

# I<sup>N</sup>: THE 'N' DIMENSIONS OF INTEROPERABILITY

Roger D. Smith  
Mystech Associates  
Orlando, Florida

## ABSTRACT

As simulations have evolved over the last two decades, interoperability between them has emerged as a fundamental technique for increasing their applicability and minimizing the cost of developing and maintaining them. From totally independent systems, to manual interfaces, automated interfaces, messaging standards, control standards, and architectural standards, the field has been transformed from a set of independent programs to a loose confederation working together to maximize each others investments. In the future, simulation value will be measured by the degree of interoperability that can be attained.

Many efforts to realize broad interoperability have been pursued, each with an increasing degree of ambitiousness. We are now envisioning systems and architectures that can support the connection of all types of simulators and are reaching into the broader realm of C4I systems. This paper will define many of the dimensions of interoperability as they exist today, as are planned for the near future, and as could ultimately be achieved. Twelve dimensions will be enumerated and an algebra defined to aid in describing the relationships and the implications of extending interoperability into the different combinations of these dimensions. This algebra is motivated from the DEVS Formalism originally developed in the book *Theory of Modelling and Simulation*. Graphical portrayals will also be explored as tools to aid in communicating the concepts. The paper champions a structured approach that lays a foundation to support future growth, realizing that the capabilities envisioned today do not encompass the dreams of tomorrow.

## BIOGRAPHY

Roger Smith is the Chief Simulation Scientist and a Corporate Technical Director for Mystech Associates. He is actively involved in designing, developing, and fielding simulations, having contributed to JSIMS, WARSIM, BLRSI, ADST, TACSIM, AWSIM, ALBAM, and others. He is very active in the simulation industry serving as the Vice Chairman for the ACM Special Interest Group on Simulation, the General Chairman for the Electronic Conference on Training Simulation (ELECSIM) series, and as a member of the editorial boards of *ACM Transactions on Modeling and Computer Simulation* and the *International Journal of Computer Simulation, Modeling, and Analysis*. His research interests include embedded simulations in operational equipment, integrated simulation-C4I systems, and problems in aggregation/disaggregation.

# I<sup>N</sup>: THE 'N' DIMENSIONS OF INTEROPERABILITY

Roger D. Smith  
Mystech Associates  
Orlando, Florida

## BACKGROUND

Around 411 BC, Plato wrote "The Republic" and in one dialogue challenged his Greek pupils to imagine a place where people lived in a cave and were chained to the floor such that they could see only the wall of the cave and not the opening. Their entire perception of the world were the shadows of objects and activities that took place outside and were cast on the wall of the cave in front of them. In this world the shadows would be, for the chained audience, their perception of the real world. They would believe that there was no more to the world than what they were seeing. He illustrates the point that reality is defined by what we are able to perceive and our world expands as our perception expands.

In 1884, Edwin Abbott wrote a small book *Flatland: A Romance in Many Dimensions*. He portrayed a world of only two dimensions and described the creatures living there, their environment, and the methods of conducting their lives. These creatures were entirely ignorant of the third dimension and interactions with a three dimensional creature appeared magical to them.

In both cases, the characters in the stories are robbed of the perception of one of our fundamental dimensions of reality. These illustrate how a world can be real with fewer dimensions than we deem natural - the simulation community is in a similar situation. We have spent several decades chained in a cave of few dimensions and now find ourselves set free to explore the many dimensions of the shared synthetic world. This paper attempts to illustrate what some of these dimensions are and how they are related.

## Simulation Evolution

Though simulations have been used by the military for analysis and training for centuries, it was not until the advent of computer networks that the issue of integrating multiple heterogeneous systems emerged. Prior to that, it was naturally assumed that each simulation was a virtual world unto itself, and that information exchange was a laborious, human-

driven process. Analytical simulations were the first to crack the "interoperability barrier" by creating simulations that were specifically designed to accept the output of another model as the input for beginning a different stage of analysis.

The virtual community created the Simulator Networking (SIMNET) systems with their associated protocols for joining multiple copies of the same simulator into a common synthetic world. Though these were intended for homogeneous simulations at a single location, they pointed the way for more ambitious projects. Distributed Interactive Simulation (DIS) protocols followed, adding the ability to join a large variety of simulations and to do so across wide area networks.

