JSB Composability and Web Services Interoperability via Extensible Modeling & Simulation Framework (XMSF), Model Driven Architecture (MDA), Component Repositories, and Web-based Visualization

Dr. Don Brutzman
Naval Postgraduate School
Modeling, Virtual Environments & Simulation (MOVES) Institute
brutzman@nps.navy.mil
work +1.831.656.2149

Dr. Andreas Tolk
Old Dominion University
Virginia Modeling Analysis & Simulation Center
atolk@odu.edu
work +1.757.686.6203

7 November 2003

Assertion: this study paper is provided to the U.S. government, public and industry with unlimited rights.

1 Overview

This paper summarizes research work conducted by organizations concerned with interoperable distributed information technology (IT) applications, in particular the Naval Postgraduate School (NPS) and Old Dominion University (ODU). Although the application focus is distributed modeling & simulation (M&S) the results and findings are in general easily applicable to other distributed concepts as well, in particular the support of operations by M&S applications, such as distributed mission operations. The core idea of this work is to show the necessity of applying open standards for component description, implementation, and integration accompanied by aligned management processes and procedures to enable continuous interoperability for legacy and new M&S components of the live, virtual, and constructive domain within the USAF Joint Synthetic Battlespace (JSB).

JSB will be a common integration framework capable of supporting the future emerging simulation needs ranging from training and battlefield rehearsal to research, system development and acquisition in alignment with other operational requirements, such as integration of command and control, support of operations, integration of training ranges comprising real systems, etc. To this end, the study describes multiple complementary Integrated Architecture Framework approaches and shows, how the various parts must be orchestrated in order to support the vision of JSB effectively and efficiently. Topics of direct relevance include Web Services via Extensible Modeling & Simulation Framework (XMSF), the Object Management Group (OMG)'s Model Driven Architecture (MDA), XML-based resource repositories, and Web-based X3D visualization. To this end, the report shows how JSB can

- Utilize Web Services throughout all components via XMSF methodologies,
- Compose diverse system visualizations using Web-based X3D graphics,
- Benefit from distributed modeling methods using MDA, and
- Best employ resource repositories for broad and consistent composability.

Furthermore, the report recommends the establishment of necessary management organizations responsible for the necessary alignment of management processes and procedures within the JSB as well as with neighbored domains. Continuous interoperability cannot be accomplished by technical standards alone. The application of technical standards targets the implementation level of the system of systems, which results in an interoperable solution valid only for the actual implementation. To insure continuity, the influence of updates, upgrades and introduction of components on the system of systems must be captured in the project management procedures of the participating systems.

Finally, the report proposes an exemplifying set of proof-of-capability demonstration prototypes and a five-year technical/institutional transformation plan. All key references are online available at http://www.movesinstitute.org/xmsf/xmsf.html (if not explicitly stated otherwise).

2 Utilizing Web Services throughout all JSB components via XMSF

2.1 Overview

XMSF is defined as a composable set of standards, profiles and recommended practices for web-based modeling & simulation (M&S). Web-based technologies applied within an extensible execution framework are enabling a new generation of M&S applications to emerge, develop, and interoperate. An Extensible Markup Language (XML) framework can provide a necessary and practical bridge between forthcoming M&S requirements and open/commercial Web standards. A Web approach for technology, software tools, content production and broad use makes great sense technically, and also provides best business cases from an enterprise-wide perspective.

Several decades of progress in modeling & simulation (M&S) continue to enjoy ongoing exponential improvements in computer price and performance. Nevertheless, the real world is a big place. Models, data and simulations of world-wide scope eventually overwhelm any single computer (or supercomputer). Large-scale virtual environments (LSVEs) must therefore consider networked architectures that allow distribution of components, both for construction and for interaction. Composability and compatibility of models is achieved through consistent design patterns utilizing the interoperability of open standards. Scalability is achieved by using both technologies and enterprise-wide business practices compatible with the only system having truly global scope: the World Wide Web. Such an approach yields a feasible path to resolve the many hundreds of stovepiped (i.e. custom-connected) C^4I systems employed by U.S. warfighters, while further connecting them to an entire field of otherwise-disjoint modeling & simulation (M&S) applications.

This technical strategy has been examined in detail by the Extensible Modeling & Simulation Framework (XMSF) effort. XMSF has been subsequently tasked by the Defense Modeling & Simulation Office (DMSO) to investigate use of Web Services (and hence related LSVE approaches) to bridge M&S and C^4I systems scalably. The work on LSVE technologies for large-scale military systems provided in this report is consistent with XMSF principles and emerging XMSF practices. Recent declarations by the Defense Information Systems Agency (DISA) that all Global Information Grid (GIG) capabilities will be connected via Web Services provides further validation of the importance of this approach. At the October 2003 WebSim Symposium, DMSO and DISA both declared publicly that Web Services and C4I/M&S interoperability are essential. This validates the fundamental importance of the XMSF work.

2.2 Integration and Implementation are Essential

The multiple challenges involved in modeling all aspects of joint operations for military simulations cannot be completely solved in isolation. 3D graphics, underlying model representations, networked distribution and virtual/real synthesis at first appear to be different topics. Nevertheless it is a fact that solutions in each area must simultaneously consider the constraints and capabilities of the other areas. Compatible integration of these technologies within a Web-based framework can allow effective construction of synthetic battlespaces incorporating diverse military entities and simulations. The approaches presented in this report enable such results by emphasizing standardized interoperability, scalable architectures, demonstrated examples and directions for future work.

The Department of Defense (DoD) is engaged in warfighting and institutional transformation for the new millennium. In parallel, DoD Modeling & Simulation (M&S) needs to identify and adopt transformational technologies providing direct tactical relevance to warfighters. The only software systems that composably scale to worldwide scope utilize World Wide Web technologies. Therefore, it is 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.

2.3 Current Shortcomings and Motivations

A number of severe gating problems are evident in the current generation of defense-related modeling and simulation systems. Hundreds of active legacy applications have limited commonality, mixed levels of support and stove-piped interoperability. Despite the best efforts of numerous programs, the difficulties inherent in current M&S strategies have thwarted the deployment of tactically useful systems into the hands of warfighters. Interoperable software, networking and message semantics are needed at all levels of activity.

This need for scalable interoperability is growing faster than ever before, as nearly all operations become coordinated joint/coalition efforts, and diverse new agencies for homeland defense and peacekeeping operations become critical partners. Current common shortcomings include:

- Few current applications successfully leverage commercial software imperatives. Interoperable reuse is essential for feasibility, life-cycle supportability, fundability and product flexibility.
- Modeling and simulation is not a significant day-to-day asset for U.S. operating forces.
- A spectrum of operational goals needs to be met: joint warfighting, homeland defense and coalition peacekeeping operations. Tactical needs are broad, immediate and interrelated, thus approaches must be scalable and take a global scope.
- Technical limitations are evident in current software. New capabilities are needed that work correctly in small scale but can also grow/aggregate into much larger scales.

- Current DoD software strategies do not leverage commercial-sector investments in interoperable web technology; so planned improvements perpetuate this disconnected state of affairs.
- Distance-learning technologies - e.g. audio/video/whiteboard/documents/Advanced Distributed Learning (ADL)/Sharable Content Object Reference Model (SCORM)/etc. - are not compatibly augmenting or utilizing available simulation technology.

Clearly many strong motivations exist for significant progress and transformational change.

2.4 Intellectual Basis

For the last decade, the XMSF partners have been directly engaged in the conduct of research and development across a range of technologies for networked modeling & simulation. Many capabilities and challenges have been revealed over time – technical, political and social. During 2001-2002 we jointly decided that Web-based M&S was the fundamental barrier and opportunity that needed to be surmounted in order to meet the broad needs of defense. To achieve this, we needed to go beyond our individual efforts and establish the intellectual basis for broad-based adoption of Web-capable technologies. Towards this end, we planned and executed a series of watershed events in 2002: an undeniably authoritative workshop to determine consensus on key challenges, and a symposium to publicly present workshop conclusions for bootstrapping theory into practice.

Results exceeded our most optimistic projections. Three dozen practicing researchers from all points of the M&S compass provided point papers on the central challenge: "what capabilities and shortfalls does Web-based M&S present to your discipline?" Three areas were examined, corresponding to the major communities of practitioners involved in such activities: Web/XML, Internet/Networking and M&S. After comparing all "can do" and "cannot do" points together, group consensus was surprisingly strong: essentially all perceived limitations were addressable by other provided capabilities. Web-based M&S was declared feasible across the full range of technical requirements. Subsequent symposium reactions from a dozen practicing program managers further endorsed these results: each saw the capabilities and conclusions of the workshop as directly relevant and usable in their program areas.

These are strong results. We have consistently placed all XMSF contributions online so that the specific opinions and results of participants can be examined in the light of these consensus conclusions. Knowing that the overall problem is challenging but feasible, with no technical showstoppers, is tremendously enabling when pursuing global architectures and systems.

XMSF integrates several high-level requirements derived from years of experience with M&S frameworks, and the challenges of their effective deployment across diverse networks and systems. XMSF must 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. Clearly XMSF must support composable, reusable model components. XMSF use must not be constrained by proprietary technology or legally encumbering patents, since such barriers discourage the free, open, *ad hoc* development of interconnected tactical models and simulations.

The precepts of XMSF are:

- Web-based technologies applied within an extensible framework will enable a new generation of 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 of 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, complementary technical approaches are usable for model definition, simulation execution, network-based education, network scalability, and 2D/3D graphics views.
- Web approaches for technology, software tools, content production and broad use provides best business cases from an enterprise-wide (i.e. world wide) perspective.

The final version of the 2002 XMSF technical report includes key findings from the XMSF Technical Challenges Workshop conducted at the Naval Postgraduate School, Monterey California on 19-20 August 2002, plus considerations and recommendations from the XMSF Strategic Opportunities Symposium held at George Mason University, Fairfax Virginia on 6 September 2002. Critical consensus points emerging from both the Workshop and the Symposium include the following findings.

- The XMSF concept must continue to be refined from a high-level concept to definitive technical recommendations, practices, and applications tailored for the M&S domain.
- A set of exemplar applications need to be identified and initiated that can collectively and clearly demonstrate the application potential of XMSF concepts. A number of existing and emerging programs are examined as possible exemplars.
- Web Services appear to be a promising application of technology for focusing future work.
- Security concerns are cross-cutting for all areas and must be addressed throughout any design process (i.e. built in from the outset).

