OBJECT CORRELATION FOR SIMULATION AND COMMAND AND CONTROL SYSTEMS

Stephen R. Kasputis
Techmatics, Inc.
Arlington, Virginia 22202

CDR Peggy A. Feldmann
Defense Advanced Research Projects Agency
Arlington, Virginia 22203

Abstract
The need and a possible approach for a general object correlation scheme for use in the Department of Defense Command and Control and Modeling and Simulation systems is discussed. The definition of the Defense Information Infrastructure and adoption of the High Level Architecture require information exchange in the form of software objects. The information required by and transmitted from different levels of command differs in the level of detail. Similarly, simulations performed at different levels of abstraction require descriptions of the battle space at different level of detail. In such object based systems, if unambiguous communication or interoperability of simulations at various levels of abstraction are to become a reality, a scheme is required which will ensure the consistent and accurate mapping of objects at one level of detail with objects at the next. The alternative to this scheme is the requirement that all systems possess the ability to properly receive and interpret descriptions of the battle space at all levels of detail. The computational overhead and communications bandwidth required under such circumstances may seriously degrade performance and compromise functional requirements. These issues will be discussed in detail and the functional requirements for an object correlation scheme will be given. The framework for such a correlation scheme and some correlation approaches will be presented. Advantages of testing such a scheme in the realm of modeling and simulation prior to applying it to command and control systems will be given.

Author Biographies
Mr. Kasputis is the Technical Director of the Undersea Warfare Directorate of Techmatics, Inc. He has 12 years experience in the establishment and management of research and development programs. His previous positions include Director, Science and Technology, Signal Corporation, Technical Director of the Navy's Fixed Distributed System, Deputy Program Manager of the Navy's second generation signal processor, and the Advanced Technology Officer for Undersea Surveillance Systems.

CDR Feldmann is the DARPA Program Manager of Synthetic Forces for the Synthetic Theater of War Advanced Concept Technology Demonstration-STOW '97. After receiving her commission via the US Naval Academy in May 1980, ENS Feldmann reported to the Naval Oceanographic Research and Development Activity where she helped define acoustic methodologies for long range detection and tracking of submarines. In 1986, LT Feldmann received her masters degree in Acoustical Engineering from the Naval Postgraduate School. After completing Diving and Salvage school, she reported to the Naval Ocean Systems Center in San Diego, CA. where she was the Test Director for the Advanced Research Projects Agency (ARPA) sponsored Project Ariadne. From 1988-1990, LT Feldmann was Assistant Officer in Charge of a team of Naval personnel acting as liaison between the Space and Naval Warfare Systems Command (SPAWAR) and the Fleet. In this capacity, she ran numerous tests determining Operator Machine Interfaces, maintainability, survivability, and applicability for subsystems in the Undersea Surveillance System. At SPAWAR from 1990-1994, LCDR Feldmann was the project manager of the surveillance community's environmental performance prediction system and Test Director of a deployable version of the Fixed Distributed System.

OBJECT CORRELATION FOR SIMULATION AND COMMAND AND CONTROL SYSTEMS

Stephen R. Kasputis
Techmatics, Inc.
Arlington, Virginia 22202

CDR Peggy A. Feldmann
Defense Advanced Research Projects Agency
Arlington, Virginia 22203

I. STATEMENT OF NEED

For the battlefield of the future, communication schemes will be part of the Defense Information Infrastructure (DII) and so must comply with its architecture. Central to this architecture is the passing of information in the form of software objects and the use of object brokers. Effective operation of multiple systems within this architecture requires that these systems share a common and unambiguous interpretation of each object. Such a shared interpretation is unlikely for communication systems independently developed by different services, different contractors, and for different purposes. Indeed, current systems that play in the DII have already shown interpretations of objects that are not consistent between systems. That is, for example, an object named “arrow” may be used by two different applications but have vastly different meanings in each. If one publishes its arrow object, the other may read it, recognize that it does not contain the attributes it expects and at best ignore it or at worst publish an error message that either halts or crashes the system.

Compounding the situation is the need for different levels of detail at different levels of command. A commander at any level requires information about the battlespace within his unit’s sphere of influence, including knowledge of enemy units whose sphere of influence includes his unit. The platoon commander’s sphere of influence is not extensive, but he requires very precise information. The division commander’s sphere of influence is more expansive, but receipt of information at the level of detail valuable to the platoon commander would generally result in information overload for him and his staff. Consider the example of a data object called “enemy position.” The platoon commander would typically like information within a few meters' accuracy on entities like individual combatants and fighting vehicles. Useful position information for the division commander would typically be on aggregate units such as companies or battalions with accuracy measured in hundreds of meters to kilometers.