As this was developing the constructive community was creating the Generic Data System (GDS) to bridge the gaps between constructive simulations. These allowed widely different simulations to exchange world data and deposit it in a common data repository and analysis system. This was followed by the Aggregate Level Simulation Protocol (ALSP) which created a more dynamic environment and provided distributed management functions.

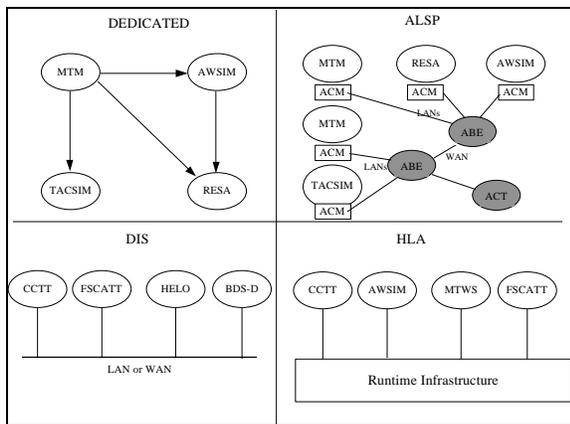
The High Level Architecture (HLA) promises to join both of these communities and allow them to work together at a level not attainable before. Simulations from many fields: staff training, crew training, live training, analysis, engineering, and testing will be able to take advantage of a single architecture for distributed simulation.

The next step in this evolution is the integration of real world C4I systems into the synthetic world. Projects like the Modular Reconfigurable C4I Interface (MRCI), Tactical Simulation System (TACSIM), and other experiments are illustrating the fact that these command systems can be extensions of the distributed simulation paradigm. TACSIM supports dedicated interfaces between simulations and C4I systems and MRCI promises to extend this to the same generic level available to simulations through the HLA Run Time Infrastructure (RTI). With this continuous

expansion of the dimensions of interoperability it is probable that the integration of other forms of combat computer systems will follow: radar trackers, radios, communication stations, operational sensor systems, and even combat aircraft. As real systems, particularly C4I systems, settle into the use of standard message formats and data exchange methods, the level of interoperability between these different fields will continue to increase.

The era of dedicated, stand alone simulation models is coming to a close. The class of problems that can be addressed with today’s technology far exceeds the ability of a single model to capitalize upon them. Distributed, cooperative simulation solutions that can still operate independently are going to be the systems of the foreseeable future.

A topological illustration of each of the levels of interoperability is shown in Figure 1.



**Figure 1. Simulation Interoperability Techniques**

**Interoperability Definitions**

As interoperability evolves we find ourselves with several definitions of what it is. These vary according to their intended audience and the point-of-view of the author, but three prominent and compatible definitions are:

“Two or more simulations/simulators are interoperable for a given exercise when their performance characteristics support a fair fight to the fidelity required by the exercise.”

COL James Shifflet, US Army STRICOM

“The ability of a [...] simulation to provide services to, and accept services from, other [...] simulations, and to use the services so exchanged to enable them to operate effectively together.”

HLA Architecture Management Group

“A domain-specific software architecture is, in effect, a multiple-point solution to a set of application-specific requirements (which define a problem domain).”

Will Tracz, DSSA Principal

These represent the operational, network, and architecture points-of-view of the same problem. Each sees different aspects of the problem as primary, and each applies a unique solution to the problem. But they are all attempting to achieve similar results.

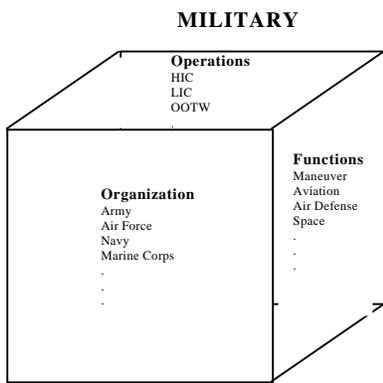
**DIMENSIONS OF INTEROPERABILITY**

We will attempt to enumerate and describe the various dimensions of interoperability that exist or are appearing on the horizon in requirements for new simulation systems. It is clear that the emergence of new technologies results in the creation of new dimensions in interoperability. Each capability brings with it the possibility of extending the existing simulation in new directions, so we do not mean to imply that this list of dimensions is comprehensive. Some of the dimensions are often implemented together in the application realm so that they appear as two or more unique dimensions only during analysis. The twelve dimensions described below may fold together to form fewer dimensions with richer characteristics when actually implemented in a simulation, but each of them does have unique properties which justify its independent identity here.

