Linked Heterogeneous BML Servers in NATO MSG-085

Dr. J. Mark Pullen and Douglas Corner
C4I Center
George Mason University
Fairfax, VA 22030, USA
+1 703 993 3682
mpullen@c4i.gmu.edu
dcorner@c4i.gmu.edu

Thomas Remmersmann and Dr. Irmtrud Trautwein
Fraunhofer Institute for Communication, Information Processing and Ergonomics
Fraunhofer Str. 20
53343 Wachtberg Germany
+49 (0)228 9435-522
thomas.remmersmann@fkie.fraunhofer.de
irmtrud.trautwein@fkie.fraunhofer.de

Keywords:
Server Architecture, Command and Control Simulation Interoperability

The Coalition Battle Management Language (C-BML) as currently implemented requires that interoperating Command and Control (C2) and Simulation systems exchange information via an interface to a Web service, using a clearly defined schema. Continued progress of Coalition Battle Management Language (C-BML) in NATO MSG-085 (Standards for Command and Control-Simulation Interoperation) has resulted in a need to extend the scope possible by doing this with a single server. The extension could be either for purposes of geographic distribution or for extended performance. In preparing for the MSG-085 final demonstration, we have developed an example of such a distributed server, achieved by interoperation between the WISE-SBML server and the FKIE BML server. This paper reports on the benefits of server interoperation, the mechanisms used and issues encountered in implementation, and the resulting capability.

1. Overview

Battle Management Language (BML) and its various proposed extensions are intended to facilitate interoperation among command and control (C2) and simulation systems (C2-Sim) by providing a common, agreed-to format for the exchange of information such as orders and reports. In recent implementation, this format has been accomplished by providing a repository service that the participating systems can use to post and retrieve messages expressed in BML. The service is implemented as middleware that is essential to the operation of BML and can be either centralized or distributed. Recent implementations have focused on use of Extensible Markup Language (XML) along with Web service (WS) technology, a choice that is consistent with the Network Centric Operations strategy currently being adopted by the US Department of Defense and its coalition allies [1].

SISO has a two-part standards effort supporting BML. The Military Scenario Definition Language (MSDL) standard [2] was approved in 2008. It is intended to reduce scenario development time and cost by enabling creation of a separable simulation independent military scenario format, focusing on real-world military scenario aspects, using the industry standard data model definition eXtensible Markup Language (XML) as input to initialize C2 and simulation systems. The Coalition BML (C-BML) standard begun in 2005 [3] provides the tasking and reporting aspects of C2-Sim. It was balloted in 2012 and is expected to be approved as a SISO standard in 2013. The approach has generally followed the Lexical Grammar approach introduced by Schade and Hieb [4,5]. Several recent studies and implementations have addressed effective combination of MSDL and C-BML [6-9]. Informing the standardization process have been multiple projects under various US DoD sponsors and an ongoing sequence of experimental BML configurations that were developed and demonstrated by the members of NATO MSG-048 and MSG-085 [10-15]. The experience gained in these activities has proved critical to shaping the MSDL and C-BML standards and implementing infrastructure, such as the translation service described in this paper, and also in demonstrating the potential applicability of BML.

References [13] and [14] describe the C2-Sim environment developed for NATO Technical Activity MSG-048, *Coalition Battle Management Language*. This activity included six national C2 systems and 5 national simulations, a scale of interoperation not previously attempted. Reference [6] describes developmental work for NATO Technical Activity MSG-085, *Standardization for C2-Simulation Interoperation,* leading to an experimental operational environment where multiple national C2 and Simulation systems can interoperate using MSDL and C-BML (see Figure 1).

Continued progress of Coalition Battle Management Language (C-BML) in NATO MSG-085 (Standards for Command and Control-Simulation Interoperation) has resulted in a need to extend the scope possible by doing this with a single server. The extension could be either for purposes of geographic distribution or for extended performance. In preparing for the MSG-085 final demonstration, we have developed an example of such a distributed server, achieved by interoperation between the WISE-SBML server and the FKIE BML server. This paper reports on the benefits of server interoperation, the mechanisms used and issues encountered in implementation, and the resulting capability, which also incorporates the schema translation capability described in [19] and [20]. More details can be found in [19], by the authors of this paper, where much of the information below originally appeared.