An analogous situation occurs in the realm of simulations which must conform to the object-based High Level Architecture (HLA). Consider simulation in support of combat system acquisition. For early concept exploration, system representations need to be of the top level performance parameters. Additional detail or resolution during this phase is typically neither desired nor possible. Once a concept is selected and preliminary design begins, aspects of the performance parameters can be allocated to the system components, and their performance can begin to be modeled. Resolution is thus increased as component level modeling begins. Increased level of resolution can continue to be added as the design matures and becomes more detailed. High resolution simulations which can support final detailed design, the manufacturing process, and test and evaluation are the culmination of this ever increasing simulation resolution.

To fully realize the potential support simulation can provide to acquisition, coherent and consistent mapping of system performance between these various levels of resolution must be established and maintained. Proper evaluation of a system’s contribution to its designated mission areas must be done in the context of the entire battlespace. Simulation of the entire battlespace at the highest level of detail is impractical from the perspective of both computational resources and model availability. Therefore, to fully and accurately evaluate the impact of proposed detailed design tradeoffs requires that the critical aspects of the low resolution battlespace be properly portrayed to the higher resolution simulation. Equally important is that the results of the high resolution simulation be accurately reflected in the battlespace simulations of lower resolution.

II. INTEROPERABILITY OPTIONS

Five general approaches are possible to address interoperability of C4I systems or simulation federations which may be working at differing levels of detail. These approaches and their more obvious strengths and weaknesses are:

a. No established structure or definition of the general object space. This would require modification of each system to minimize the impact of receipt of an object not conforming to expected content of an object of that name. The one strength of this approach is that it is low risk and technically simple to enact. The first obvious weakness of this approach is the high likelihood of critical information being ignored by the receiving system. This is obviously unacceptable and other weaknesses such as increased software error-checking overhead and excessive bandwidth requirements in a multi-cast environment become moot.

b. Bilateral agreements on the definition of the object space between every pair of C4I or simulation systems that have the need to interoperate. Again, this is a low risk, low technology approach. The primary weaknesses of this approach include the fact that it would be very labor intensive. The object space of every application would have to be altered once for every other application with which it would need to interoperate. This means several alteration efforts for each application. Additionally, several applications would have to be altered every time a new application is introduced. In the simulation realm, this approach is equivalent to the need to form a new federation with all its overhead startup costs between every pair of simulations which might need to interact. This approach also results in a very poorly coordinated and difficult to maintain total object space. The subset of the total object space of which each application would need to be aware would also need to be larger than optimum. Configuration management issues of object definitions would eventually force the evolution of this approach to:

c. Universal agreement on the definition of the total object space. Every C4I or simulation system would have to recognize and know how to deal with every object in this space. In the vocabulary of the Defense Modeling and Simulation Office's High Level Architecture, every simulation would adhere to the same Federation Object Model, which would include the entire object space. This approach again has the strength of low technical risk. A weakness is the burden placed on object brokers by the excessively large object set it must manage. This burden will certainly affect communications efficiency and may result in intolerable delays in object routing. Execution of this approach would require strict configuration management of the object space. Considering the extent of the space and number of parties dependent upon it, such strict management is probably not executable in practice.

d. Specific translators designed to interpret objects between one application and another. This approach is again rather low risk and low technology and avoids many of the undesirable object space definition issues of the previous approaches. This approach too, however, is labor intensive in that the number of translators could grow rapidly with the number of applications which comprise the system. The number of translators required for each application can also result in a large translator application with significant hardware requirements and a second layer of overhead needed for routing of incoming objects to the proper translator.

e. Segmentation of the total object space through levels of detail and development of a generic object mapping scheme to allow for interoperability of systems operating at all levels of command through all phases of mission planning and execution. The strengths of this approach are minimal impact to application, ease of addition or upgrade of applications, and a structured, well-defined, verified, and easily maintained object space. The primary weakness of this system is the need for significant supporting knowledge acquisition and research and development efforts. The need for object space standardization and configuration management is again a serious consideration. This issue is mitigated to some extent since standardization is required only within subsets of the entire object space.

Despite the upfront effort and need for some amount of disdained standardization, this final option presents the best potential for acceptable performance and affordable life cycle costs for all associated systems. It is, therefore, this option upon which further discussion will focus.