Our current efforts are addressing all of these priorities. Most prominent are standards-organization activities, and public exemplars at the annual I/ITSEC conference. Further insights include the following “what it is, what it is not” summaries.

Frequently asked question #1*: what does XMSF look like?*

- Web, internet and XML technologies for open interoperability in M&S
- Data and metadata standards for semantic consistency among systems
- Profile specifications, associated with standards, to define common capability levels needed for user requirements and application support
    - Specification of mandatory (and optional) standards and recommended practices
    - Recommendations and guidelines for implementation (e.g. composability requirements, recommended technologies, application guidelines, recommended hardware configuration)
    - Implementation and evaluation metrics to measure conformance and capabilities

Frequently asked question #2*: what doesn’t XMSF look like?*

- A single, exclusive, tightly coupled architecture
- Proprietary technologies which require licenses or royalties for use

These points are chosen (out of many possible points) because they drive data languages, programming projects and joint architectures into proven patterns of success that have led to broad Web-based success. Just as such approaches are proving feasible for XMSF efforts, and DMSO/DISA strategies, and even larger industry practices, they are also feasible as organizing principles and the central technical basis for an all-encompassing JSB architecture.

The foundation for XMSF's future success is based on multiple, compatible, self-reinforcing technical and programmatic strategies. Viewed from an enterprise perspective, commitments to open standards processes are the most cost effective approach over the long-term lifecycle of technology development and deployment. It is also important to have common business models for delivering expert services and developing compatible domain-specific applications. Partnerships with commercial industry can leverage technology opportunities to improve interoperability and achieve greater defense capabilities. Many incentives exist to begin demonstrating XMSF capabilities immediately as a prelude to transformational change for the tactical, training and acquisition arenas.

2.5 Key Challenges for XMSF

Many issues and goals have been identified. Top-level XMSF challenges include:

- Utilize web-based technologies for more powerful and cost-effective government-wide networking, serving, client-side rendering and user interaction.
- Provide open, affordable, extensible modeling and simulation capabilities for tactical scenarios of direct use to participants engaged in conflict and peace operations.
- Employ mainstream practices of enterprise-wide software development.
- Improve ease of use for developers and users, fueling rapid growth of interoperable simulations.
- Provide support for all types and domains of M&S: constructive, analytical, live, virtual, playback-driven, agent-based, human-in-the-loop, heterogeneously distributed, logistical, and others.
- Models of interest reflect reality. Both simulations and tactical exercises are the behavior of models over time. Models and simulations need to match tactical requirements for <u>rehearsal</u>, <u>reality</u> and <u>replay</u> to meet operational needs.

Each key challenge will help guide emerging technical and programmatic strategies for XMSF.

2.6 Web and XML Considerations

The ambitious nature of the many requirements and challenges of defense M&S requires aggressive reliance on standardized, openly available, legally unencumbered, commercially available technologies. Sufficient support for DoD M&S needs will require active engagement with standards development groups such as Institute of Electrical and Electronics Engineers (IEEE), Internet Engineering Task Force (IETF), International Standards Organization (ISO), Object Management Group (OMG), Open GIS Consortium (OGC), Organization for Advancement of Structured Information Standards (OASIS), the Simulation Interoperability Standards Organization (SISO), the World Wide Web Consortium (W3C), and the Web3D Consortium.

The diversity of defense, government, public, scientific and international needs for M&S means that cross-platform capabilities are essential. No single operating system or monolithic hardware architecture can possibly be forced upon so many existing and legacy systems. Cross-platform data interoperability is critically important when considering the plethora of customized tactical systems connecting to worldwide tactical networks.

A particular strength of an XMSF approach based on web technologies is that the most difficult interoperability challenges are already resolved (or else are being solved) by the development of tightly interdependent and highly complementary Web standards. The W3C and the IETF are the leading drivers in these efforts. Thus it appears that this web-technology strategy for XMSF can provide the most technically robust solutions, with the most reliable future-growth processes and best-case business practices. This is particularly important when viewed from an enterprise-wide (i.e. USAF-wide, DoD-wide and coalition-wide) perspective.

To meet these larger requirements, XMSF systems will employ object-oriented paradigms and validatable structured data in a language-independent and object-system-independent manner. Design patterns will unambiguously define programming-language bindings by mapping representations and component models from root XML schemas to multiple programming languages and application programming interface (API) bindings. The Interface Description Language (IDL) provides further good capabilities in this area. Software component functionality and interactions will be further documented using the Unified Modeling Language (UML) plus well-defined design patterns, harmonized at a high level of interoperability by the Object Management Group (OMG) Model Driven Architecture (MDA).

XMSF will have a modular framework, with kernel plug-ins to support extensions and modifications to framework layers as low as the network layer. Design patterns for modular extensibility are needed at all levels and across system lifecycles, in order to support future growth and backwards compatibility as well as multiple-system interoperability.

To support real-world military secure communications systems, XMSF must be compatible with currently fielded wireless and wired military technologies to include data/voice Single Channel Ground and Airborne Radio System (SINCGARS), Ultra High Frequency (UHF)/Very High Frequency (VHF) radios and Digital Subscriber Network (DSN). Diverse network channels and transport mechanisms will thus drive some application-level design decisions when applying various web technologies.

Functional Requirements

Many of the functional requirements described below overlap, complement or build on one another. The crux of these requirements is that they are considered the key properties that a framework must have in order for it to be platform-independent, flexible, extensible, secure, distributed and dynamically reconfigurable.

a. Data Representations

Data is defined as any information of interest that is to be exchanged between two systems. XMSF will need to be able to represent exchangeable data in a language-independent manner. For troubleshooting and confidence, data must be readable both by humans and by a complete variety of computer languages, e.g. Ada, C++, Java, Perl, Prolog, etc. Such data interchange is typically addressed by using structured text-based standards.

The logical implication of data being machine-readable is that the data representation will need to be structured and self-defining. For future capabilities, most data representations need to allow for facile extension of the represented data.

Given the verbose nature of most text-based representations, data representations will also need to support compression schemes, applicable both to documents and streams equally. Default (i.e. run-time replaceable) compression algorithms must be offered, probably as code components and addable stream filters. Of particular note is that compression is closely interrelated to encryption, authentication, composition, key management, and completeness of delivery.

The current state of standards evolution already accounts well for most of these requirements. XML is the preferred structured-data standard for platform-independent representation that, when carefully applied, can meet most (and often all) of these requirements.

b. Security Considerations

XMSF security will encompass identification, authentication, authorization and encryption. Functional access restrictions (e.g. role-based permissions) are considered to be the responsibility of the application, or the application environment.

It is desirable for a framework such as the XMSF to offer utilities (probably through a code component and/or stream filter) that include one or more default encryption algorithms. This can allow applications to interact in a commonly acceptable way if they do not need a specific encryption implementation.

The framework must also select a standard for signing messages and documents. The existing XML Digital Signature (DS) specification (RFCs 3076 and 3275) is a likely candidate. The signature itself does not provide authentication, but rather associates an identity with data. Developments of related interest include industry efforts such as the Liberty Alliance project (http://www.projectliberty.org) and Passport (http://www.passport.com).

Following on from identification, the framework must define standards for authentication. As is the case for encryption, it appears preferable that a pre-existing mechanism (outside applications) be made available to provide authentication services. This might be implemented via authentication servers. A requirement that follows from the nature of dynamic reconfiguration is that there needs to be a mechanism for defining groups and group membership. Additionally, the membership of those groups needs to be dynamic. A further consideration is that the groups must be definable in such a way as to apply to either a single service, or span multiple services (as in the case of large-scale distributed multi-application simulations interconnected with large-scale live systems).

The increasing focus on security means that XMSF must be underpinned by the strongest and most current web security technologies. These are additional capabilities that can augment military-grade security for classified, unclassified and administrative networked systems. Security is cross-cutting issue that must work sufficiently and simultaneously across all three areas (Web/XML, Internet/networking and M&S) or new vulnerabilities will result.

Classified information security systems remain responsible for meeting military security requirements. Web-based content does not replace or jeopardize any of those existing, externally controlled techniques.

c. Service Descriptions and Bindings

Web services typically include a logically coherent set of functions offered for discovery and remote invocation across the internet by a code component. A code component may use many such services.

The functionality offered by a code component will need to be represented in a language-independent manner. This means that for the various programming languages of interest (e.g. C++, Java, Fortran, etc.) used to develop the code component, and for the platforms on which the code is deployed, common-denominator representations of the exposed functions and the parameters of those functions (i.e. the interface) will need to be represented consistently. Thus the service description needs to be binding independent. The corollary of that implication is that the service description needs to employ a common binding specification.

If the underlying mechanism employed for defining API binding mechanisms is the same as for data representation (e.g. XML), then many of the issues relating to platform independence are resolved. Loose binding and catalog lookup of available services further allows graceful upgrading of offered capabilities without breaking existing systems, keeping global information dependencies strong and robust throughout multiple simultaneous system lifecycles.

d. Graphical User Interface (GUI) Descriptions

A Graphical User Interface (GUI) is defined as a man-machine interface of a graphical (as opposed to a textual) nature. Typically these are things like windows, toolbars and dialogs, but 3D virtual environments are also encompassed.

In a similar manner to Service Descriptions, a GUI description will need to represent user interface elements in a computer-independent (language and platform) manner. Further, the GUI description will need to define not only the appearance of graphical elements, but also their behavior. In this case behavior is the component’s response to user stimulus.

The aim of a GUI description is to define a consistent look and feel across operating systems. Use of the Model-View-Controller (MVC) design pattern/paradigm is closely related and widely repeatable, since MVC clearly and cleanly separates functionality, presentation and control.

e. State-Transition Description

State transition is defined as the progression of a system through a set of logical states. In effect this can translate to allowable sequences of messages. Some simulation applications may need to share state-transition definitions in order to effectively model certain shared processes.

Since state transitions deal with the logical domain rather than the physical domain, there are fewer issues of representation. If a workflow representation is simply exchangable between systems, then it suffices to use a computer-independent data representation to address platform independence issues. All that then remains is for syntax to be developed for the workflow representation.

One interesting requirement is that even though a set of logical state transitions may be published, these do not necessarily reveal the actual internal logic (or internal state transitions) of the implementing code component. Published state transitions are the basis for functional interoperability among participating entities, and may simply serve as a critical subset of the actual state transitions in each system.

f. Transactions

A transaction is defined as a logical set of changes that must be made as a single activity, e.g. a funds transfer from one account to another must both debit the source account and also credit the destination account, all as a single atomic action. Similar usage patterns are common across the spectrum of tasks performed in information-based operations.