**Expanding Heterogeneity**

When a simulation is built to solve a single, self-contained, bounded problem it is difficult to conceive of dimensions of interoperability as a significant problem. But, as simulations are connected to others of very diverse natures and the whole applied to different problems in each domain, the interoperability problems multiply. Current desires to join all forms of simulations and military computers create the high dimension environment we are working to capture and describe.

**Military Dimensions.** Within the military domain, the first class of dimensions are naturally those of Military activities (Figure 2). The services have spent the last two centuries defining and separating themselves from each other, creating their own unique identity. Simulation interoperability is now asking them to identify where they are the same so that they can operate together. We must deal with the differences that have been erected over two centuries between Military Organizations: Army, Air Force, Navy, Marine Corps, Coast Guard, Special Forces, Intelligence Organizations, the Department of Justice, and the Drug Enforcement Agency.



**Figure 2. Military Class Cube**

Strictly speaking each of these is not a part of the DOD. However, in the modern world they are called upon to interact and support one another to implement the domestic and foreign policies of the nation to which they belong. So it may be just as important for the DEA to exchange information with the Defense Intelligence Agency as it is for the Army or Navy.

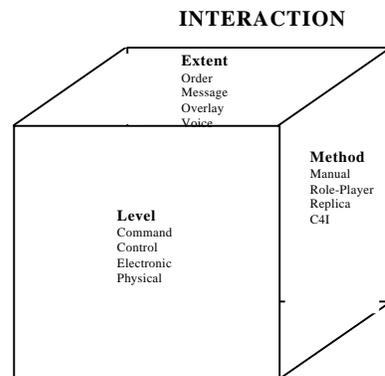
The second dimension in the military class is that of Function. Within each service, specific functions have been isolated and have created their own operational environment and culture. They have specialized themselves in order to better perform their mission, but this makes them less like the other functional areas and generates barriers to communication and interoperation which must be surmounted. Some of these functional areas include: Maneuver, Intelligence, Air Defense, Artillery, Engineering, Aviation, and Combat Service Support.

The third dimension in the military class is Operations. Simulations exist which correctly describe the interactions of forces in large scale

combat, but incorrectly model small contingency operations. In these cases it is necessary to join two simulations which portray specific operations accurately. Some of the operations that must be interoperable are: High Intensity Conflict, Low Intensity Conflict, Operations Other Than War, Law Enforcement, Intelligence Collection, Diplomatic Negotiations, and Psychological Operations.

In any given simulation there may be characteristics from all of these dimensions modeled, though it appears to be common to find certain characteristics modeled together, such as Army-Artillery-HIC or Special Forces-Aviation-OOTW.

**Interaction Dimensions.** The next class of dimensions is Interactions with simulations (Figure 3). These are labeled as the Level, Method, and Extent of interaction between the simulation system (software and hardware) and the humans who are using it.



**Figure 3. Interaction Class Cube**

The Level of Interaction dimension describes the training audiences mental perspective when working with the simulation. At the Command level the trainees are interested in managing large groups of entities that may be aggregated into units. He/she is interested in telling the units what to do and seeing how they responds to those commands. This level is most common in constructive, staff training simulations. For finer detail there is the Control level in which the trainee is actually controlling a simulated device and the intent is to wrap the simulation around them without their noticing that it is not real. This level is most common in virtual vehicle simulators. In the Electronic level the trainee has an electronic connection to the simulation. A computer, radio, radar screen, or other device is their window into the combat world

