STANDARD PROTOCOL DATA UNITS FOR ENTITY INFORMATION AND INTERACTION IN A DISTRIBUTED INTERACTIVE SIMULATION

L. Bruce McDonald, Ph.D. • Christina Pinon Bouwens
Institute For Simulation and Training • University of Central Florida

Ronald Hofer, Ph.D. • Gene Wiehagen • Karen Danisas
Army Project Manager For Training Devices

Colonel James Shiflett
OSD Defense Models and Simulations Office

ABSTRACT

PM TRADE and DARPA have funded the University of Central Florida Institute for Simulation and Training (UCF/IST) to develop a Draft Standard for the Interoperability of Defense Simulations at the protocol data unit level. The second draft of the standard was completed in February 1991. The consensus of government and industry opinion was that the document represented a major step forward toward interoperability of dissimilar simulations. Based on inputs from government, industry and academia at four workshops, IST has developed a final draft standard for submittal as a DoD standard. This paper presents the contents of this standard, its intended use and its anticipated impact on the simulation and training industry. Discussion of the standard's contents include the protocol data units, their intended use and the underlying communications architecture. The paper also addresses future revisions of the standard and the attendant expanded capabilities.

INTRODUCTION

Over the last two decades, the United States military has developed an impressive array of simulation and training systems. These devices are extremely adept at training soldiers to do their jobs as individuals or as members of a small team. In addition, the test community has developed simulations that test the ability of equipment to perform its mission as an individual unit. However, the United States found in Grenada, Libya and Panama that the ability to perform a mission as an individual does not guarantee the ability to function as a member of a coordinated combined arms force.

The United States military has developed means for conducting large combined arms, multi-service exercises. However, these exercises are extremely expensive, subject to major environmental constraints, and can sometimes be interpreted as militarily provocative. DARPA and PM TRADE have developed Distributed Interactive Simulation (DIS), based on SIMNET technology, that will be used to train individuals to function as part of a large coordinated force. Also, DIS will allow the testing of developmental systems under realistic combat conditions. DIS can be used as a substitute for some field training and can allow for practice of warfighting skills when cost, safety, environmental and political constraints will not permit the field training required to maintain readiness.

DIS will take advantage of currently installed and future simulations manufactured by different organizations. Consequently, a means must be found for assuring interoperability between dissimilar simulations. The first step in achieving this interoperability is to develop a communications protocol. There must be an agreed-upon set of messages that communicate between host computers, the states of simulated and real entities, and their interactions.

DARPA and PM TRADE have funded the University of Central Florida Institute for Simulation and Training (UCF/IST) to develop a Draft Standard for the Interoperability of Defense Simulations at the protocol data unit level. The second draft of the standard was completed in February 1991. The consensus of government and industry opinion was that the document represented a major step forward toward interoperability of dissimilar Simulation.

A standard communications protocol must be developed for these dissimilar simulations to communicate with each other. The objective of the standard addressed in this paper is to define this communications protocol at the protocol data unit level. Since the emerging standard is primarily concerned with interoperability, the concept of Open Systems has become an important issue. This subject has been dealt with quite thoroughly by the International Organization for Standardization (ISO), whose primary concern is the communications between heterogeneous computer systems developed by different vendors. ISO's efforts have led to the development of the Open Systems Interconnection (OSI) Reference Model. The standard was written with the assumption that the protocol will be implemented as part of the application layer of the OSI Reference Model.

The standardization process and recommendations for Distributed Interactive Simulation (DIS) are discussed below under the headings of History, Scope, Requirements, Protocol Data Units and Areas for Further Research.

HISTORY

The current work on standards began in August 1989 with the first workshop on Standards for the Interoperability of Defense Simulations. A second workshop took place in January 1990. As a result of these workshops and subsequent subgroup meetings, over 70 position papers containing recommendations for the standard were submitted to the Institute for Simulation and Training (IST). Using the work of SIMNET as a baseline and considering recommendations made in meetings and position papers, IST developed the first draft for a military standard which describes the form and types of messages to be exchanged between simulated entities in a Distributed Interactive Simulation (DIS). This draft standard was distributed to industry and government for review and comment in June 1990.

Workshop Reviews of Standard

A third workshop was conducted in August 1990 in which industry and government provided feedback on the proposed standard. These comments were incorporated into the standard and the final draft standard was submitted in January 1991 for approval by the workshop working groups. The working groups approved the final draft standard with minor changes, which are now being incorporated by IST. Recommended changes to the final standard are listed below.