The remainder of this paper begins with a description of the NATO MSG-085 plan to use linked heterogeneous servers, followed by descriptions of the two servers involved: the GMU SBMLServer and the FKIE server. This is followed by a description of the methods we used to link the two servers, including the possibility to link three or more servers. The paper ends with a conclusion that projects future use of increasingly server combinations.

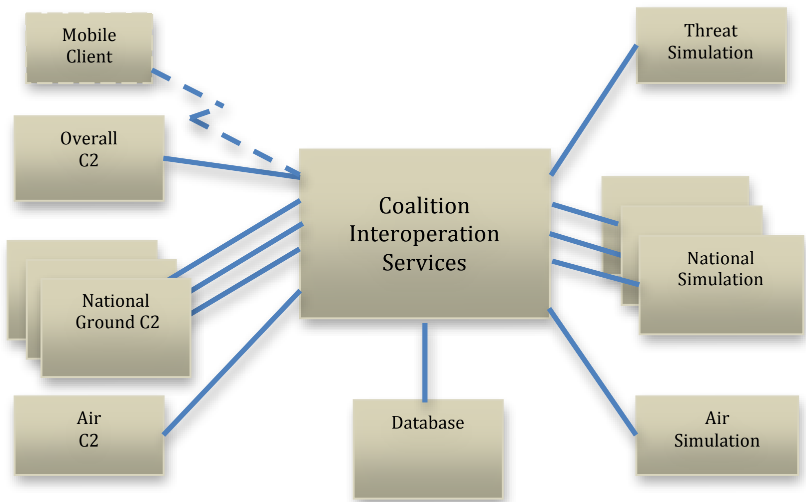

Figure 1: C2-Sim Coalition Services Architecture

2. Distributed Servers for MSG-085

The NATO Modeling and Simulation Group (MSG) charters technical activities conducted by groups from nations that are members or NATO or Partners for Peace to improve understanding and utility of technology involving modeling and simulation. Currently, technical activity MSG-085 is nearing the end of its chartered mission to assess the operational relevance of MSDL and C-BML while enhancing the technical readiness level of their available implementations.

MSG-085 is planning its final demonstration in December 2013, involving eleven C2 and simulation systems operated by eight different nations. The scope of the planned demonstration is such that MSG-085 plans to combine two BML servers: the WISE-SBML server developed by the George Mason University C4I Center and the BML server developed by the Fraunhofer Institute for Communication, Information Processing and Ergonomics (FKIE). The FKIE server will support a coalition of systems previously demonstrated by France and Germany [7], operating under the Integrated BML (IBML) schema as used in 2009 by MSG-048 and enhanced for logistics, which we will refer to here as the CIG Land Ops schema. The WISE-SBML server will support the remaining national C2 and simulation systems, and will translate among four schemata to do so: the two variants defined by the finalized C-BML Phase 1 standard, the original IBML schema, and the CIG Land Ops schema (for details on schema translation see [20]). The general architecture of the linked servers is shown in figure 2.

3. Scripted BML Server

The George Mason University C4I Center, under management of US Army PM OneSAF and in close cooperation with MITRE and QinetiQ personnel, has developed a set of services that provide infrastructure to support implementation of MSDL/C-BML in MSG-085 C2 and simulation systems. The top-level architecture of a C2-simulation coalition using these services is shown in Figure 1. These implementations are available at http://c4i.gmu.edu/OpenBML as open source software.