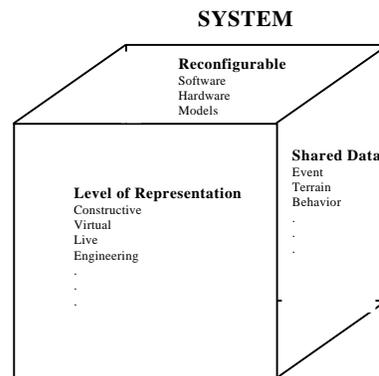
and their physical surroundings are primarily insignificant to their mission. Lower still is the Physical level in which the trainee is actually in a combat environment which is real up to the point of lethal exchanges. A soldier at the National Training Center is in this position, the surrounding environment has everything to do with his mission and human eyes and ears are the window into the world. Finally, there is a Listen-Only level in which the soldier is expected to receive information to analyze and disseminate. He/she is not expected to react to the incoming events, merely to record them.

The Method of Interaction is the second dimension in this class. It describes the actual tools that are used to connect the trainee to the simulation. The Level dimension above often implies the use of certain tools based on past experience, but the focus there is on the humans mental perspective. Here we are interested in connecting the tools so they can be used in the simulation. The Manual method is the most ancient. Here the soldier is directly manipulating the battlefield as it lays in front of him as with board wargames, chess, and computerized versions of these games. The Role-Player method is used to isolate the trainee from the artificialities of the simulation, placing a human between the trainee and the simulation system. This person plays the part of an actor maintaining the realistic environment for the trainee. The Replica method constructs a shell around the trainee so that the simulator looks and feels like real combat equipment. This is the method of virtual vehicle simulators. The C4I System method is a hybrid of the Replica and Role-Player. It attempts to provide the same realistic interaction to command and staff players that has been experienced by vehicle crews. However, it must interact with the trainee at a much higher mental level, appearing to converse with him/her just as the role-player did in the past.

The Extent of Interaction is the dimension that describes the detail to which the above methods are carried. This dimension determines the depth, richness, and realism that the trainee experiences in the Method dimension. The poorest Extent is to provide order parsing where the simulation understands only the syntax developed specifically to operate it. This tends to corrupt realism and call for the use of a role-player to negate the effects. The Message Parsing extent allows the simulation to receive stimuli from the trainee in a form that comes from the real world. For staff trainers this may be tactical messages. For crew trainers this may be

steering and firing signals from real equipment. Overlay Parsing provides an extent which allows the exchange of much more complex information. It augments the information content to include pictorial information as well as text. Voice Recognition allows both staff and crew trainers to control the simulation with the same stimuli that they would use to control real world assets and equipment. Adding Voice Generation closes the loop and allows the simulation to respond to human voice input in kind.

**System Dimensions.** The System class of dimensions deals with the implementation of the simulation (Figure 4). Different capabilities in interoperability are dependent upon how a simulation is actually created and how the real world is represented. The first of these dimensions is the classic Level of Representation. This has been divided conveniently into the categories: Constructive, Virtual, Live, Analytic, Testing, and Engineering. These usually describe the structure of the model as chosen to provide the answers or functionality needed by its creators and customers. Simulations at different levels of representation face the problem that information available on events or objects is not compatible with other simulations. In the best case, the information in one is an aggregate or disaggregate of the information in another, and there is some hope of joining them. In other cases the information is in no way related, functionality provided in one was considered inappropriate or insignificant in the other. In some cases, this also results in a situation in which there is no mission need to join these systems.



**Figure 4. System Class Cube**

The next dimension of this class is Shared Data. In the past, the information exchanged between simulations has been related to specific events. This

has left environmental and behavioral data uncorrelated. For example, the terrain in one simulator may look very different than the terrain in another, which may be due to differences in the data sources or differences in the interpretation of data sources. The same is true of the visual models of objects, light intensity, and shadowing in different simulations. Significant differences may also exist in the implementations of behavior, making it more effective to attack a unit in simulator A rather than in B because it is known that simulator A has a more passive behavior, or integrates the effects of morale and fatigue, where simulator B does not.