A common pattern for such transactions is a 2-phase commit procedure. Unfortunately, this approach can suffer from latency and heavy resource utilization when implemented across the Internet. An alternative approach to 2-phase committal is that of adding undo operations to individual atomic actions. The idea is that certain (simpler) actions can be reversed by another action, e.g. the request to be added to a mailing list can be undone by a request to be removed from the mailing list. Client, server and communications mechanisms are all affected by such patterns, hence their importance.

A requirement for the M&S framework is that a transaction pattern (that may encompass more than one application paradigm) needs to be defined and supported. Supported approaches need to allow for both simple request-response situations that do not require the overhead of a 2-phase commit, and also more complex situations that do require more sophisticated 2-phase commit procedures.

g. Ontologies

An ontology is defined as a basis of meaning. This is a fundamentally difficult area that has seen much research progress in recent few years as part of the W3C's Semantic Web.

The first requirement in the area of ontologies is to allow definition and approval of complementary taxonomies that can be applied across multiple XMSF application domains. This will allow for the consistent classification of data and services via precise vocabularies. XML Schema and XML Namespaces are the primary mechanisms for defining and referring to such vocabularies.

A subsequent requirement is to establish consensus on common meanings. It does not suffice for there to be agreed meaning within a group, since to be truly useful, there needs to be a mechanism for defining the equivalence of terms between groups. Such agreements are needed for both extensibility and for interoperability. XML Schema annotations and XML

Internationalization (I18N) / XML Localization (L10N) provide the mechanisms for recording and translating accepted meanings in a reviewable fashion.

An open issue is the establishment of XML schema and ontology repositories for common service representations. The following semantic representations are expected to be of particular interest.

- Resource Description Framework (RDF) and Web Ontology Language (OWL)
- DARPA Agent Modeling Language (DAML) and Ontology Integration Language (OIL)
- NATO-developed Generic Hub information-exchange data model for tactical operations

It will be particularly interesting to consider the implications of ontologies like Generic Hub that help to establish commonalities between services and coalition partners. Development of effective ontologies for military operations orders (which contain tactical versions of who, what, when, where and how) is a strategically important application area deserving dedicated further work.

h. Repositories

A repository is defined as a logically related collection of information, accessible through a common point of reference. XMSF applications will need numerous repositories across different levels of abstraction, presumably exposed via Web Services. Work is needed to identify potential libraries of components that can be made available to support reusability, encourage interoperability, and reduce user learning curves. Example application-level repositories are likely to include:

- 2D/3D datasets, imagery, models, metadata (e.g. order of battle catalogs)
- Portable computational models, such as physics of entity and sensor interactions
- Software-agent templates with requested capabilities
- Stream-specific adaptors/components
- Exercise simulation management
- Operational recording of simulated or actual interactions
- Order of battle (inventories and functional characteristics of friendly and opposing forces)

It appears likely that each logical level of a “XMSF stack” (probably corresponding to an augmented Web Services stack) may have one or more associated repositories. For the purposes of this report, the requirement for repositories will be assumed to be an implicit requirement for each of the preceding areas discussed. XMSF profiles will document necessary levels of data, software and resource support.

A shared requirement necessary for the effective use of repositories is that common interfaces are defined to allow consistent access to contained information by search engines and other interested applications. Universal Description, Discovery and Integration (UDDI) fulfills this need for Web Services, and may be sufficient for XMSF. Registry functionality is intrinsic to the usefulness and growth capabilities of repositories.

i. Search Engine Capabilities

A search engine is defined as a code component that extracts information matching a specified set of criteria from one or more repositories. One of the great challenges of the Internet is locating information. In order for XMSF to not fall prey to the same shortcomings it is important to provide sufficient support for capable search engines. Self-describing information incorporated within each object of interest as commonly understandable (i.e. common ontology) metadata will become a critical prerequisite for all software design patterns, structured data and integrated systems.

The areas discussed in preceding sections are a good starting point for search topics in the various repositories. Hence common search criteria will likely include topics such as Provider, Type of Service, Name, Quality of Service (QoS), Security and other constraints. It is likely that typical e-commerce web-service descriptions will need to be augmented to fully describe needed functionality pertaining to distributed M&S applications contributing to JSB capabilities.

j. Composability

Composability is defined for XMSF as the ability to select and combine components in various combinations to create new functionality which satisfies specific user requirements across a variety of application domains. This applies both during design and implementation, and during runtime. Automated, tool-based support is a composability goal.

Run-time composition of new components and existing components is a long-running area of research that finally appears to be ready for widespread practical application. Both backwards compatibility (for legacy applications) and forwards compatibility (with as-yet unknown applications) can be enabled through composable software.

It is interesting to consider that the platform-independent techniques used by Web Services can significantly reduce the number of software components which need to be directly composable. Exposing object-method functionality via XML-based remote procedure calls (e.g. XML- RPC, SOAP) can provide lightweight client-side access to heavyweight server-side capabilities.

Web Services has been an active area of work for several years. While there is no fixed definition or locked-down architecture, certain capabilities appear common. A summary table follows which presents a possible XMSF Stack for Web Services.

Repositories Locations for providing approved (or ad hoc) Web Services with integrated registry services.	**Administrative** Exemplars: DoD XML Registry, XML.Gov http://diides.ncr.disa.mil/xmlreg/user/index.cfm http://xml.gov/efforts.htm
Services Discovery Centralized access via repositories is made accessible to web-based applications via service publish and search capabilities	**UDDI, LDAP** Universal Description, Discovery and Integration, Lightweight Directory Access Protocol OASIS: http://www.uddi.org IETF: http://www.ietf.org/rfc/rfc2251.txt
Services Description Describe detailed methods and parameter signatures of each service	**WSDL** Web Services Description Language W3C: http://www.w3.org/2002/ws
XML Messaging Express messages in common XML formats for simple encoding and decoding	**XML-RPC, SOAP, XMLP** Remote Procedure Calls, SOAP, XML Protocol http://www.xmlrpc.org , http://www.w3.org/2000/xp/Group
Service Transport Transporting messages between applications. Typically requires reliable (i.e. guaranteed) delivery.	**HTTP, SMTP, FTP, BEEP** Hypertext Transfer Protocol, Simple Mail Transfer Protocol, File Transfer Protocol, Blocks Extensible Exchange Protocol

Table 1. Multiple Layers of Functionality, Composed to Provide Accessible Web Services.

3 Composing diverse system visualizations using Web-based X3D graphics

Identification and establishment of appropriate standards for future M&S and C^4ISR system integration will also be accomplished. Web-based visualization using ISO-approved open-standard Extensible 3D (X3D) Graphics provides new opportunities for visual interaction, insight, and dissemination. The project will further explore syntactic and semantic integration, showing rapid production of customized messaging and network protocols to achieve convenient syntactic integration, together with needed semantic consistency of context-sensitive messages across diverse systems through XML-based tactical ontologies.

The potential impact of Web-based X3D graphics for JSB is cross-cutting and huge. The overall need to understand the tremendous variety of information is alone sufficient to justify a coherent strategy for 3D graphics. We have omitted the necessary analysis from this report for reasons of length. An extensive report detailing the many topics listed in Figure 1 is online at http://web.nps.navy.mil/~brutzman/DeformableSurfacesTechnicalReport2003November.pdf

> The Deformable Surfaces report examines multiple technologies, constraints and strategies for Web-based 3D rendering of dynamic deformation structures in military simulations. The motivating goal is to show how all manner of 3D objects can be modeled, animated and manipulated, in a scalable and repeatable fashion, in support of distributed large-scale virtual environments (LSVEs). Such capabilities have broad training, analysis and operational value.
>
> Web-based 3D graphics are the critical technology needed for rendering the dynamic deformation of structures in distributed military simulations. This is a broad subject area with many specific requirements. The Extensible 3D (X3D) Graphics standard undergoing ISO review includes nearly the full range of capabilities needed. This approach differs from other technical possibilities through adherence to an open standard, royalty-free licensing for any use, availability of both commercial and open-source implementations, provision for alternate software application programming interfaces (APIs), multiple extensibility mechanisms, and a growing content base of compatible 3D models.
>
> Of particular importance is that X3D supports encoding of scenes using the Extensible Markup Language (XML), which ensures that files are self-validating and capable of reliable processing. Since XML is well suited to both Web interchange and database interoperability, X3D using XML encodings provides new capabilities for large-scale production.
>
> Advanced X3D capabilities include geospatial referencing, humanoid animation, shared-state distribution using the IEEE Distributed Interactive Simulation (DIS) protocol, building prototypes, server-produced custom terrain, image overlay of photographic, cartographic or pseudocolor images, integrated physics for entity motion and sensor projection, a variety of user-interaction techniques, and scalable loading/unloading of interrelated scenes.
>
> The multiple challenges involved in modeling deformable surfaces cannot each be solved in isolation. 3D graphics, underlying model representations and networked distribution at first appear to be different topics. Nevertheless, solutions for each area must simultaneously consider constraints and capabilities of other areas. Compatible integration within a Web-based framework allows effective use of deformable surfaces for diverse military simulations. This report presents results emphasizing standards-based interoperability, scalable architectures, demonstrated examples and directions for future work.

Figure 1: Deformable Surfaces report: comprehensive review of Web-based 3D graphics.

4 Recommendations for a Framework ensuring continuous Interoperability, Reusability, and Composability

During a panel discussion during the co-hosted Spring Workshops of the Simulation Interoperability Standards Organization (SISO) and the Society for M&S International (SCS) leading experts in the field of M&S pointed out that while actual M&S standards, such as the Distributed Interactive Simulation (DIS) protocol, the Aggregated Level Simulation Protocol (ALSP) and the High Level Architecture (HLA), all are targeting the implementation level for real interoperation of models, standards on the conceptual modeling level are necessary. In other words, ***to enable meaningful interoperability on the simulation system level, composable models of the conceptual model level are necessary***.

The implication is trivial: if we cannot align the conceptual models, the federation of the systems based on these conceptual models doesn't make sense. There will be no system-of-systems but a muddle-of-systems, in which the exchange of bits and bytes may be possible, however, no meaningful collaboration of the participating components can be achieved. In the best case, this will result in parallel, not connected processes for which the federation building was unnecessary. In the worst case, we will build federations that are wrong due to failures in conceptual model composition, aggregation, and disaggregation, or other issues of multi-resolution model collaboration. Many reports such as the RAND Report on Multi-Resolution Modeling[1] issues are dealing with the challenges to be met so that we don't have to work them out in detail in the context of this report.