Collision PDU - Change the mass field from 64-bit floating point to 32-bit floating point.

Update Threshold PDU - Delete this PDU until it is tested and include in later revision.

Radar PDU - Place in the optional section of the standard.

Detonation PDU - Articulated parts record should be included to indicate affected articulated parts. Remove energy, directionality and momentum fields.

Entity State PDU - Include force ID and guise type and modify articulated parts record.

MIL-STD Approval Process

This document will be submitted to the three military services to serve as the baseline standard. After approval by the services, the standard will be submitted for approval as a DoD standard. During this DoD standards approval process, the workshops will continue and a revision one and two will be developed with expanded capabilities. These revisions to the standard will also be submitted for approval as a DoD standard. We also intend to develop standards for the required correlation between simulated environments in different simulators as well as performance measures for evaluating the actions of the participants.

SCOPE OF STANDARD

The standard addressed in this paper establishes the requirements for data units exchanged between simulated elements in a distributed interactive simulation. It encompasses a portion of the application layer of a communications architecture as defined by the International Organization for Standardization's (ISO) Open Systems Interconnection (OSI) Reference Model.

REQUIREMENTS FOR DISTRIBUTED INTERACTIVE SIMULATION

The term Distributed Interactive Simulation refers to an architectural approach in which a simulation is distributed across a number of independent and self-sufficient computers instead of just one central computer. The term interactive reflects how these computers constantly interact by sending messages describing the current state of the simulation entities under their control. These messages allow the other computers to incorporate these state changes into their simulations.

Distributed Interactive Simulation can be defined as follows:

Distributed Interactive Simulation (DIS) is an exercise involving the interconnection of many simulation devices where the simulated entities are able to interact within a computer generated environment. The simulation devices may be present in one location, interconnected by a Local Area Network (LAN) or may be widely distributed on a Wide Area Network (WAN).

In order to fulfill its functional requirements, DIS must provide:

- Entity Information
- Entity Interaction

A brief description of each requirement follows:

Entity Information

Because of the great variety of simulated entities that will be involved in a single exercise, it is important to be able to transmit detailed information about each entity. This information should include the entity's identity, its orientation, and its appearance to others. Below are classifications of types of information needed.

Types. Since a simulated entity can be a vehicle, a building, a missile, or even a cloud of smoke, a method for identifying the types of entities is needed.

Location and Orientation. The location, orientation, velocity, and acceleration (when appropriate) of a simulated entity are important for its representation by a computer. In order to keep network traffic within acceptable limits, the location and orientation information should contain velocity and sometimes acceleration. This information would allow the receiving computer to model (Dead Reckon) the position of the entity over time (based on last reported velocity and acceleration vector) without requiring constant updates over the network.

Appearances. The appearance of a simulated entity can be expressed in two ways: by the reflection of visible light or by the emission of acoustic or electromagnetic energy such as heat, radar, radio, etc. For example, besides its visual appearance, an entity can be identified by its unique infrared signature. If the exercise is taking place in the ocean, the entity can be identified by the sound it makes. Therefore, a method for communicating the different appearances of an entity is needed.

Entity Interaction

Throughout a simulation exercise simulated entities interact with each other. This interaction may take the form of weapons fire, update rate control, logistics support, or collisions.

Weapons Fire. When a simulated entity fires its weapon, its simulator needs to be able to communicate the location of the firing weapon and the type of munition fired. Depending on the munition type, the firing entity will determine the impact location. Given the munition type and the location of impact, all simulators can then assess their own entity damage.

Logistic Support. Other services such as resupply or repair of vehicles should be represented in a simulated exercise because of their significant impact on the outcome of military engagements. These functions and similar ones are provided by logistics support in a real battle scenario. Similarly, they should be provided by logistics support in a simulated battle.

Collisions. It is necessary to represent the collision of entities in a simulation. When a collision occurs, both entities need to be aware of the collision and each must determine any resulting damage to itself. DIS needs a way to communicate this type of collision information.

PROTOCOL DATA UNITS FOR DISTRIBUTED INTERACTIVE SIMULATION

DIS protocol is used by simulators to communicate information about the simulated world. Table I contains a list of the Protocol Data Units recommended for the standard. They are organized according to the information requirement category of which they are a part (e.g. entity information and entity interaction).