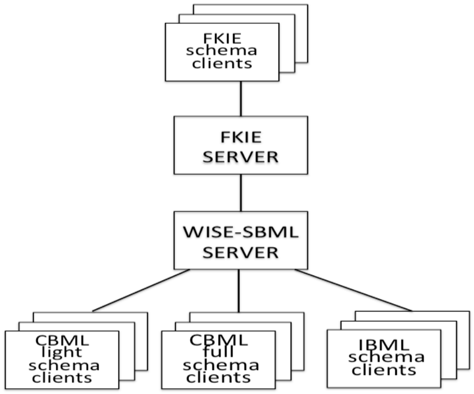

Figure 2: MSG-85 Linked Server Architecture

Experience to date in development of BML indicates that the language will continue to grow and change. This is likely to be true of both the BML itself and of the underlying database representation used to implement the scripted server capability. However, it also has become clear that some aspects of BML middleware are likely to remain the same for a considerable time: namely, the XML input structure and the need for a repository server to store a representation of BML in a well-structured relational database, accessed via the Structured Query Language (SQL). This implies an opportunity for a re-usable system component: a scripted server that can convert between a relational database and XML documents based on a set of mapping files and XML Schema files. The scripted server introduced in [17] and now named "SBMLServer," accepts *push* and *pull* transactions (BML/MSDL XML documents) and processes them according to a script (or mapping file, also written in XML). While the scripted approach may have lower performance when compared to hard-coded implementations, it has several advantages:

- new BML constructs can be implemented and tested rapidly
- changes to the data model that underlies the database can be implemented and tested rapidly
- the ability to change the service rapidly reduces cost and facilitates prototyping
- the script provides a concise definition of BML-to-data model mappings that facilitates review and interchange needed for collaboration and standardization

Since its initial use in NATO MSG-048 [18], SBMLServer has been enhanced considerably by:

- Supporting XML namespaces: XML tags can be qualified by addition of a "namespace" prefix, allowing tags from different sources to work together.
- Schema validation: the server confirms that each document received conforms to the schema, which identifies a likely source of incompatibilities, at the cost of slowing the service
- Filtering data to restrict delivery based on user-defined criteria: SBML supports dynamic definition of Publish/Subscribe Topics
- Logging/replay: the server writes a file showing every transaction it receives, with time stamps. The server is then capable of replaying this file to recreate the original sequence of Orders and Reports at original time intervals.
- Multithreading for performance: server throughput can be improved by processing multiple messages in parallel
- RESTful Web service interface: initially, SBMLServer supported only the traditional Web service protocol SOAP, which is intended to support remote procedure calls. However, the need in BML is for transfer of documents, which can be achieved more efficiently via Representational State Transfer (REST) protocol, reducing overhead and facilitating C++ implementation. MSG-085 has adopted the RESTful approach.
- Aggregating MSDL inputs.
- Schema translation: because SBMLServer parses the BML input documents and stores their XML elements in a database, it is possible to generate a version of the document translated to comply with a different, semantically similar schema.

Reference [19] describes a new generation server, WISE-SBML, developed on the experience base gained in building SBMLServer.

4. FKIE Server

The FKIE Server was developed as an alternative to the SBMLServer with the focus on allowing a simple, fast exchange of BML while keeping flexibility to change the schema. Incoming messages are distributed over the messaging service and are stored in the file system for later requests. The XML of the messages is neither changed nor validated by the server. However, for some messages types the server does a search for predefined strings to extract order, report or request IDs. This makes the server fast and allows changing the schema from one experiment to another without also changing the server. This allowed fast development of new schemata and schema extensions.

Currently, the FKIE Server supports IBML, SISO Phase 1 and the CIG Land Ops schema [7]. Since there is no converting done by the server, all clients must agree upon one of the mentioned schemata at the start of a session or experiment. In addition to BML message exchange, the FKIE server supports MSDL for scenario initialization. Messaging services JMS and STOMP currently are supported, while SOAP and a RESTful interface are offered for Web services.

5. Linking Two Servers

We found a straightforward way to link two servers. Since each server implements the publish/subscribe protocol STOMP for its outputs to clients, the outputs can be captured by the other server, listening to the STOMP publication. Each server therefore implements a back-to-back (B2B) client, which contains both a STOMP subscriber and a REST output, and shown in figure 3. The B2B client starts after both clients are running, subscribes to the source server, and forwards all received BML documents to the sink server, subject to a filter.

