr>
<th>OWL, OWL-S, OWL-Rules</th>
<th>Service Entities, Relations, Rules</th>
</tr>
</thead>
<tbody>
<tr>
<td>RDF/S</td>
<td>Service Instances</td>
</tr>
<tr>
<td>BPEL4WS (Business Process Execution Language for Web Services)</td>
<td>Service Flow &amp; Composition</td>
</tr>
<tr>
<td>Trading Partner Agreement</td>
<td>Service Agreement</td>
</tr>
<tr>
<td>UDDI/WS Inspection</td>
<td>Service Discovery (focused &amp; unfocused)</td>
</tr>
<tr>
<td>UDDI</td>
<td>Service Publication</td>
</tr>
<tr>
<td>WSDL</td>
<td>Service Description</td>
</tr>
<tr>
<td>WS Security</td>
<td>Secure Messaging</td>
</tr>
<tr>
<td>SOAP</td>
<td>XML Messaging</td>
</tr>
<tr>
<td>HTTP, FTP, SMTP, MQ, RMI over IIOP</td>
<td>Transport</td>
</tr>
</tbody>
</table>

Figure 3.3. The Semantic Web Services Stack combines knowledge representation with service representation for intelligent selection and interaction with Web services. From [26] and [31]

Clearly, these are not limited to Web services but apply to a broader set of applications. However, for Web services, particular techniques and standards are emerging as summarized in a representation of the Web Services Security Stack in Figure 3.4. XML-Encryption and XML-Digital Signature can be applied to any XML message transmissions, and therefore form a base on which the higher layers of Web Services Security rest. The intent of the WS-Security layer is to secure routed, multi-hop SOAP messages by encryption [32]. Full specifications for the security layers are being advanced by the Organization for the Advancement of Structured Information Standards (OASIS).
Web Services Security Stack

<table>
<thead>
<tr>
<th>WS-SecureConversation</th>
<th>WS-Federation</th>
<th>WS-Authorization</th>
</tr>
</thead>
<tbody>
<tr>
<td>WS-Policy</td>
<td>WS-Trust</td>
<td>WS-Privacy</td>
</tr>
</tbody>
</table>

- WS-Security
- SOAP
- WSDL
- XML Digital Signature
- XML Encryption

Figure 3.4. The Web Services Security Stack builds on the foundation of XML-Digital Signature and XML-Encryption to create levels of trusted, secure end-to-end service interactions. Adapted from [32], [33], and [34].

As discussed previously, an XMSF Profile needs to identify Security implementation details since it will affect the ability of the application to interact with other applications across all interoperability levels. An application designed for a high level of interoperability (e.g., pragmatic/dynamic or conceptual) may be able to interoperate with one application at that level through a particular security setting but be restricted to a lower level of interoperability with another application through a different security setting. These techniques may enable new approaches addressing multi-level security concerns.

3.3 An Elementary Profile Approach

Based on the foregoing discussion, we can move forward with a profiling approach that identifies (1) an Interoperability Profile, taken as the level of interoperability according to the LCIM; (2) an Implementation Profile from identification of Web technologies from the Semantic Web Services Stack in Figure 3.3; and (3) a Security Profile from identification of security implementation standards from the Web Services Security Stack in Figure 3.4. As stated earlier, this enables us to address, at least in an initial way, the first two parts of the XMSF Profile definition; namely (1) applicable Web technologies and protocol standards and (2) applicable data and metadata standards. It is important to note that the naming of these “profiles” is done as a convenience for reference in this paper – this approach is purely exploratory and has not been sanctioned by the XMSF Profile Study Group per se (although the author is a member of that group).