Table I
List of DIS Protocol Data Units

I. Entity Information

  A. Entity State PDU

II. Entity Interaction

  A. Weapons Fire
    1. Fire PDU
    2. Detonation PDU

  B. Logistics Support
    1. Service Request PDU
    2. Resupply Offer PDU
    3. Resupply Received PDU
    4. Resupply Cancel PDU
    5. Repair Complete PDU
    6. Repair Response PDU

  C. Collisions
    1. Collision PDU

A detailed discussion of these PDUs is beyond the scope of this paper. However, a brief discussion of a few important PDUs is presented. The two most important PDUs listed above are the Entity State PDU and the Fire PDU. Each of these example PDUs is discussed separately below.

Entity State PDU

A simulator periodically reports information about an entity it is simulating so that other simulators may correctly depict that entity. This information will be communicated using the ENTITY STATE PDU. Physical entities present in the simulation exercise include platforms, munitions, life forms, and environmental and cultural features. The various subcategories of entity types appear in Table II.

Table II
Entity Sub-types

Platforms
- Land
- Air
- Surface
- Subsurface
- Space

Munitions
- Miscellaneous
- Detonator
- Ballistic
- Guided
  - Anti-Air
  - Anti-Armor
  - Anti-Missile
  - Anti-Radar
  - Anti-Satellite
  - Anti-Ship
  - Anti-Submarine
  - Battlefield Support
  - Strategic
- Petroleum, Oil and Lubricants

Life Forms
- SEALS
- Scouts
- Dismounted Infantry
  - Categorized by Weapon Carried

Environmental
- Smoke
- Fog
- Dust
- Flock of Birds
- Cloud
- Cloud With Rain Falling
- Cloud With Snow Falling
- Thermocline
- Knot
- School of Fish
- Whale
- School of Shrimp

Cultural Features
- Bridge
- Building
- Defensive Embankment
- Crater
- Ditch

The Entity State PDU will be issued by a simulator when the following conditions exist:

1. The discrepancy between an entity's high fidelity model and its dead reckoned model exceeds a predetermined threshold (generally occurs when the platform changes its velocity vector).

2. A predetermined amount of time has elapsed since the issuing of the last PDU. The purpose of this issue is to inform new simulated entities of existing entities. It also serves to remind the existing entities that the issuing entity is still active.

Figure 1 lists field contents of the Entity State PDU.

FIELD SIZE (bits)	ENTITY STATE PDU FIELDS	
48	ENTITY ID	SITE - 16 - bit unsigned integer
		HOST - 16 - bit unsigned integer
		ENTITY - 16 - bit unsigned integer
8	PADDING	16 bits unused
8	FORCE ID	8 bits unsigned integer
64	ENTITY TYPE	ENTITY KIND - 8 - bit enumeration
		DOMAIN - 8 - bit enumeration
		COUNTRY - 16 - bit enumeration
		CATEGORY - 8 - bit enumeration
		SUBCATEGORY - 8 - bit enumeration
		SPECIFIC - 8 - bit enumeration
		EXTRA - 8 - bit enumeration
64	ALTERNATE ENTITY TYPE (GUISE)	ENTITY KIND - 8 - bit enumeration
		DOMAIN - 8 - bit enumeration
		COUNTRY - 16 - bit enumeration
		CATEGORY - 8 - bit enumeration
		SUBCATEGORY - 8 - bit enumeration
		SPECIFIC - 8 - bit enumeration
		EXTRA - 8 - bit enumeration
32	TIME STAMP	32 - bit unsigned integer
192	ENTITY LOCATION	X - Component - 64 - bit floating point
		Y - Component - 64 - bit floating point
		Z - Component - 64 - bit floating point
96	ENTITY LINEAR VELOCITY	X - Component - 32 - bit floating point
		Y - Component - 32 - bit floating point
		Z - Component- 32 - bit floating point
96	ENTITY ORIENTATION	Psi - 32 - bit BAM
		Theta - 32 - bit BAM
		Phi - 32 - bit BAM
256	DEAD RECKONING PARAMETERS	Linear Accel - 3 32 - bit floating points
		Angular Velocity - 3 32 - bit signal integers
		64 bits TBD

Figure 1 Entity State PDU

FIELD SIZE (bits)	ENTITY STATE PDU FIELDS (CONT'D)	
32	ENTITY APPEARANCE	32 - bits
96	ENTITY MARKING	Character set 11 Characters
32	CAPABILITIES	32 Boolean fields
24	PADDING	Unused
8	# ARTICULATED PARAMETERS	8 - bits unsigned integer
Varies	ARTICULATED PARAMETERS	Change - 16 bits
		ID - attached to - 16 bits
		Parameter type - 32 bits
		Parameter value - 64 bits

For each Articulated Parameter

Figure 1 Entity State PDU continued

The contents of each of these fields is described below:

General PDU Information - PDU Header

1. Protocol Version: Specifies the version of DIS protocol used in this PDU.

2. Exercise Identification: Specifies the Exercise to which the PDU pertains. This Feature allows multiple exercises to occur on the same network simultaneously.