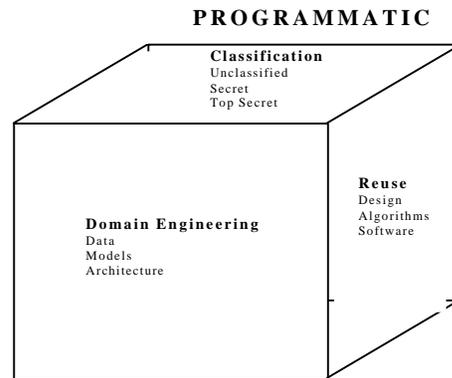
The last dimension in this class is Reconfigurability. This defines the ability of the simulator to be used in more than one configuration. At the hardware level this speaks to the ability to host a whole family of simulations on different sets of computer and communication networks. This may allow the instantiation of a capability at one location, or a rotation of instantiations to support varying training missions. At the software level this describes the ability to change the simulator to represent more than one vehicle, level of representation, method of interaction, etc. Most simulators are built to operate in a single configuration with one customer and one basic mission. It may be possible to extend this to allow a single system to serve multiple missions and customers.

**Programmatic Dimensions.** The

Programmatic class discusses issues external to the actual use of the simulation and focuses more on its design and architecture (Figure 5). The first of these is the Domain Engineering dimension. The architecture of a simulation can be constructed uniquely to serve a single customer, or it may be part of a larger interoperable architecture that joins the proponents of many simulators. If this dimension is exploited, a large of set of simulation interactions are satisfied in the design of the system and do not have to be addressed separately, and less effectively, during individual system implementation.

Reuse is the second dimension of this class. It measures the degree to which simulation capability is shared among multiple simulations. Where reuse is high there is the potential for greater consistency in both the environment and behavior of simulators as described in the Shared Data dimension above. Reuse may be at several levels: design that defines capability, algorithms that control activities, software

that implements algorithms, and hardware that hosts software.

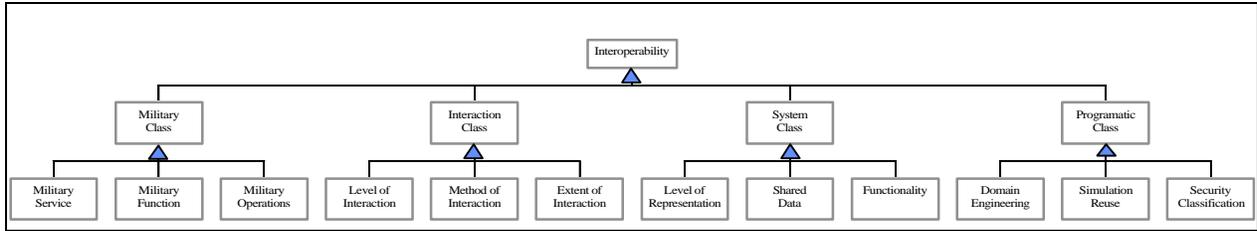


**Figure 5. Programmatic Class Cube**

Finally, Classification divides simulations according to the sensitive nature of the information and capability they portray. This is related to the Military Function and Shared Data dimensions described above, however, due to its unique position it must be addressed and bridged separately. In the past, it has been rare for simulations at different levels of classification to work together, though there are instance of this. Where it is allowed, the classification dimension has always been significant and separate from other dimensions of interoperability.

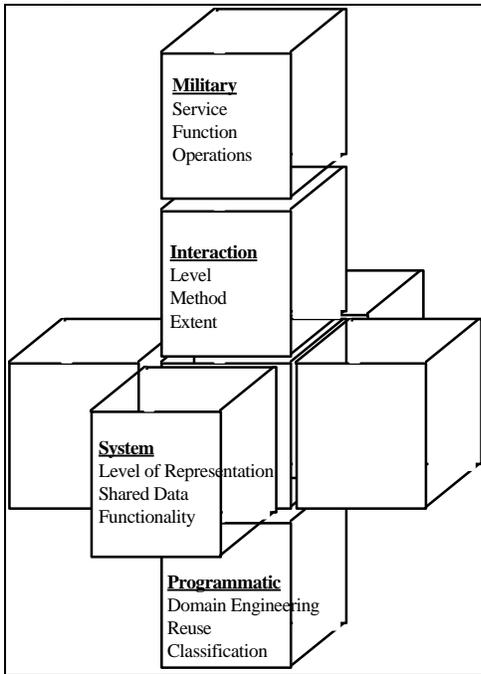
**Graphic Representation**

The sections above have organized the dimensions into classes to aid in describing them. Here we would like to give a graphic representation to these dimensions to aid in their manipulation, addition, and organization. We are providing two different pictures of the interoperability space, the first is a traditional class diagram (Figure 6) which implies that each dimension shares some base characteristics with the others in the class. This may be true, but it is too early to determine whether the correct class groupings have been reached. It also begs the question of what these shared characteristics are, a question we are not prepared to answer in this paper.