III. ARCHITECTURE AND REQUIREMENTS FOR RECOMMENDED APPROACH

The proposed architecture for the recommended approach is shown in Figure 1. Any particular system at a specific location is seen to be operating at a definite level within the mission space. It has all necessary knowledge of the objects that are exchanged between systems at that mission space level. It need not be able to interpret the objects that are used at any other level above or below the level at which it is operating. Communications between levels of the mission space are through an object resolution mapping mechanism.

All applications that must deal with a particular level of the mission space, regardless of functional area, must be considered when defining the mission space. For example, if one simulation system has historically been used to support acquisition and another to support training, but they both represent systems at similar levels of abstraction, they would both be considered when determining the object set for that level of abstraction. This architecture thus replaces traditional stovepipes based on application with pie plates based upon level of detail. Unlike stovepipes which have little or no intra-communications, the basic tenant of this architecture is providing the means for systems on different pie plates to freely exchange information. Interactions between systems on different levels of the mission space occur through a resolution mapping mechanism or "resolution bridge."

It is thus easy to see that there are two features which are central to this architecture. First is the establishment of standardized mission space levels; the Conceptual Model of the Mission Space (CMMS) for simulation systems or its equivalent for C4I systems, a Mission Space Definition (MSD). Second is an object mapping mechanism to allow for communication between different mission space levels. C4I or simulation systems operate within one of the standardized mission levels. The object space these systems must use is derived from and consistent with those standardized levels. The mapping mechanisms will unite the object spaces of each of level to allow for complete representation of the battlespace.

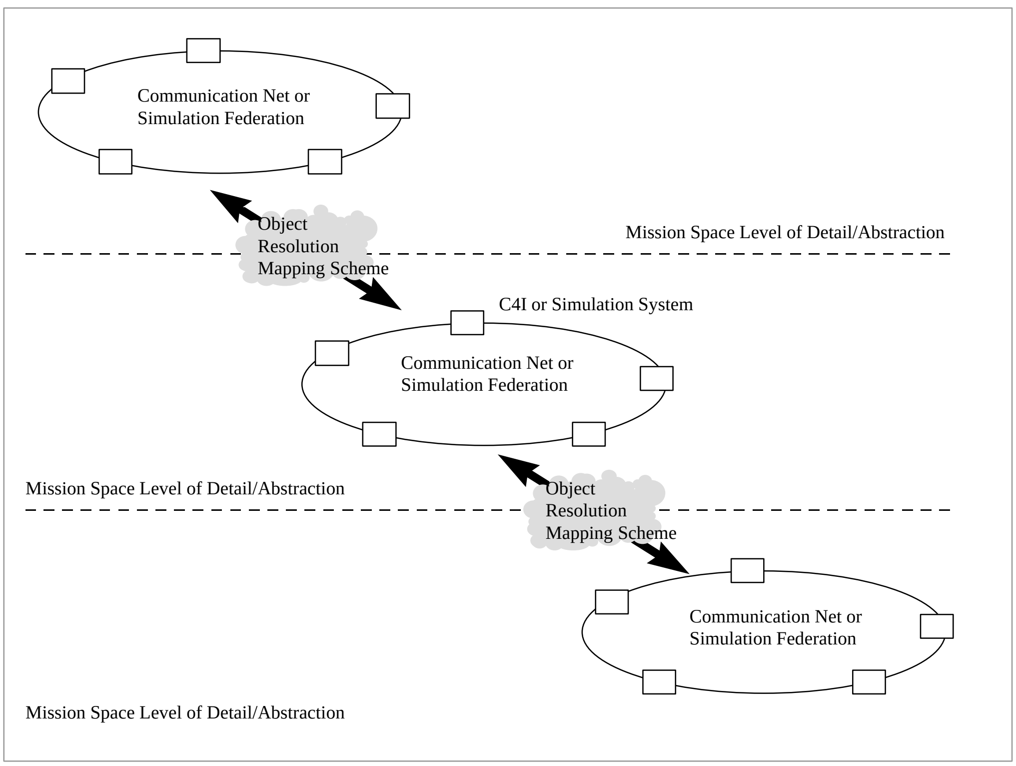

Figure 1. Architectural Concept

IV. STANDARDIZED MISSION SPACE

Full delineation of the MSD is neither required nor a realistic expectation to allow for initial implementation of the architecture. Indeed, it should be expected that the MSD will grow and change as the architecture and applications mature and the roles and missions of the military change. One of the strengths of the proposed architecture is its ability to economically accommodate change. What is required to begin implementation of this architecture is definition of the standardized levels for the MSD. This is analogous to defining the levels of abstraction in the current CMMS efforts.