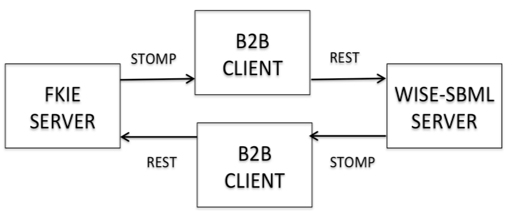

Figure 3: Server Linking Architecture

One potential problem was detected when using this arrangement: a BML document could circulate between the two servers indefinitely unless some mechanism is provided to break the loop. We achieved breaking the loop by adding a parameter to the REST and STOMP interfaces, intended for use by the B2B clients but not by regular C2-Sim clients. This parameter *firstforwarder* is set by the first server to publish each document, to identify itself by a unique code (we use the server's IP address for this). The B2B client checks this code for each received document and does not forward any document to the server that was the first forwarder.

We have considered the question of whether the architecture shown in figure 3 is scalable to multiple servers. The answer appears to be "no", because a loop can exist that does not include the first forwarder. In the general case, it would be necessary to devise a mechanism that can break every possible loop. We will leave the creation of this mechanism to a future paper.

6. Conclusions

Work in C2-simulation interoperation, using emerging SISO standards, continues to make progress as described in this and companion papers. Practical implementation by MSG-085 team members is leading to understanding of how military operations can be supported effectively by this technology. National implementations in both C2 and simulation systems, coupled with supporting open source server software, make the feasibility of this approach clear. This work has two commendable results: the interoperating systems will support operational experimentation now being planned by MSG-085, and also will continue to provide the experience needed for SISO MSDL and C-BML product development groups to produce effective standards, based on technical approaches that have been demonstrated to be effective.

The linked-server architecture described in this paper is only the first step toward larger BML systems, based on distributed servers. We anticipate that the future where coalition partners routinely will incorporate simulation into their C2 systems, with each coalition partner providing a simulation that represents their own forces. Such a future could require complex architectures consisting of C2 and simulation systems, all of which are clients of a distributed system of servers. There is considerable room for research on these topics.

References

[1] Sudnikovich, W., J. Pullen, M. Kleiner, and S. Carey, "Extensible Battle Management Language as a Transformation Enabler," in *SIMULATION,* 80:669-680, 2004

[2] Simulation Interoperability Standards Organization, *Standard for: Military Scenario Definition Language (MSDL)* http://www.sisostds.org/DigitalLibrary.aspx?Command=Core_Download&EntryId=30830

[3] Blais, C., K. Galvin and M. Hieb, "Coalition Battle Management Language (C-BML) Study Group Report," IEEE Fall Simulation Interoperability Workshop, Orlando FL, 2005

[4] Schade, U. and Hieb, M., "Formalizing Battle Management Language: A Grammar for Specifying Orders," *2006 Spring Simulation Interoperability Workshop*, IEEE Spring Simulation Interoperability Workshop, Huntsville AL, 2006

[5] Hieb, M. and U. Schade, "Formalizing Command Intent through Development of a Command and Control Grammar," 12th International Command and Control Research and Technology Symposium, Newport, RI, 2007

[6] Pullen, J., D. Corner, R. Wittman, A. Brook, O. Mevassvik, and A. Alstad, "Technical and Operational Issues in Combining MSDL and C-BML Standards for C2-Simulation Interoperation in MSG-085," NATO Modeling and Simulation Symposium, Stockholm, Sweden, October 2012

[7] Remmersmann, T., U. Schade, L. Khimeche, and B. Gautreau, "Lessons Recognized: How to Combine