**Figure 6. Dimensional Class Structure**

The second diagram (Figure 7) is derived from a four dimensional hypercube that has been unfolded into three dimensional space. This was first described in 1940 by Robert Heinlein in his classic short story “And He Built a Crooked House”. It implies that there are definite adjacencies between dimensions as shown in the three dimensional picture and there are adjacencies that occur when all dimensions are integrated into a single system, similar to the effect of folding the 3-D diagram into a single 4-D cube. This diagram may be more useful for illustration of relationships than for general work on improving relationship definitions.



**Figure 7. Dimensional Hypercube**

### Dimensional Algebra

In his 1976 book *Theory of Modeling and Simulation*, Bernard Zeigler developed a formal specification for simulations. It is desirable to have a similar specification for the dimensions of

interoperability to aid in the definition of ideas and their manipulation. This has proven to be quite difficult and the specification provided here is the shadow of a beginning.

Interoperability, I, is a function of many different variables. In this paper these are grouped and characterized as dimensions, d. The dimensions are grouped into classes, c, to illustrate relationships and potential dependencies. Each dimension has discrete instances, i, and each instance consists of a set of variables, v, as illustrated in Equation Box 1.

A specific instance of interoperability, such as marine corps models, would be written as:

$$I(c\text{-military}(d\text{-service}(i\text{-marines}()))).$$

A specific variable within the marine corps model would add one more level of detail than has not been discussed in this paper:

$$I(c\text{-military}(d\text{-service}(i\text{-marines}(v\text{-org structure}))))).$$

### STRUCTURED APPROACH

The fields of biology and chemistry struggled to impose structure on their problem space for centuries. Some of those taxonomies are useful to today’s simulation community as we try to do the same thing. As we seek to define object models of the simulation, federation, and domain we should recognize that the biologic community has developed a hierarchical taxonomy to group animals and plants so they can work with them in an organized fashion, and the chemistry community has created the table of atomic elements, both of which perform a similar function to our object models.

### Equation 1. Dimensional Algebra Illustration

$$\begin{aligned}
 I &= f(\text{dimensions}) \Rightarrow \\
 &I(\text{c-military, c-interaction, c-system, c-program}) \Rightarrow \\
 &\quad \text{c-military}(\text{d-service, d-function, d-operations}), \text{c-interaction}(\text{d-level, d-method, d-extent}), \\
 &\quad \text{c-system}(\text{d-rep, d-share, d-reconfig}), \text{c-program}(\text{d-domain, d-reuse, d-classification}) \Rightarrow \\
 &\quad \quad \text{d-service}(\text{i-army, i-af, i-navy, i-marines, ...}), \text{d-function}(\text{i-mvr, i-css, i-intel, ...}), \\
 &\quad \quad \text{d-operations}(\text{i-hic, i-lic, i-ootw, ...}), \dots \Rightarrow \\
 &\quad \quad \quad \text{i-marines}(\text{v-org structure, v-culture, v-mission, ...})
 \end{aligned}$$

I = Interoperability

c-\* = four classes of interoperability described in paper

d-\* = individual domains within each class

i-\* = discrete instances within each dimension

v-\* = simulation variables which define instances

The simulation community has sought to define a structure that is useful for creating models of different problem spaces, as in Zeigler's book. One of the difficulties of creating a taxonomy of simulation is caused by the breadth of the field. We are apt to create a simulation of any process in the world, a problem space that would require a World Object Model (WOM) to completely support it. Even within the military training community there is a vast ocean of possible modeling needs. The use of a domain object model and domain architecture is a step that is being taken within the Joint Simulation System (JSIMS) program. This will attempt to pull together the divergent, service-specific models and give them a level of commonality in the architecture which will support interoperability and reuse in other dimensions.