The general question is how to formally structure knowledge in form of models in a usable, composable way. The JSB infrastructure must meet this challenge. The technical integration framework must be accompanied by management procedures ensuring reusability and composability of legacy and future simulation solutions. The following three sections are presenting elements of the framework facilitating to reach these goals:

- An engineering approach applicable to existing and future simulation components and systems including migration support for legacy applications(it will be proposed that new systems as well as the migration will be based on the Model Driven Architecture);
- A recommendation for a resources repository comprising applicable components of the JSB including necessary metadata;
- A proposal for the alignment of the necessary management processes instead of enabling reaching only point solutions ensuring continuous interoperability, reusability, and composability of JSB components.

Although the three parts can be implemented independently and gradually, it is highly recommended to establish an overarching integrated product development team (OIPDT) comprising experts from all three sections to ensure consistent and harmonized development.

---

[1] Paul K. Davis and J. H. Bigelow, Experiments in Multi-Resolution Modeling (MRM), RAND Document MR-1004-DARPA, 1998; <www.rand.org/publications/MR/MR1004>

4.1 Benefiting from distributed modeling methods using MDA

The vision of JSB is to become a common integration framework capable of supporting the future emerging simulation needs ranging from training and battlefield rehearsal to research, system development, and acquisition in alignment with other operational requirements.

The necessary functionality will be comprised of components that can be composed to deliver the user driven functionality for a given application. For effective and efficient reuse, the functionality has to be defined unambiguously. The functionality of legacy applications must be documented in the same way that functionality requirements for additional or alternative future solutions are captured. Furthermore, the functionality documentation shouldn't be driven by platform or language specific constraints, but it must be independent from the actual implementation. This kind of view definition is well known to the military community used to the operational view, the system view, and the technical view as defined within the C4ISR Architecture Framework, which evolved into the DoD Architecture Framework.[2] The use of these documents for the purpose of JSB will be elaborated in more detail in section 4.3 on management processes. For this section, it is essential to see the necessity of having a platform- and implementation- independent model of the necessary functionality to discover what functionality is supported by which systems in the overall context of JSB. This collection is often also referred to as the conceptual model of the federation (Please note that the term federation should not imply the use of the High Level Architecture; a federation is much more the collaboration of various participating systems using a common infrastructure).

While the development of such a conceptual model has been seen as an art for quite a long time, the necessity of applying engineering principles to move from the individual solution to an engineering solution that can be shared and discussed in a group, and that is based on rigid standards became obvious in the recent past. Research conducted at the Virginia Modeling Analysis and Simulation Center (VMASC) of the Old Dominion University (ODU) supports the assumption that engineering methods can be applied. Of particular interest is the use of the Unified Modeling Language (UML).[3] Since having been standardized by the Object Management Group (OMG) in 1997, it has become of interest to management consulting firms, business analysts, system analysts, software developers, and programmers. Furthermore, UML can be seen as the standard for blueprints of software solutions. It can be stated that, over the last years, the UML became something like the *lingua franca* for modeling purposes. Furthermore, in summer 2003, version 2.0 became standardized. This new version of UML is overcoming a lot of shortcomings of the first version, among them issues like multiple resolution modeling, composability, and improved time management. In summary, UML has the potential to be used as the standardized approach to create platform- and implementation- independent documentation and specification of functionality.

The popularity and success of UML increased the visibility of the OMG, which began independent operations as a not-for-profit corporation in 1989, founded by eleven companies. Through OMG's commitment to developing technically excellent, commercially viable and vendor independent specifications for the software industry, the consortium today includes over

---

[2] U.S. DoD Architecture Framework Working Group, The DoD Architecture Framework Version 1.0 (Final Draft), January 2003;

[3] Unified Modeling Language Website; <http://www.omg.org/uml>

800 members. The OMG was initially formed to create a component-based software marketplace by supporting the introduction of **standardized object software**. The organization's charter includes the establishment of industry guidelines and detailed object management specifications to provide a common framework for application development. Conformance to these specifications makes it possible to develop a **heterogeneous computing environment** across all major hardware platforms and operating systems, in particular including those that can sustain JSB computational requirements in an environment where its components communicate via Web and Internet technologies. Today, implementations of OMG specifications can be found on many operating systems across the world. OMG's series of specifications detail the necessary standard interfaces for Distributed Object Computing. The OMG has led the way in providing vendor and language independent interoperability standards to the enterprise. Its goal is to enable a global information appliance.

Looking at the requirements of JSB it becomes obvious that OMG proposed recommendations for heterogeneous computing environments based on standardized software building, commercially viable solutions are supporting the vision of the U.S. Air Force. The use of UML as a cornerstone already was proposed. Actually, the OMG is working on a new approach that has the potential to revolutionize the interoperability domain of distributed software development which is not only platform- but also language- and vendor- independent, i.e., even goes beyond the interoperability and inter-platform portability of Java based applications. This approach comprises the established standards and standard development procedures of OMG and raises them to a new level: the Model Driven Architecture (MDA).[4]

The underlying methods ensure that components can be described in a common way, and that the processes of composing these components as well as orchestrating them in a common choreography are commonly understood and standardized. Main objective of the MDA is the ability to derive code from a stable model as the underlying infrastructure shifts over time. In other words, the model of the application is captured in an implementation- and platform-independent language. The specification of this core model is based on the established OMG standards Unified Modeling Language (UML), the Meta-Object Facility (MOF), and the Common Warehouse Metamodel (CWM). The three standards are tightly connected. Knowing the UML, learning MOF and CWM is relatively easy as the similarities outnumber the differences of the three approaches.[5] This core model of the application is defined as the *Platform Independent Model* (PIM). It can be interpreted is a general solution model meeting the operationally driven requirements of the final customer of the system.

When applying the MDA to develop software, choosing the target platform is the next step in the process. While the PIM copes with the general components, algorithms, and data to solve a given problem, the next model in the hierarchy is dealing with the problems of the implementation. This model is defined as the *Platform Specific Model* (PSM). Contrary to the unique PIM, several PSMs may exist to solve a given problem. For the most often used middleware solutions in the domain of the OMG, standards to map a PIM to a respective PSMs

---

[4] Model Driven Architecture Website; <http://www.omg.org/mda>

[5] All standards have references at the OMG website following the scheme http://www.omg.org/<name>. In addition, many IT books are covering the scope. All standards, however, are freely delivered and maintained on the websites open to the public.

are already under development. Respective middleware solutions are, e.g., CORBA, Web Services, XMI/XML, .NET, and Java.

The derivation of source code from the PSM normally can be tool driven. In general, the PSM will be a UML model which takes the specifics of the chosen middleware solution and the target platform into account. The compilation of the models and the assembly and binding can be done automatically as well. The processes are visualized in Figure 2.

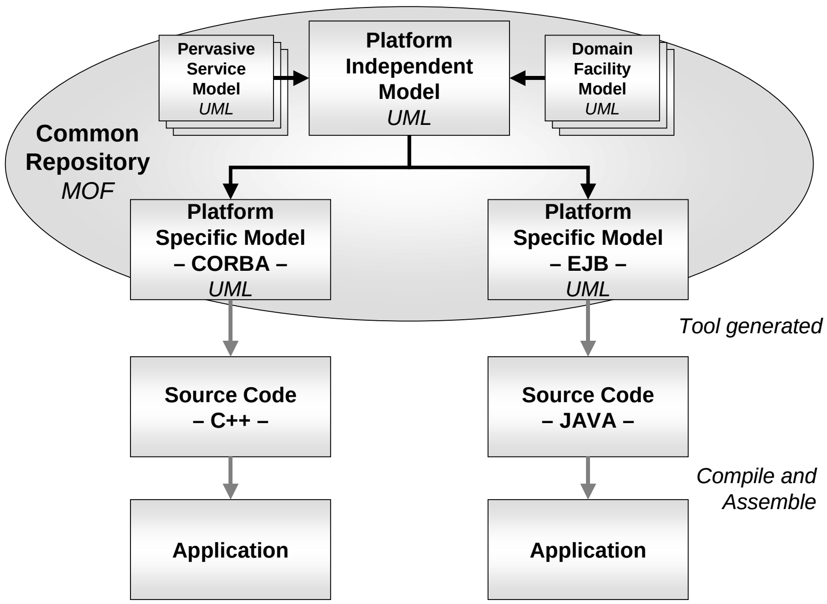

Figure 2: Applying the Model Driven Architecture[6]

This concept directly addresses the composability and reusability issues being addressed within the JSB requirements. If a simulation system is delivered together with a PIM, or a PIM is re-engineered, this ensures several benefits:

- If the underlying infrastructure shifts, the re-implementation on new platforms using new protocols is supported;
- If several solutions for a new component, such as a new flight simulator or C2 simulator, are proposed accompanied by a PIM, the comparison of the functionality is facilitated;
- If a component is to be replaced, the PIM facilitates the development of request for proposals as well as the evaluation of proposed solutions;
- The place of a component within the overarching operational view is easier to find when a PIM is used. Actually, the collection of PIMs can be used to show which parts of the operational battle sphere are already covered by the JSB and where additional work is needed.

[6] Tolk, Andreas, Avoiding Another Green Elephant – A Proposal for the Next Generation HLA based on the Model Driven Architecture (02F-SIW-004, Best Paper Award), Fall Simulation Interoperability Workshop, Orlando, Florida, September 2002

This enumeration shows why the ideas of a Platform Independent Model are crucial for the idea of the common management of projects merging into the JSB, the alignment of these processes, and the establishment of a component repository comprising elements that can be federated on the fly to deliver user driven solutions. The following section will elaborate this in more detail.

Furthermore, another major benefit of the MDA approach lies in the evolving creation of a common repository of standardized functionality the system developer can use to define his system. The *Domain Facility Models* and *Pervasive Service Models*, which are displayed as components of the common repository, are a valuable heritage of the CORBA community. Within the *Object Management Architecture* (OMA) of the OMG, beside the middleware solution itself also a set of general *Object Services* (like naming, finding, etc.) as well as *Common Facilities* is standardized. This set of domain-independent standards that are used by many distributed object programs has evolved into the *Pervasive Service Models* that include directory services, event handling, persistence, transactions, and security. Additionally, due to the wide range of application domains of the OMA, domain specific standards have been developed by *Domain Task Forces* (DTF) of OMG members. Their findings and recommendations evolved into the *Domain Facility Models* representing the common features of all applications in its domain. There are already standards for Telecom, E-Commerce, Finance, Manufacturing, Transportation, Space, Health Care, and more. The websites of the OMG enable access to all recent and active DTF, recommended standards, etc.