Once the standard mission levels are defined, the object space for each of those levels can begin to be established. As already implied, the object set that represents each level of the mission space must be comprehensive for that level. While applications need only to be able to discern that subset of the object space they require for complete functionality, the object set that represents any level of the mission space must include all objects that the collected systems of that level may require as input or output. For simulation systems, this is equivalent to defining a master Federated Object Model (FOM) for each level of abstraction. Since it is possible that systems operating at different levels may require some pieces of identical information, it should be noted that the definition of identical objects at different levels of the mission space is both possible and allowed.

Defining the standard levels of the mission space and initial establishment of associated object spaces is a non-trivial undertaking. A considerable amount of knowledge acquisition and systems engineering is required to make intelligent choices in these areas. Resources applied early in the effort to ensure optimal design will, however, minimize life cycle costs for maintaining system interoperability by minimizing the number of significant changes to the object spaces.

Also critical to the long-term compatibility of systems is the effective configuration management of the object spaces. At each level of the mission space, configuration management of the object space realization of that level must be enforced. The configuration management of each level of the mission space could conceivably, however, be maintained independently from the others.

Ideally, the CMMS and MSD and their object space realizations should be nearly identical, since the simulation should reflect reality to the greatest extent possible. If practical considerations initially preclude this, it should certainly be a goal for the two to merge as the system architecture matures and simulations improve and become an embedded part of C4I systems.

V. OBJECT RESOLUTION MAPPING

The mapping of objects between levels of the mission space is primarily a mapping of objects of one level of resolution or detail to objects of another level of detail. This mapping is not the same as the aggregation/deaggregation issue of concern to the modeling and simulation community. That is an issue within a simulation. In the context of modeling and simulation, the resolution mapping discussed here is an issue of communication between simulations. Some top level functional requirements for the schema of information passing and resolution mapping between the various levels of the mission space object can be easily identified.

First, the mapping schema must abide to the interface specifications of the networks or federations with which it communicates. That is, from the perspective of the applications, the resolution bridge is simply another application. It must abide by the communication protocols of the networks. It must also be capable of properly using the routing space schema and algorithms of each network. Obviously, it must also have knowledge of the full object description of each level of the mission space with which it is to communicate. Maintenance of the resolution bridges must, therefore, be closely tied to the configuration management process of the mission space objects.

Another functional requirement for the object mapping schema is that it be fast enough so as not to impose excessive or unacceptable overhead on the systems between which it is adjudicating. Some overhead and time must be expected and accounted for in the engineering of the individual systems and networks or federations. However, the overhead allocated to the mapping schema should ideally be small compared to the object manipulation and use time of the applications.

The most obtuse functional requirement of any schema for the mapping of objects at one level of resolution to those at a different level is the preservation of an accurate presentation of the situation. Reality must not be changed by the mapping process. Representations of the situation must be consistent between the mission space levels even though resolution differs.

To clarify this last point, consider the following example. A simulation at the platform entity level of resolution may represent the performance of a passive sonar system against a submarine without need for specific acoustic source levels. One way this could be done is through a look-up table of typical performance. Acoustic source information would, therefore, be unlikely to exist at this level of abstraction. At the more detailed level of simulation, the representation of the sonar system performance requires specific source level information about the submarine. The mapping scheme that translates the existence of the submarine from the lower resolution simulation to the higher resolution simulation must, therefore, supply this information. The source levels provided, however, must be appropriate for the type of submarine and its operational state. Providing source levels which are twice the realistic level would, for example, result in a representation of the sonar's performance which would be far more optimistic than and grossly inconsistent with the representation of the performance of the sonar system at the lower level resolution.

The resolution bridge must also be capable of declaring object interest in both of the object spaces between which it arbitrates. Minimum functionality in this area would entail capturing and translating all objects which could be mapped into the other mission space layer. This could result in substantial excess overhead if a large subset of the mappings are not required for a particular communications or simulation scheme. It is thus also desirable that the resolution bridge have the capability to easily tailor the object set in which it declares interest. This tailoring could conceivably be either done once at system initiation or dynamically during system utilization.