In the following sections, we look at two current projects that involve significant application of XMSF concepts. We use the profile approach outlined above to characterize the work in progress on these projects, both in terms of an initial assessment of the applications before adding capabilities aligning with XMSF concepts and the vision of the nature of the applications after the current (in progress) XMSF work is completed. The “before” and “after” profile description of the projects will help characterize the nature of the work being performed. Perhaps more importantly, the attempt to use the profiling approach on actual systems provides insights that can inform the ongoing XMSF Profile Study Group efforts. It is interesting to see that in each case there are XMSF concepts that were designed into the various software applications from the start, as well as XMSF concepts that are being added into the products as capability upgrades to enable greater interoperability with other systems.

4. OPNAV N81 World-Class Modeling Exemplar Project

Tasking to the NPS MOVES Institute from OPNAV N81 under the World-Class Modeling (WCM) project is designed to evaluate the ability of XMSF concepts to provide an advanced framework for M&S collaboration and to build a technical foundation for a “marketplace of capabilities.” To prove the concept can work and produce meaningful data from known methods, the goal is to fuse the best methods from three government-owned simulation programs. Simkit [35] is an open source discrete event simulation (DES) engine written in Java and developed at the NPS. The Naval Simulation System (NSS) (see [36], [37], and [38]) is a Naval combat object-oriented simulation model that emphasizes the Naval Commander’s perception of the battlefield, communications and commanders response. N81 has sponsored considerable work in expanding NSS capability to capture C4ISR issues, particularly to address analyses relating to the FORCEnet concept [39]. CombatXXI [40] is a ground combat object-oriented simulation model that emphasizes Army and USMC C4ISR issues. CombatXXI is currently under development, written in Java and uses the Simkit DES API. By fusing these systems together the resulting software is expected to provide a significant joint modeling capability incorporating the best methods from Naval and Army analytical models that can
provide insight into Network Centric Warfare for all services.

Table 4.1 characterizes the existing components (prior to the start of the current OPNAV N81 program) in terms of the XMSF profiling approach described above. Discussion of this characterization follows.

Table 4.1 WCM “Before”: XMSF Profile
Characterization of WCM components at the start of current software development efforts.

<table>
<thead>
<tr>
<th>XMSF Profile</th>
<th>Simkit</th>
<th>NSS</th>
<th>CombatXXI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interoperability Profile</td>
<td>L0 No connection</td>
<td>L2 Syntactical (HLA)</td>
<td>L2 Syntactical (HLA)</td>
</tr>
<tr>
<td>Implementation Profile - XML</td>
<td>None</td>
<td>None</td>
<td>Data representation</td>
</tr>
<tr>
<td>Security Profile</td>
<td>None</td>
<td>Username/password authentication to server</td>
<td>None</td>
</tr>
</tbody>
</table>

The LCIM level (L2, Syntactical) for NSS and CombatXXI is based on prior and planned (respectively) implementation of the High Level Architecture (HLA) [41] in both systems. NSS has previously participated in HLA federations. CombatXXI was initially designed to support HLA; however, it has not yet been certified for HLA operation in its current state of development. CombatXXI may also be fielded with capability to interact with other systems through the Distributed Interactive Simulation (DIS) [42] interface. Simkit inherently has no connectivity to other systems, although simulations developed using Simkit (e.g., CombatXXI and numerous NPS thesis research efforts) have demonstrated LCIM level 1 and 2 interoperability.

Whereas use of XML has been a major focus in the representation of data in CombatXXI throughout its design and development, there have been limited investigations into the use of XML in NSS (e.g., see [43]) prior to current N81 XMSF efforts.

As shown, the components do not reflect Web-based security characteristics identified in the Web Services Security Stack (user authorization in NSS does not use these emerging security mechanisms).

Table 4.2 characterizes the XMSF Profile of the components given successful completion of efforts in progress to enable the systems to interact through the Simkit API implemented as a Web service.

Table 4.2 WCM “After”: XMSF Profile
Characterization of WCM components after current and near-term software development efforts.