To summarize the overview, the MDA is still a very young effort, but it has very strong roots in mature methods and standards. This may explain why the MDA has been named as a key trend in the software industry by PricewaterhouseCoopers in their recently published Technology Forecast for 2002-2004.[7] The forecast reports that the MDA is poised to revolutionize the software design and development process. It is therefore very likely that the MDA will become a success story like the middleware solution CORBA as well as the modeling language UML already are. The MDA approach is likely to help the commercial industry to reach a new level of interoperability within this decade.

These ideas reached the expected level of maturity. In particular, web services are benefiting from actual developments as this emerging market is pushed by the IT industry and consequently new methods such as the MDA are used to promote them. Several CORBA solutions have been successfully transferred to web services already.

Commercial organizations compete in the global market. This competition is characterized by the necessity to organize knowledge in a way that allows management to get access to it to deal with opportunities and risks derived from technological changes as well as actions of the ever-increasing competition. The main means for knowledge transfer nowadays are information systems. Logically, methods to ensure the procurement of intellectual capital are of high value and a lot of efforts are going on in this domain. Object-oriented techniques to facilitate reuse, the use of patterns to capture solutions to recurring problems, and the use of components to actualize reusable parts have emerged. To summarize it, a standardized way to encapsulate knowledge in a reusable form is supported by actual solutions of commercial engineers. The MDA is the actual last step in the evolution. These arguments are one-to-one transferable to JSB requirements. Furthermore, the MDA will support the concept of software components as well.

[7] PricewaterhouseCoopers Technology Center. 2002. Technology Forecast: 2002-2004, Volume 1: Navigating the Future of Software. Menlo Park, California

If such components comprising valuable reusable solutions are defined by means of the MDA, i.e., if they are defined and documented in form of Platform Independent Models, their reuse will be much easier than it is today.

In addition, the whole range of concepts of the MDA – i.e., the common repository comprising the pervasive service models and domain facility models and other reusable solutions – is supporting the objective of composability in the sense that systems can be composed out of existing components (bottom up approach) as well as user requirements can be fulfilled by reusing solutions for the derived challenges (top down approach). The MDA comprises the necessary artifacts to ensure reusability and composability of software components, as the necessary different views are comprised in the set of standards. All needed aspects – static, dynamic, and procedural – can be found in the MDA, be it time constraints or internal states of the system necessary to be considered to reach full interoperable solutions.

To summarize, the application of the MDA to M&S system development will help to overcome many shortcomings of actual applied standards. The applicability of the MDA to M&S already has been shown and applied in a commercially viable manner by IT partners supporting the M&S industry. The ideas published by the Australian based company Calytrix Technologies Pty Ltd are used as an example. Their Integrated Development Environment (IDE) called SIMplicity implements the core concepts of MDA. At the user interface level SIMplicity presents the developer with a modeling environment to specify the *Platform Independent* and *Platform Specific* models for their simulation, applying UML notation wherever applicable. The modeling process supports the developer through the design, implementation, and execution phases of the simulation development life cycle. From the model a code generation engine is employed to automatically create all the integration and component stub-code required to support the simulation design on the targeted Platform Specific middleware. SIMplicity uses the MDA design and development process.[8] M&S components can be derived for various platforms and middleware solutions, such as the various Runtime Infrastructure (RTI) derivatives 1.3 NG or IEEE 1516, or the generation of a necessary Distributed Interactive Simulation (DIS) protocol access layer. SIMplicity has proven to be flexible and useful for rapid integration while every integrated component becomes a reference component in form of a PIM in their repository. Success stories posted on the website and presented on various recent workshop prove the applicability of the MDA idea in the context of M&S infrastructures and common integration framework on the small and mid scale. It is perceived to be worthwhile to evaluate the applicability on the scale of JSB.

It should be pointed out that the main advantage of the MDA is its ability to embrace and integrate former working solutions successfully. Migration is part of the concept. Whether using CORBA based integration, Java driven solutions, web services, M&S middleware solutions using an RTI, or other well-defined methods, they can be transferred and integrated into the MDA method suite. However, guides of best practices for application and extension with regard to the special requirements of JSB are necessary. It is therefore recommended to conduct feasibility studies to deliver the proof of concept and, assuming successful completion and acceptable performance and results, to establish a best practices' guide on the application of methods and tools. These guides must go hand in hand with the establishment of the resource

---

[8] Shawn Parr, R. Keith-Magee, R, Making the Case for MDA (03F-SIW-026), Fall Simulation Interoperability Workshop, Orlando, Florida, September 2003

repository guidelines as well as with the alignment of management processes of which the application of the guidelines is one recommendation, both dealt with in the following sections.

4.2 Concepts for employing resource repositories for broad and consistent composability

The theoretical concepts of MDA only make sense operationally when they first support a common integration framework and second support a set of reusable components that can be "plugged" into the framework. JSB must be understood as a system of systems with a **common infrastructure and a collection of reusable and composable components and common rules** that ensure that interoperability, reusability, and composability are not just point solutions in time but instead a continuous state over the lifecycle of JSB. In addition, migration concepts for legacy solutions are necessary to make the JSB idea commercially viable.

As already pointed out before, composability on the conceptual level is necessary; therefore, a Requirement Component Documentation and Code Repository for JSB will be needed to reach meaningful interoperability on the technical level. It is vital to JSB to understand that there are various levels of interoperability, and that only some levels can be dealt with using technical solutions. The awarded Levels of Conceptual Interoperability Model (LCIM)[9] as well as work being conducted at the University of the Federal Armed Forces of Germany[10] are providing definition and examples. Linguistic perspectives on model communication, looking at semiotic aspects, have influenced both papers. The recommended levels that interoperability has to deal with can be summarized in Table 2.

In other words, to reach meaningful interoperability within the system of systems between the components, we have to ensure that the components

- Are technically connected (technical level),
- Use the same protocols to exchange data (syntactical level),
- Know the context of the data in form of unambiguous definition of the entities, attributes, and relations (semantic level),
- Know how the information will be used when being transferred to a component, i.e., what is done with the information, how does the component behave in various contexts, e.g., does the some information always trigger the same behavior, etc. (pragmatic level), and
- Know the functionality of the component within the common conceptual view of the world to ensure that assumptions and constraints are taken into account respectively, i.e., are there relevant relations that are not coped with in any component implementation – because it made no sense for the components, but it will make sense for the new combination of components – (conceptual level).

The documentation of JSB components must take this into account. Most actual standards are mainly targeting the technical and syntactical level; only a few of them are targeting the semantic level and the support of the pragmatic and conceptual levels is in its infancy.

However, many technologies are available to support a repository as effective and efficient as possible. This paper enumerates important ones that have to be evaluated in detail and specified

---

[9] Andreas Tolk, James Muguira, The Levels of Conceptual Interoperability Model (LCIM) (03F-SIW-007), Fall Simulation Interoperability Workshop, Orlando, Florida, September 2003

[10] Marko A. Hofman, Essential preconditions for coupling model-based information systems, Proceedings of the NATO M&S Group Conference on C3I and M&S Interoperability (MSG-022), RTO-MP-123

how they contribute to a common component repository in follow-on activities. The following list is therefore neither exclusive nor complete, it is a first draft to show that the JSB repository must comprise more than interface specifications.

Level of Interoperability	**Description**	**Examples**
Technical Level	Connectivity is established allowing bits and bytes to be exchange	Physical Connections, Network Layers
Syntactical Level	Data can be exchanged in standardized formats	HLA OMT, CORBA IDL, DIS PDU, XML, WSDL[11]
Semantic Level	Data and its contexts (Information) can be exchanged	Reference Data and Object Models, such as the C2IEDM[12] are managed XML Tag Sets
Pragmatic Level	Information and its use and applicability (Knowledge) can be exchanged	Reference Models of Processes, UML documentation, DEVS[13]
Conceptual Level	Common view of the world (system-of-systems wide conceptual model) and the place of the model within this view can be exchanged	Common conceptual models using engineering based standards, such as UML and DEVS

Table 2: Levels of Interoperability

4.2.1 Data Management and Alignment based on Reference Data Models, such as the Command and Control Information Exchange Data Model (C2IEDM)

Based on the recommendations formulated in recent studies, a lot of work on finding a common ontology is going on. They are targeting the semantic level of interoperability. It is often argued that the use of the Real-time Platform Reference Federation Object Model (RPR FOM) is sufficient, but linguistic studies show that this is not the case. Furthermore, the RPR FOM is very M&S centric, i.e., the integration of Command and Control systems is only possible with very hard constraints. Alternatives, such as the approach defined in the paper of Perme and Long[14], using various layers and data mapping technologies to cope with the alignment of information needed in Command, Control, Communications, Computers, Intelligence,

[11] High Level Architecture Object Model Template (HLA OMT); Common Object Request Broker Architecture Interface Description Language (CORBA IDL); Distributed Interactive Simulation Protocol Data Unit (DIS PDU); Extensible Markup Language (XML), Web Service Description Language (WSDL)

[12] Command and Control Information Exchange Data Model (C2IEDM)

[13] Discrete Event Specification (DEVS)

[14] David F. Perme, Richard Long, Post AOC, Air Tasking Order Release Decision Flow (02F-SIW-042), Fall Simulation Interoperability Workshop, Orlando, Florida, September 2002

Surveillance, and Reconnaissance (C4ISR) systems, e.g. to generate Air Task Orders and support the necessary processes in a collaborative way, are perceived to be much more promising for JSB than using pure M&S approaches. However, layered and mapping approaches need common reference models to cope with the various ways to cope with information on the semantic level.