BML and MSDL," IEEE Spring Simulation Interoperability Workshop, Orlando, FL, 2012
[8] Heffner, K. C. Blais and K. Gupton, "Strategies for Alignment and Convergence of C-BML and MSDL," IEEE Fall 2012 Simulation Interoperability Workshop, Orlando, FY, 2012
[9] Pullen, J., D. Corner and R. Wittman, "Next Steps in MSDL and C-BML Alignment for Convergence, IEEE Spring 2013 Simulation Interoperability Workshop, San Diego, CA, 2013
[10] Sudnikovich, W., A. Ritchie, P. de Champs, M. Hieb, and J. Pullen, "NATO Exploratory Team – 016 Integration Lessons Learned for C2IEDM and C-BML," IEEE Spring Simulation Interoperability Workshop, San Diego CA, 2006
[11] Galvin, K., W. Sudnikovich, P. deChamps, M. Hieb, J. Pullen, and L. Khimeche, "Delivering C2 to M&S Interoperability for NATO - Demonstrating Coalition Battle Management Language (C-BML) and the Way Ahead," IEEE Fall Simulation Interoperability Workshop, September 2006
[12] Pullen, J., M. Hieb, S. Levine, A. Tolk, and C. Blais, "Joint Battle Management Language (JBML) - US Contribution to the C-BML PDG and NATO MSG-048 TA," IEEE European Simulation Interoperability Workshop, June 2007
[13] de Reus, N., R. de Krom, O. Mevassvik, A. Alstad, U. Schade and M. Frey, "BML-enabling national C2 systems for coupling to Simulation,", IEEE Spring Simulation Interoperability Workshop, Newport, RI, 2008
[14] Gustavsson, P., M.R. Hieb, M. Groenkvist, V. Kamath, Jakob Blomberg, and Joakim Wemmergard. "BLACK-CACTUS – Towards an Agile Joint/Coalition Embedded C2 Training Environment," IEEE Spring Simulation Interoperability Workshop, Providence, RI, 2008
[15] Levine, S., L. Topor, T. Troccola, and J. Pullen, "A Practical Example of the Integration of Simulations, Battle Command, and Modern Technology," IEEE European Simulation Interoperability Workshop, Istanbul, Turkey, 2009
[16] Pullen, J., D. Corner, P. Gustavsson, and M. Grönkvist, "Incorporating C2--‐Simulation Interoperability Services into an Operational C2 System," International Command and Control Research and Technology Symposium 2013, Alexandria, VA
[17] Pullen, J., D. Corner and S. Singapogu, "Scripted Battle Management Language Web Service Version 1.0: Operation and Mapping Description Language," IEEE Spring 2009 Simulation Interoperability Workshop, San Diego CA, 2009
[18] Pullen, J. and K. Heffner, "Supporting Coalition Battle Management Language Experiments with Scripted Web Services," NATO 2009 Modeling and Simulation Symposium, Brussels, Belgium, 2009
[19] Pullen, J., D. Corner, R. Wittman, A. Brook, P. Gustavsson, U. Schade and T. Remmersmann, "Multi-Schema and Multi-Server Advances for C2-Simulation Interoperation in MSG-085," NATO Modeling and Simulation Symposium 2013 (submitted)
[20] Pullen, J., D. Corner, P. Gustavsson, and R. Wittman, "Order and Report Schema Translation in WISE-SBML Server," IEEE Fall 2013 Simulation Interoperability Workshop, Orlando, FL, 2013

Author Biographies

DR. J. MARK PULLEN is Professor of Computer Science at George Mason University, where he serves as Director of the C4I Center and also heads the Center's Networking and Simulation Laboratory. He is a Fellow of the IEEE, Fellow of the ACM and licensed Professional Engineer. He has served as Principal Investigator of the XBML, and JBML, IBML and SBML projects.

DOUGLAS CORNER is a member of the staff of the George Mason University C4I Center. He is the lead software developer on the SBML scripting engine.

THOMAS REMMERSMANN M.Sc. is a Scientific Employee and Software Engineer with Fraunhofer FKIE in Wachtberg, Germany. He has several years of experience with building and evaluating C2 interfaces to interact with simulation systems and robotic forces, especially with the use of BML.

DR. IRMTRUD TRAUTWEIN is Scientific Employee and Software Engineer with Fraunhofer FKIE in Wachtberg, Germany. She has several years of experience with building and evaluating graphical user interfaces for C2 systems and other military applications.