<table>
<thead>
<tr>
<th>XMSF Profile</th>
<th>Simkit</th>
<th>NSS</th>
<th>CombatXXI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interoperability Profile</td>
<td>L4 Pragmatic/Dynamic</td>
<td>L3 Semantic</td>
<td>L3 Semantic</td>
</tr>
<tr>
<td>Implementation Profile - XML</td>
<td>Modeling</td>
<td>Data representation</td>
<td>Messaging</td>
</tr>
<tr>
<td>- SOAP</td>
<td>Messaging</td>
<td></td>
<td></td>
</tr>
<tr>
<td>- WSDL</td>
<td>Service description</td>
<td></td>
<td></td>
</tr>
<tr>
<td>- UDDI</td>
<td>Service registration and discovery</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Security Profile</td>
<td>None</td>
<td>Username/password authentication to server</td>
<td>None</td>
</tr>
</tbody>
</table>

The dramatic increase in LCIM level for Simkit results from exposure of API calls through Web services and development of an XML Schema representation of event graph notation (see [44] and [45] for information on event graph notation). A visual Simkit modeling tool is in development that enables a user to create new Simkit models in event graph notation. The tool also provides the user access to a library of stored models and model components that can be used to simplify model generation, extension, and adaptation. The visual representation is converted to storage in a self-validating XML format from which Java software is auto-generated to create executable Simkit models. The formalized representation creates a basis for controlled, repeatable, and reusable software components, as well as a semantic basis for interoperability with other systems. With the semantic formalization, exposure of the API as Web services enables other systems to readily interoperate with the Simkit framework.

Creation of a common data interchange language provides rationale for elevation of the NSS and CombatXXI Interoperability Profiles to LCIM Level 2 (Semantic). Current efforts are creating an XML representation of NSS data and common semantics for
runtime data interchange through the Simkit Web services. Planned efforts will examine existing standard data model formalizations, such as C2IEDM [46], to solidify the interoperability between these systems and across other systems employing that data model (also discussed in the FAST exemplar in the next section of this paper).

The XML representation of the event graph is sufficient to autogenerate Java code using the Simkit library. Transformation from the event graph representation to the Simulation Reference Markup Language (SRML) [47] is an interesting area for follow-on research. In principle, transformation from one XML representation to the other using XSLT is feasible.

Unfortunately, current and immediately planned efforts do not address the Security Profile aspects of these components. This remains a significant area for further study and development in the XMSF community.

5. Flexible Asymmetric Simulation Technologies (FAST) Exemplar Project

As stated earlier, the objective of the FAST OOTW Toolbox project is to provide an integrated set of combat simulations, databases, and computational tools to military analysts deploying to theater or supporting operations from a reach-back center, with primary focus on tools supporting OOTW mission planning and assessment.

Components currently included in the toolbox are:
- Toolbox Controller
- Unit Order of Battle Data Access Tool (UOB DAT) [48]
- Diplomatic and Military Operations in a Non-warfighting Domain (DIAMOND) [49]
- Joint Conflict and Tactical Simulation (JCATS) [50]
- Interim Static Stability Model (ISSM) [51]
- Canadian Forces Landmine Database (CFLD) [52]
- XML Management Tool (XMT)

Table 5.1 provides a characterization of the toolbox components in terms of the XMSF Profile approach described earlier. Development of the toolbox has included creation of a common data language in XML – the FAST Data Interchange Format (DIF) – being used for data interchange across the models and tools.

The data interchange is implemented as a static data transfer prior to execution of the models. The XMT tool enables a Toolbox user to identify what files to transform from one format to another by invoking an XSLT file on the source data file. For example, UOB exports force structure data in XML format that can be transformed into a DIAMOND or JCATS XML representation to help initialize a scenario. Recent project work demonstrated the benefit of having an open, well-defined standard data representation through use of the FAST DIF to initialize a custom agent-based simulation. Analysts can use the shared data representation for any number of different purposes with different software tools.

Table 5.1 XMSF Profile Characterization of FAST OOTW Toolbox components.