Outside the U.S., more and more programs are using the long years of experience of the NATO data modeling groups that led to the formulation of the Command and Control Information Exchange Data Model (C2IEDM), formerly known as the Army/Allied Tactical Command and Control Information System (ATCCIS) Data Model or the Generic Hub.[15] This a very mature data model, in particular the new partners of NATO as well as the Partnership for Peace nations are interested in using these efforts to improve their development of NATO standard compliant systems. But also in the U.S. work is going on in this domain, some based on C2IEDM, some based on major U.S. Command and Control Data Models; and data alignment studies of the past have proven that a lot of these efforts can be applied to U.S. as well as C2IEDM solutions. The important fact is that all of these efforts are trying to use C4ISR hubs as kernels for M&S interoperability solutions. Although the project is still going on, the C4I-M&S Reference Object Model (C-ROM) will be used as an example of how future projects can benefit from respective common ontology efforts.[16]

Similar efforts are going on in Europe as well. These efforts started even before the ones referenced above. Starting with a pre-feasibility study for NATO on a kernel application for an Allied Command Europe (ACE) integrated database, presented during the Joint Warrior Interoperability Demonstration (JWID) 1999, the Data Mediation Team of the *"Industrie-anlagenbetriebsgesellschaft mbH"* (IABG) started to use the ATCCIS data model to harmonize various data models in order to derive standardized shared data elements (SDE). Since 1999, in addition to the Link 1, Link 11, and Link 16 messages, the Over-the-Horizon (OTH) Gold messages and the Allied Data Publications (ADatP) of the German Navy were mapped to the Shared Data Model to instantiate the Core Data Model of the German Navy. Furthermore, various models developed under the sponsorship of the German Army Office (*"Heeresamt"*) with various resolution and aggregation levels were mapped to the model as well.[17]

Within the United Kingdom, the Ministry of Defense set up the UK MOD Central Data Management Authority (CDMA).[18] This is a tri-service /civil service organization responsible for ensuring data is defined consistently across the MOD to enable information systems interoperability. The main task of the CDMA is data management, i.e., conducting the process leading to agreed data definitions for information elements stored in a common Data Defense Repository (DDR). The CDMA results from a study on the "Way Forward for Defense Data Management" conducted on behalf of the UK MOD were presented and published in August 2000.

---

[15] C2IEDM is the NATO Standard AdatP-32, most comprehensive information is collected on the ATCCIS Website support by MIP; <http://www.mip-site.org/ATCCIS/ATCCIS_Home.htm>

[16] Brian A. Haugh, Francisco Loaiza, Richard Morton, Steven P. Wartik, The Army C4I-M&S Reference Object Model: Phase I Development (03S-SIW-111), Spring Simulation Interoperability Workshop, Orlando, Florida, April 2003

[17] LTC Bernd Zimmermann, Integrated Army Modeling and Simulation Data Network, German Army Office, Proceedings of the NATO M&S Group Conference on C3I and M&S Interoperability (MSG-022), RTO-MP-123

[18] Website of the CDMA; <http://www.cdma.mod.uk>

In summary, the use of the C2IEDM as a common reference model on the semantic level can be seen as a promising way to go in particular in joint and combined environments.

4.2.2 Base Object Models (BOM)

To increase simulation interoperability by developing a component reuse methodology, several international experts are working on the concept of Base Object Models. Most of these efforts have been conducted in preparation for or during several recent SIW supported by SISO. Actually, a Product Development Group (PDG) is standardizing the results and the results will be published soon (Drafts are available via the SISO BOM PDG). The main ideas are best described in the overview paper by Paul Gustavson et al.[19]

A Base Object Model (BOM) is defined as a simulation component representing a single aspect of federation interplay that can be used as a building block of Federation Object Models (FOM) and Simulation Object Models (SOM). It is documented as a set of data elements as required by the High Level Architecture Object Model Template (OMT) augmented with additional meta-data such as requirements, conceptual model data, development and use history, and other aspects facilitating the exploitation of its potential for reuse. Whereas a FOM generally deals with the multiple object model set, a BOM focuses on a single, ideally atomic aspect. The general idea is that this will facilitate the composition of necessary interplays. To this end, the BOM must be a complete model of the simulation interplay activity, covering all aspects of systems' interoperability (including timing, internal effects, etc.) The resulting schemata are captured using XML. The tag set used is part of the actual standard proposal.

In summary, the BOM methodology provides an appropriate building block mechanism useful for advanced distributed simulations as well as advanced distance learning and other distributed applications using M&S functionality. The developed concepts are definitely applicable to deal with issues on the syntactical level, and within limits can handle issues at the semantic and pragmatic levels. An evaluation of these approaches and contributions to the theory of composability and reusability is recommended, in particular concerning the contribution to the technical specification of the components of the resource repository.

4.2.3 Simulation Reference Markup Language (SRML)

Several major projects within Boeing have adopted a flexible modeling and simulation approach based on an XML schema known as the Simulation Reference Markup Language (SRML).[20] The approach stemmed from several years of developing large-scale availability simulations that included thousands of interconnected objects. With SRML, Boeing has been able to more effectively manage the complexity of models as their features increase. SRML provides the schema that allows simulation markup to be included in any XML document. Using XML, a

---

[19] Paul L. Gustavson, John P. Hancock, Christopher Stapleton, The Base Object Model (BOM) Primer: A Distilled Look at a Component Reuse Methodology for Simulation Interoperability (01S-SIW-086), Spring Simulation Interoperability Workshop, Orlando, Florida, March 2001;
This paper and related information are online available at <http://simventions.com/boms>

[20] SRML has been submitted to the World Wide Web Consortium (W3C) for standardization. The specification is available at <http://www.w3.org/TR/SRML/>. Prototypical implementations and the development software necessary to install SRML for evaluation can be downloaded (with some restrictions) for the website <http://www.boeing.com/assocproducts/srml/>.

modeler constructs a domain-specific schema and a corresponding BOM document, which also references the standard BOM schema. To encapsulate the BOM behavior, the model also references the SRML schema and then specifies the model behavior in the BOM document using SRML elements. The SRML schema specifies a minimal set of elements that provide object-oriented constructs for implementing identity, modularity, classes, associations, behavior, communication, inheritance, polymorphism, and extensibility--which form the foundation of reuse within highly sophisticated models. Just as in using the Hypertext Markup Languages (HTML), SRML is programming language independent through language plug-ins like JavaScript. The plug-in architecture of the simulator also integrates the behavior of pre-compiled components. When combined with an interoperability layer provided by HLA, SRML enables BOM to operate in a distributed environment through a universal simulator.

Features provided by most simulation engines are well understood, thus making it possible to define a standard set of primitive services for a universal simulator. For example, engines fundamentally provide item management, event management, random number generation, mathematics and statistics, and plug-ins for everything else. The Simulation Reference Simulator provides all of these services within a runtime environment that intrinsically uses the Document Object Model (DOM) in item management, and plug-in compilers for behavior. Event management services include functions to Send, Post, Broadcast, and Schedule events among local and remote items.

SRML and BOM are connected, but independent concepts. It is possible to use BOM without SRML and the other way. However, the development teams are meshed and BOM and SRML are well aligned. SISO recently launched a Study Group evaluating if and how SRML can be standardized. SRML completes BOM and is applicable to define reference implementations for components to be used to handle issues on the semantic and pragmatic level, in particular when continuously applied and used as a standardized way to cope with the behavior representation of components in a common context. SRML can be perceived as an executable way of documentation of dynamic and agile behavior.

4.2.4 Unified Modeling Language (UML)

The use and recent developments of UML already have been dealt with earlier in this paper when motivating why UML builds the kernel of the MDA and is the backbone of its Platform Independent Models (PIM). In the context of describing components for the resource repository, UML uses an already standardized way to cope with documentation of the underlying model. It should be pointed out that this model is not limited to IT supported models, but conceptual models and operational models, i.e., views of the Warfighter, can be supported as well, although no 100% solution can be reached without enhancements.[21]

The U.S. Navy presented a very valuable approach during the recent Fall SIW.[22] The U.S. Navy has established a Probability of Raid Annihilation Assessment Process to be used for each new

---

[21] Andreas Tolk, Beyond Technical Interoperability – Introducing a Reference Model for Measures of Merit for Coalition Interoperability, 2003 Command and Control Research and Technology Symposium, Washington, D.C., June 2003

[22] Donna W. Blake, Carolyn Little, Judy Morse, The Navy's Probability of Raid Annihilation Assessment Process Standards & Architecture and Systems Engineering Concept Model (03F-SIW-057), Fall Simulation Interoperability Workshop, Orlando, Florida, September 2003

ship class. This process provides the ship class results to meet OPEVAL requirements across ship classes in a consistent and adequate manner. In particular, the use of use cases describing the real operational environment to set up a UML based description of the synthetic operational environment to be used as a basis of documentation, discussion, and alignment and harmonization of otherwise independent programs and processes is worth mentioning.

For JSB, it is highly recommended to follow this approach as well. Based on the Platform Independent Models of participating simulators, simulations, and other systems, the union of PIMs must gradually evolve into a common description of the JSB mission space. This description enables alignment on the pragmatic, and with limitations, even on the conceptual level and would be the first effort on such a big scale. Feasibility studies may strengthen the credibility of this proposal before implementing tools and procedures for the whole program.

In addition to these efforts, it may be of interest that OMG is working on prototypes of executable UML (xUML) version, where you can execute your model before you implement it. This xUML is embedded into the MDA concepts.

4.2.5 Applying the DoD Architecture Framework

Using UML as a powerful *lingua franca* for modeling makes sense for engineers in general and in particular for programming experts. However, UML can become a gap between the Warfighter and the engineer. To cope with the Warfighters expertise using his language is always preferable to using an engineering artifact the Warfighter is not used to. A possible approach was chosen by the integrated development team under the aegis of SAIC conducting the Active Layered Theatre Ballistic Missile Defense (ALTBMD) study for the Theater Missile Defense Command and Control branch of the NATO C3 Agency.[23] The idea was to use the views defined in the C4ISR Architecture Framework (respectively defined in the NATO C3 System Architecture Framework) to define the functionality of the missile defense system architecture to be evaluated. The users were able to define their requirements and see them directly mapped to operational necessities captured in the operational views. OMG is actually working on a generalized model to map the DoD Architecture Framework products to UML artifacts.

The general use of this approach for necessary VV&A processes was addressed in the VV&A Forum of the SIW in Spring 2003.[24] This approach was furthermore used to support the U.S.JFCOM for the definition of a common framework of future experimentation planning, although due to budget constraints the recommended system was not installed yet. However, there are additional efforts going on to map the operational views, system views, and technical views used in the DoD Architecture Framework, which directly evolved from the C4ISR Architecture Framework, to the Unified Modeling Language.