### CONCLUSION

The dimensions of simulation interoperability will continue to increase as the demands we place on them increase and the military continues to computerize its forces. The need to define, structure, and manage these dimensions is becoming more evident all the time. Only by organizing this aspect of simulation can we hope to control the development cost, harness the potential of software reuse, and implement domain architectures.

The dimensional algebra described is only the seed of the type of formal definition that is needed to aid in managing and manipulating dimensions of

interoperability. When this is coupled with a structured object modeling approach it will allow us to identify simulation capabilities that are useful beyond the military training domain (Table 1). Though often referred to as technology transfer, these civilian similarities are an acknowledgment that the US military is an integrated part of the American society and is attempting to solve some of the same problems faced in many other areas.

**Table 1. Civilian Applications**

<b>Military Function</b>	<b>Civilian Application</b>
Combat Service Support	Product Distribution, Medical Services, Social Services
Command, Control, and Communications	Telephone System, Internet, Electronic Banking
Maneuver	Emergency Management, Prison Riots, Highway System
Intelligence	Business Strategy, Market Analysis
Air Defense	Federal Aviation Administration
Engineers	Construction, Emergency Management, Civil Services
Electronic Warfare	Radio and Television Spectrum Analysis
Aviation	Airline Route Planning
Space Operations	Communications Satellites

## BIBLIOGRAPHY

Abbott, Edwin. 1884. *Flatland: A Romance of Many Dimensions*. 1983 Reprint. Barnes & Noble. New York, NY.

Ben-Natan, Ron. 1995. *CORBA: A Guide to Common Object Request Broker Architecture*. McGraw Hill. New York, NY.

Defense Modeling and Simulation Office. 1996. High Level Architecture - Object Model Templates. DMSO. Alexandria, VA.

Defense Modeling and Simulation Office. 1995. Department of Defense High Level Architecture for Modeling and Simulation, Version 0.2. DMSO. Alexandria, VA.

Defense Modeling and Simulation Office. 1996. HLA Management Plan: High Level Architecture for Modeling and Simulation. DMSO. Alexandria, VA.

Feustel, Edward. 1995. Common Model of the Mission Space - Working Draft v.3. Modeling and Simulation Information System. Orlando, Florida.

Heinlein, Robert A. 1940. And He Built a Crooked House. 1983 reprint in *The Unpleasant Profession of Jonathan Hoag*. Berkley Books. New York, NY.

McGarry, Stephen. 1995. The DoD High Level Architecture (HLA) Run-Time Infrastructure (RTI) and Its Relationship to Distributed Simulation. *Summary Report - 13th Workshop on Standards for the Interoperability of Distributed Simulations*. Institute for Simulation and Training. Orlando, FL.

Mowbray, Thomas J. and Zahavi, Ron. 1995. *The Essential CORBA: System Integration Using Distributed Objects*. John Wiley & Sons Inc. New York, New York.

Plato. 1937. "The Republic". *The Dialogues of Plato*. Translated by B. Jowett. Random House. New York, NY. (Originally written circa 380 B.C.)

Rucker, Rudy. 1984. *The Fourth Dimension: A Guided Tour of the Higher Universes*. Houghton Mifflin Company. Boston, MA.

Smith, Roger D. 1995. The Conflict Between Heterogeneous Simulations and Interoperability. *Proceedings of the 1995 Interservice / Industry*

*Training Systems and Education Conference*. Albuquerque, NM.

Smith, Roger D. 1995. Techniques in Simulation Interoperability, *1995 Georgia Tech Course on Modeling, Simulation, and Gaming of Warfare*. Georgia Tech Research Institute. Atlanta, GA.

Smith, Roger D. 1996. *Proceedings of the Electronic Conference on Interoperability in Training Simulation*. Mystech Associates.  
<http://www.mystech.com/~smithr/elecsim96>.

Tracz, Will. 1995. *Confessions of a Used Program Salesman: Institutionalizing Software Reuse*. Addison-Wesley Publishing company. Reading, MA.

Zeigler, Bernard P. 1976. *Theory of Modelling and Simulation*. Robert E. Krieger Publishing Company. Malabar, FL.