Closely related to the idea of a tailoring of object interest declaration is an optional but desired function that the resolution bridge have specific knowledge of the applications at both levels of resolution so the mapping can be tailored for each specific transaction. Such knowledge could result in faster mapping and decreased bandwidth requirements. Consider another illustrative example. Suppose an object in one mission space level is mapped to 10 objects at the level of next higher resolution. If the specific application at the higher resolution level in need of the information only uses four of these objects and it is, at the time, the only application with an interest in the object of lower resolution, the mapping schema need only produce those four objects. The time and processor cycles required to produce the other six objects could be saved. Additionally, the communications pathways need only transport four objects between the resolution bridge and the application. The bandwidth to support the transport of all ten objects is not required.

For simulation systems, one additional functional requirement should be considered. Simulations can run at scaled time or be event based. If simulations with different execution timing scales or bases are to be joined, the resolution bridge must ensure that synchronization is maintained. This synchronization is in addition to resolution mapping and techniques which may be used for this are not addressed in this paper.

VI. RESOLUTION MAPPING TECHNIQUES

Stratification of the mission space as proposed is only a viable option if resolution mapping can indeed occur. This leads to the question of how such mapping could be accomplished. Below are some basic techniques that can be applied to the mapping of objects across levels of resolution.

Unity mapping. The simplest of all mappings occurs when the object definition is identical in both levels of the mission space. Object mapping in this instance is a trivial pass through or exact one-to-one mapping.

Class/object/attribute hierarchy transition. Since information will be passed in the form of objects, a simple hierarchical transition of classes to objects and objects to attributes may be possible in mapping to levels of lesser detail. Translations in the other direction could allow objects to classes and attributes to objects, but determination of the higher detail attributes would require one of the other methods below. It may also occur that only a subset of the attributes needed at the higher level of resolution are required by a system at a lower level of detail. In this case, a simple filtering of attributes may provide an adequate mapping.

Models and algorithms. Models and algorithms which describe the detailed aspects of an entity or situation can be used to provide additional detail when required. For example, if at one level of resolution the effects of an artillery piece is modeled and at the next level the flight of the projectiles of that artillery piece is required, it would be possible to model the flight path given the characteristics of the artillery.

Tables. Similar to the use of models and algorithms, this method may indeed be considered an inelegant subset of that technique. An example here might be that aspect dependent radar cross section is maintained in a table that the higher resolution system accesses using state information from the lower level system to determine which entry to retrieve.

Initialization or Instantiation. A new object may be instantiated if required. Again using a simulation example, suppose that a missile is represented at one level of resolution as a single entity. At the next higher level of detail, it may be necessary to represent the missile as several constituent parts. Many or all of these parts, objects in higher detail representation, may not be attributes in the original representation. New objects will have to be instantiated and initialized based upon the state data of the lower resolution representation of the missile.

VII. TESTING THE ARCHITECTURE

Proof of principle testing is required to demonstrate the viability of the architecture and its key components. The issues associated with the definition of the mission space levels are essentially of politics and information gathering and standard engineering practices. The major technology issues reside with the resolution bridge. The technology to handle this in the general sense is far from a production state and presents a significant research and development effort to enable the implementation of the proposed architecture. Much of the technology effort, however, lies in the integration and generalization of the building block techniques discussed above. Since several of these basic building blocks already exist, prototypes of resolution bridge can be produced. Such prototypes can be used to refine the requirements of the object mapping schema and define the development areas to be pursued.

Development and testing of the prototype components and architecture can easily be accomplished in the realm of modeling and simulation. Demonstrating the interoperability of two simulation federations which model at different levels of resolution will prove that the most technically challenging aspects of the proposed architecture are achievable. Using simulations which currently exist or are in development avoids the most controversial aspects of mission space level definition. There are other advantages with the use of simulation systems over communications systems. There is a richer set of maturing object based simulations from which to choose. Scenarios and interactions in simulations can be carefully controlled and ground truth precisely determined and recorded. This is an important aspect when attempting to validate the consistency of the mapping schemes. Simulations can be run at development activities with little operational impact or need of support from the warfighters or their equipment.

VIII. CONCLUSIONS

An architecture is required for the organization and management of the objects which will describe the mission space under the DII and HLA. Such an architecture is proposed. Strengths and weaknesses of alternatives were presented. Functional requirements for the key infrastructure piece of this architecture were discussed and techniques for its implementation presented. Reasons for developing and testing the architecture in a simulation environment were given.