---

[23] Simon Adshead, Thomas Kreitmair, Andreas Tolk, Definition of ALTBMD Architectures by Applying the C4ISR Architecture Framework (01F-SIW-112), Fall Simulation Interoperability Workshop, Orlando, Florida, September 2001

[24] Andreas Tolk, Susan Solick, Using the C4ISR Architecture Framework as a Tool to Facilitate V&V for Simulation Systems within the Military Application Domain (03S-SIW-029), Spring Simulation Interoperability Workshop, Orlando, Florida, April 2003

4.2.6 Extensible Mark-up Language (XML) / XML Metadata Interchange (XMI) Format

The general usefulness of XML doesn't have to be evaluated further. In particular, with the ongoing developments in the domain of binary XML, many efficiency problems of XML based solutions will be overcome. However, it should be pointed out that XML itself is not the Rosetta Stone for which it was so often sold in the recent past. It cannot be stressed enough that XML itself is nothing more or less then a very flexible transfer mechanism that does not address the need for common semantics of data. It can be used for standardized schema transformation, but not for schema integration without respective management efforts. Therefore, the conclusion of some people, that with the introduction of XML the elimination of ongoing data standardization programs is justified and respective efforts should be replaced by the use of XML tag repositories, is wrong. We should understand the difficulty of creating standard tags. Standard tags are nothing but another way to write down a common reference model. The management process for building a federation can and has to be used to create the respective standards tags that XML can use to exchange the data elements between the heterogeneous systems or system components. Some implications for the necessary Net Centric Data Management processes and what have to be added to be able to cope with M&S requirements and other agile components to ensure consistency on the semantic, pragmatic, and conceptual levels were only recently addresses.[25] XML and XML repositories well definitely have their role in JSB. However, the danger exists that superficial interoperability by applying unaligned XML data exchange will backfire on the user, as XML itself only contributes to interoperability on the syntactical level.

A standard which enormous potential for the M&S community and for JSB hasn't been fully recognized yet is the XML Metadata Interchange (XMI) format. In summary, XMI enables to communicate UML models via XML. XMI enables merging the power of UML on the conceptual and the pragmatic level with the efficiency of XML to bridge the syntactic and the technical level. It is highly recommended to evaluate this standard and its applicability to solve the JSB challenges to improve interoperability, reusability, and composability.

As stated before, XML is only dealing with interoperability up to the syntactical level. Management processes must ensure everything above. An alternative is the extension of the language, which generally contradicts the idea of using open standards.

4.2.7 Web Service Description Language (WSDL) / Universal Description, Discovery, and Integration (UDDI)

As web services and their potential have been dealt with before intensively, in this section the necessity to enhance the current capabilities should be stressed. To avoid misunderstandings: web services are clearly perceived as the way to go, however, the state of the art is focusing on the technical and the syntactical level. WSDL and UDDI are focusing on the exchange of static information (data in context) based on web services with behavior descriptions. They assume that web services are independent sub-tasks that independently contribute to the solution. In complex systems defined by the dense mesh non-linear relations between the entities this is seldom the case. WSDL and UDDI are as powerful as the federation agreements within HLA federations without the technical limitations, however, time management, services alignment,

[26] Extending the overview on reuse of components given by Shawn et al., see footnote 8.

multi-resolution problems, and all the other challenges still open are not solved by using web services instead of another technical approach.

It is therefore recommended to evaluate the necessary extensions of WSDL and UDDI and actively drive the standardization process as no other community has such hard requirements as the JSB community has. It is furthermore recommended to embed this effort into the broader MDA approach using web service driven solutions on the Platform Specific Model level.

4.2.8 Discrete Event Specification (DEVS)

Although many specifications for M&S systems exist the Discrete Event Specification (DEVS) introduced by Zeigler can be seen as something special because it is an overarching approach covering all levels of interoperability by using a system-theoretic worldview from which the necessary technical solutions on lower levels are derived. It describes the component-based characterization of dynamical systems in discrete and continuous forms. It furthermore comprises methods for modular, hierarchical model composition targeting reusability and composability issues. The main disadvantage of DEVS is that it is M&S specific and not well known in other domains. It seems worthwhile to evaluate the possibility to merge DEVS with embracing concepts like the MDA, in particular supporting and defining the necessary pervasive services for JSB in an academically sound and commercially viable context.

4.2.9 Extensible M&S Framework (XMSF) Profiles

Under the umbrella of SISO, another study group recently was launched which definitely is of interest to the JSB community. Based on the positive and promising results of early XMSF demonstrations, a standard to define and document applications that can be used in the XMSF context has to be developed. In the case that JSB utilizes XMSF as an enabling technology and method for the common integration framework, the use of these so-called XMSF profiles must become mandatory. At this point, the study group has developed the following definition:

XMSF profiles are formal technical specifications for application of interoperable web based technologies enabling composable and reusable modeling and simulation, and facilitating enterprise integration. The objectives of XMSF profiles are to

- *Provide unambiguous specification of the functionality of components, and interfaces among 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*

As the study group only recently was established (Fall SIW 2003), it is explicitly recommended to participate in these efforts actively and shape the definitions and recommendations with regards to the specified JSB requirements. Supporting XMSF profiles is perceived as shaping the future of the next generation of distributed simulation systems.

The following figure shows a high level view of some of the aspects of the repository needed for the JSB vision as stated in this report. Although the figure already is quite busy it is neither complete nor exclusive and is meant to give a first overview of necessary aspects of a repository beyond technical aspects. The danger of applying all these various techniques and methods is that the result is a patchwork instead of an orchestrated approach. This observation directly leads over to the necessity to align the accompanying management processes.

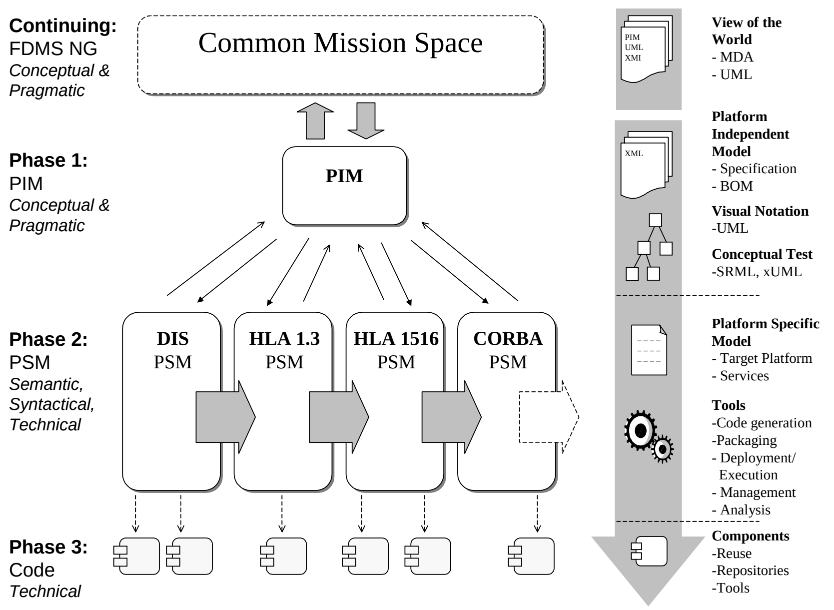

Figure 3: High Level View on Aspects of the Resource Repository[26]

4.3 Alignment of Management Processes

After presenting a subset of the actual applicable concepts for employing resource repositories for broad and consistent composability in the last section, some of them already accompanied by tentative recommendations, the purpose of this section is to bring them together in a form applicable to meeting the JSB requirements. However, before presenting the recommendation, some additional analyses conducted concerning comparable activities in domains of relevance will be described to motivate the recommendation by giving some background information.

4.3.1 Necessity for Common Data Engineering

Common to most actual interoperability solutions is that the system designer tasked with the integration has to know *what* data is located *where*, the *meaning* of data and its *context*, and into what *format* the data have to be transformed to be used in respective distributed applications within the overall system. To generate the answers to these questions is the objective of data administration, data management, data alignment, and data transformation, which can be defined as the building blocks of a new role in the interoperability process: the tasks of *data engineering*. These four steps are necessary first achievements in a broader interoperability framework to ensure that the integration framework meets the requirements of the JSB. The terms are defined as follows:[27]

- *Data Administration* is the process of managing the information exchange needs that exist within a group of systems, including the documentation of the source, the format, context of validity, and fidelity and credibility of the data. Data Administration therefore is part of the overall information management process.
- *Data Management* is planning, organizing and managing of data by defining and using rules, methods, tools and respective resources to identify, clarify, define and standardize the meaning of data as of their relations.
- *Data Alignment* ensures that the data to be exchanged exist in the participating systems as an information entity or that the necessary information can be derived from the data available, e.g., using the means of aggregation or disaggregation.
- *Data Transformation* is the technical process – often implemented by respective algorithms within the gateways and interfaces – of aggregation and/or disaggregation of the information entities of the embedding systems to match the information exchange requirements including the adjustment of the data formats as needed.

We recommend that data transformation for JSB should be supported by XML based solutions. Agreeing to use XML, data management is mainly focusing on the definition of a standard tag set and the mapping of existing tag set to this standard. The namespace management actually introduced to the U.S. Armed Forces can be seen as a necessary first step into this direction. Based on the results of data transformation and data management, data alignment will be tremendously facilitated, as it becomes part of the data management process. The same is true for data administration.

It is recommended to establish a data management agency for JSB following the example of the NATO Data Administration Group (NDAG) or other national data management agencies. The use of the C2IDEM as the core model not only provides a good start, it also facilitates the integration of command and control systems in the long term. If necessary, a study can help to define the data management process inthe form of a Data Management Guideline.

4.3.2 Defining a Common Mission Space

In order to reach interoperability on all levels defined in Table 2, it is necessary to define a common mission space for JSB as the digitized equivalent of the battle sphere to be covered in JSB supported operations. This task may not be accomplished in one step but the common mission space can be developed gradually. Based on the described concepts for employing resource repositories for broad and consistent composability the development of this common mission space is envisioned as follows:

- The Warfighters view of the mission space is defined using the operational views as described in the DoD Architecture Framework. It is recommended to use at least the

[27] Andreas Tolk, Common Data Administration, Data Management, and Data Alignment as a Necessary Requirement for Coupling C4ISR Systems and M&S Systems, (paper accepted to be published in) International Journal on Information & Security, Volume 12

system views to validate that and how the operational requirements are met by the systems delivering the necessary functionality.