<table>
<thead>
<tr>
<th>XMSF Profile</th>
<th>Interoperability Profile</th>
<th>Implementation Profile</th>
<th>Security Profile</th>
</tr>
</thead>
<tbody>
<tr>
<td>Toolbox Controller</td>
<td>L0 No connection</td>
<td>None</td>
<td>None</td>
</tr>
<tr>
<td>UOB</td>
<td>L1 Technical</td>
<td>Client/server software product</td>
<td>Username/password authentication to server</td>
</tr>
<tr>
<td>DIAMOND</td>
<td>L2 Syntactical</td>
<td>XML data representation</td>
<td>None</td>
</tr>
<tr>
<td>JCATS</td>
<td>L2 Syntactical</td>
<td>XML data representation</td>
<td>None</td>
</tr>
<tr>
<td>ISSM</td>
<td>L0 No connection</td>
<td>None</td>
<td>None</td>
</tr>
<tr>
<td>CFLD</td>
<td>L0 No connection</td>
<td>HTTP/HTML</td>
<td>None</td>
</tr>
<tr>
<td>XMT</td>
<td>L2 Syntactical</td>
<td>XML data transformations</td>
<td>None</td>
</tr>
</tbody>
</table>

Currently, the UOB tool is a client/server product. It offers a clear opportunity for implementation as a Web service to expose the UOB server functionality, thereby allowing software (and software agents) to access the force structure database services without the necessity of downloading and executing the separate client application.

The ISSM product is a spreadsheet application that could potentially use data computed from execution of the other models as a way of updating information in the spreadsheet. Work is ongoing in the FAST program to define such a data exchange. The CFLD tool is a hypertext application providing access to information about landmines around the world. As such, it is at the lowest level of the Implementation
Profile, but offers opportunity for enhancement as a Web service in the future.

Primary ongoing efforts involve strengthening the common XML representation of data for interchange across the models and between C4I systems and the models. Research is in progress to assess standard data models, such as the C2IEDM, for applicability as a common data exchange format across the models and between the Toolbox and C4I systems.

As in the WCM project work described previously, implementation of Security Profile levels remains a significant challenge for future work.

### 6. Challenges and Opportunities

This paper has described current efforts to incorporate and demonstrate XMSF principles across several analytic combat models. As described, we have only begun to scratch the surface in defining practical techniques for specifying XMSF profiles for existing and future applications. The work described in this paper is one of the earliest approaches to be laid out in some detail. The SISO XMSF Profiles Study Group will evaluate the ideas expressed here as well as many others emerging from the M&S community before drafting its findings to SISO for community consideration. Other developers are encouraged to join this process through Study Group participation and through examination of particular exemplars as done here. Only through broad community involvement can we bring together policies, practices, standards, and procedures that will benefit DoD M&S development as we continue this exciting Information Technology (r)evolution exploiting Web-based technologies.

### 7. References


http://www.w3.org/TR/wsdl (viewed March 2004)

[18] Organization for the Advancement of Structured Information Standards (OASIS): “Universal Description, Discovery and Integration (UDDI) Specifications.”


[34] Netegrity: Introduction to XML, Web Services Standards and Security


[38] L. C. Yu, J. S. Steinman, and G. E. Blank: “Adapting Your Simulation for HLA,” SIMULATION:


[42] Institute for Electrical and Electronic Engineers (IEEE): IEEE 1278 Standard for Distributed Interactive Simulation. (see above site for access)


Author Biography

CURTIS BLAIS is a Research Associate at the Naval Postgraduate School (NPS) in the Modeling, Virtual Environments, and Simulation (MOVES) Institute. His principal research areas include Web-based technologies for military Modeling and Simulation, Web-based 3D graphics for military education and training, and multi-agent simulations. Prior to joining NPS in 1999, Mr. Blais spent 25 years in software development in industry and government working on Navy and Marine Corps command staff training systems and C4I simulations. Mr. Blais has a B.S. and M.S. in Mathematics from the University of Notre Dame and has advanced to candidacy in the MOVES Ph.D. program at NPS.