- Within the C4ISR community, the use of so-called C4I Support Plans is mandatory for systems contributing to the C4ISR processes in the battle sphere. This allows the evaluation of the contribution of the special system to the overall process, which requirements are fulfilled by this system, etc. In the same way, components of JSB should support the common mission space by a similar plan.
- Based on standardized mappings and transformations as worked on by the OMG, these views are mapped to UML representations. These UML views should have similar objectives as the U.S. Navy approach defined earlier. However, the JSB view should go one step further, namely to define the common mission space based on UML as a Platform Independent Model approach supported by the concepts of the MDA.
- Legacy systems will contribute to the common mission space in bottom-up manner by re-engineering the PIM and merging its contribution to the mission space. Operational requirements contribute to the development in a top-down manner, defining necessary functionality in the form of a PIM that has to be implemented (a) by a new system specified by the PIM, or (b) by a composition of existing solutions who's union of PIMs covers the operationally required part of the mission space. As UML 2.0 supports the concept of multi-resolution schemas, simulators and simulation systems of different granularity and resolution can be coped with in a consistent manner. Using XMI, the PIMs can be communicated efficiently between various instances and components of JSB as well as between services, partners, allies, and agencies.
- The implementation will generally be web enabled and will integrate the data engineering results described before. However, as the PIM is platform and middleware independent, alternative implementations, such as DIS federations, HLA federations, or XMSF profiles for web based distributed simulation, are possible.

As pointed out before as well as in various papers dealing with this aspect, the existence of a common conceptual picture of the mission space is unavoidable for interoperability of all levels.[28] It is also required for common validation and verification. The application of formal M&S methods such as DEVS can further contribute to the fidelity and credibility of the processes and the expected results.

4.3.3 Defining the Resource Repository

Obviously, the resource repository must be more than a simple collection of components that are technically connectable to the integration framework used for JSB. Independent from the chosen solution, the components have to be accompanied by explaining metadata, which comprise information about the component. Based on actual code of best practices, the following data should be stored in the repository:

---

[28] The same ideas have been presented in a JSB related domain by Judith Dahmann and Captain (USN) Jeff Wilson in their paper "MDA and HLA: Applying Standards to Development, Integration and Test of the Single Integrated Air Picture Integrated Architecture Behavior Model," (03F-SIW-044), Fall Simulation Interoperability Workshop, Orlando, Florida, September 2003

- The executable component including the Platform Independent Model and the Mission-Space-Support-Plan as defined above.
- Description of the necessary data, where this data normally can be obtained, validity constraints, etc.
- Validation, Verification, and Accreditation results, documents, etc.
- Metadata on pedigree, reference applications, fidelity, credibility, reliability, user acceptance in past applications, etc.
- SRML version of the component as sort of a reference implementation enabling the test of the functionality before downloading or obtaining the real executable for the purpose of analysis, training, experimentation, or mission support.

It is not necessary that all information must be stored within the repository, but often references are applicable as well. It just has to be assured that these references are up-to-date and valid and that the metadata is up-to-date as well (e.g., when source code of simulation systems are updated that the PIM and the Mission-Space-Support-Plan are updated as well). Furthermore, this list is just a core and definitely extensible.

4.3.4 Setting up an Overarching Integrated Product Development Team (OIPDT)

The requirements of JSB concerning reusability, composability, and interoperability can only be met be a harmonized approached in which technical possibilities are used based on aligned management processes. In order to establish a solution that is not only technically but also organizationally feasible and commercially viable and supportable, the definition has to be conducted by a heterogeneous team comprising

- Technical experts from industry, government, and academia,
- Project managers from industry and government,
- Project management experts from academia,
- Decision makers from government and industry.

It is obvious that not all team members have to join every meeting, but it is important that the members of the JSB community achieve the highest technical skill level that can be integrated into the participating organizations. Furthermore, commitment of the decision makers is essential form both sides, which includes commitment inthe form of funding and R&D.

The outline of the problems to be solved was summarized in a winter simulation conference paper 2000 by leading experts in the field of M&S.[29] As Davis et al. state, five components are key to the success of a repository as described in this approach:

- **Representation:** We need to address the type of representations that we will accept for components. The PIM approach of the MDA seems to be a viable and working solution from experiences so far.
- **Requirements:** Without knowing the requirements for a simulation, we are hard-pressed to invent compositions. The use of the DoD Architecture Framework to guide the mapping from operational necessities to required functionality inthe form of PIM support this.

[29] Paul C. Davis, Paul A. Fishwick, C, Michael Overstreet, C. Dennis Pedgen, Model Composability as a research investment: responses to the featured paper, 2000 Winter Simulation Conference, pp. 1585 - 1591

- **Marketplace:** Only if there is a business case, will industry invest in writing and selling components. The use of web services embraced by the MDA in combination with the intent of the U.S. Air Force to make JSB a reality delivers this business case.
- **Plug and Play:** Components have to hook together perfectly. The MDA approach accompanied by data engineering and management mandates to participate (possibly in the form of no play no pay) can technically and organizationally support this. The common mission space helps to perceive and gradually close gaps through migration and new developments.
- **Finding Components:** The web should be the central vehicle for implementations of composability. The approach presented here as well as the management mandates concerning the population of the repository do ensure that based on requirements, existing solutions can be found, gaps can be identified, and new developments can be specified accordingly.

In summary, the recommendations given in this approach meet the key components perceived to be necessary for a successful approach as envisioned by the JSB community. However, it should be clear by now that whatever technical implementation will be chosen for the integration framework and the resource repository for JSB, if we do not align the accompanying management process the only thing reached will be another interoperability problem on a new technical level.

5 An Example for a Technical/Institutional Transformation Plan

In general, the Naval Postgraduate School (NPS) as well as the Old Dominion University (ODU) are mainly interested to support the JSB efforts by escorting R&D efforts and consultant activities. Both universities are convinced that professional software development should be part of the participating industry partners, which does not exclude the development of prototypical software solutions or conducting feasibility studies to proof the concept of feasibility or to evaluate the applicability of new methods and ideas.

Based on this general observation, are proposing to conduct the following activities to orchestrate the various prototypes and recommendations for the JSB activities:

1. **Establishment of an independent technical advisory board** to support the JSB team with technical expertise and evaluation of recommendations, testing and evaluation of prototypes, etc. In particular the correct and consistent application of recommended open standards should be evaluated by the universities. NPS is setting up a simulation center comprising the main joint and service simulation systems. VMASC of ODU already established its Decision Support Battle Lab which already has been successfully integrated into the Joint National Training Capability (JNTC) infrastructure of the Training Directorate J7 as well as into the Distributed Continuous Experimentation Environment (DCEE) of the Experimentation Directorate J9, both part of the Joint Forces Command. Furthermore, connectivity between NPS and ODU facilities has been established and distributed applications successfully demonstrated. The C3I Center of

George Mason University is already part of this distributed testing and research unit. The XMSF group in general and NPS and ODU in particular established an international reputation in the application of the open standards described in the section of this paper. It is therefore recommended to make use of this expertise for the JSB development.

2. **Evaluation of the management processes** that have to be aligned to support the JSB development and the migration of legacy applications. Independently from the open standard family to be chosen for the implementation of the JSB infrastructure as well as the corresponding repository efforts, the alignment of management processes and the establishment of best practice guidance documents is mandatory to establish continuous interoperability instead of technical point solutions in time as well known from past efforts. It is therefore recommended to fund an expert study group coping with alternatives in an initial phase (preparing a decision document which way to go) and writing the guidance documents by escorting respective integration efforts in the industry in a second phase. A limited experiment, e.g. coupling a legacy simulation system such as AWSIM with a legacy simulator on the basis of the MDA and establishing two repository entries in an integrated environment such as SIMplicity could help to cope with general issues and would result in convincing prototypes.

3. **Establishing the XML namespace management for JSB** must be a high priority. It is recommended to follow the path described in the previous section, namely to use the XML version of the NATO Data Administration model C2IEDM is the core to establish a Data Management Agency (DMA) for JSB. Every data exchange requirement has to be formulated in XML (allowing the application of white box principles without having to open the source code, which means that industry interests will be saved without sacrificing the requirements for white box systems on the side of the JSB user). The DMA enhances and extends the core model based on international standards (well defined rules for C2IEDM are established) and generated the necessary XSLT schema. NPS as well as ODU have knowledge of the international usage of C2IEDM, including its use in the Multinational Interoperability Program (MIP), which uses C2IEDM to couple the C4I systems of the Alliance.

4. It is furthermore recommended to make use of the **Study Groups of the Simulation Interoperability Standards Organization** by funding core people to participate in activities such as the XMSF profile study group or the conceptual model study group. As SISO is establishing the standards necessary for JSB it is highly recommended to actively shape the development of these standards instead of using proprietary solutions or having to adapt the JSB solution to not aligned SISO standard on the long term.

5. The XMSF Working Group of the **Web3D Consortium** is another forum to influence (and take advantage of) broad and rapid advances in 3D graphics, applied in ways that are actionable by the huge data-driven capabilities of the JSB. Further public partnered collaborations among major standards organizations (OMG, Open GIS, SISO, Web3D and even W3C) will stabilize requirements for XML-based interoperability, likely even influencing government policy as benefits and capabilities become impossible to ignore.

6 *Summary and General Conclusions*

Both universities are convinced that JSB can and must be established on the basis of the rigorous use of open standard. Exceptions are only tolerable for niche solutions not being of general interest to the JSB community. The level of maturity that actually is reached by standards such as XML, XSLT, WSDL, XMI, MDA and others is sufficient for industrial use on the big scale. There is no good reason to assume that the requirements of JSB cannot be met by them with the same degree of effectiveness and efficiency. However, without rigid control and evaluation of accordance of recommended solutions to the family of JSB standards this vision will never become reality. Furthermore, technical solutions are insufficient for solving conceptual problems, which means that overarching concepts have to be applied based on well defined guidelines. Finally, as important as orchestration of technical solutions and standards is, only the alignment of applied management processes will ensure the continuous interoperability envision by the JSB concept.

Based on their experience made within the various XMSF projects, many of them referred to in this paper and more obtainable via the XMSF website, NPS and ODU are convinced that the orchestration of industry partners based on open standards and non-intrusive white box solutions are acceptable, feasible, and capable to fulfill the requirements of JSB. The experts of the XMSF group and their partners are eager to support and consult the JSB core team to make the vision of JSB become reality.