3. Protocol Data Unit Type: Indicates the type of PDU to follow.

Static Entity Information

1. Entity Identification: Identifies the entity issuing the PDU.

2. Marking: Identifies any unique markings on an entity (e.g. a bumper number or country symbols).

3. Capabilities: Identifies the entity's capabilities in terms of logistics support to other entities.

Dynamic Entity Information

4. Time of Issue: Describes the time at which the PDU was issued.

5. Entity Appearance: Describes the dynamic changes to the entity's appearance such as on fire, destroyed, TOW launcher raised, etc.

6. Entity Location: Describes an entity's physical location in the simulated world.

7. Entity Velocity: Describes an entity's linear velocity in millimeters per second.

8. Entity Orientation: Describes the entity's orientation in terms of three angles.

9. Dead Reckoning Parameters: Elements of this field provide information required for dead reckoning an entity's position and orientation. These parameters consist of the following elements:

   Entity Acceleration. Describes an entity's linear acceleration in millimeters per second squared.
   Entity Angular Velocity: Describes the entity's angular velocity around its own axis.

10. Articulated Parts. Describes the orientation of each articulated part.

FIRE PDU

A FIRE PDU describes the type of munition fired, the location of the weapon from which it was fired, and the velocity of the munition. Also present in the PDU is the target range used for the fire control system and the kind of munition selected to aid analysis of the exercise. The contents of the Fire PDU are listed in Figure 2.

FIELD SIZE (bits)	FIRE PDU FIELDS	
16	EVENT ID	16 bit uns int
48	FIRING ENTITY ID	SITE ID - 16 bit uns int
		HOST - 16 bit uns int
		ENTITY - 16 bit uns int
48	TARGET ENTITY ID	SITE ID - 16 bit uns int
		HOST - 16 bit uns int
		ENTITY - 16 bit uns int
48	MUNITION ID	SITE ID - 16 bit uns int
		HOST - 16 bit uns int
		ENTITY - 16 bit uns int
96	BURST DESCRIPTOR	MUNITION - 32 bit uns int
		DETONATOR - 32 bit uns int
		QUANTITY - 16 bit uns int
		RATE - 16 bit uns int
96	LOCATION	X COORDINATE - 32 bit signed integer
		Y COORDINATE - 32 bit signed integer
		Z COORDINATE - 32 bit signed integer
96	VELOCITY VECTOR	X COORDINATE - 32 bit signed integer
		Y COORDINATE - 32 bit signed integer
		Z COORDINATE - 32 bit signed integer
32	RANGE	32 bit uns int

Figure 2 Fire PDU

The contents of the Fire PDU are described below:

1. Event Identification. Contains a number generated by the firing simulator to associate related events.

2. Firing Entity Identification. Identifies the firing entity.

3. Target Identification. Identifies the intended target.

4. Munition Identification. Gives an ENTITY ID to the munition. The Entity ID identifies the munition as a unique entity.

5. Burst Descriptor. Describes the type of munition fired, the quantity, and rate.

6. Location. Specifies the location from which the munition was launched.

7. Velocity Vector. Specifies the speed in millimeters per second and the direction of the fired munition.

8. Range. Specifies the range (in meters) that an entity's fire control system has assumed in computing the ballistic solution.

9. Timestamp. Specifies the time at which the data is valid.

Modeling the Trajectory of the Munition. There are no PDUs associated with the modeling of the trajectory of a munition. If the munition is the result of Direct or Indirect Fire, only its firing and detonation are reported. If the munition is a Guided munition, Entity State PDUs are transmitted for the munition throughout its flight.

Detonation of the Munitions. A Detonation PDU is issued when the trajectory of the fired munition is terminated. The firing simulator uses this PDU to inform other entities of the munition's detonation or impact location. In this way, other entities are FIGURE 3. Fire PDU informed of the munition's detonation so they might produce the appropriate visual and aural effects and assess damage.

Damage Determination. Once the location and type of detonation has been determined, each entity assesses its own damage based on its location in relation to the detonation. No PDUs are associated with this action.

AREAS FOR FURTHER RESEARCH

Communications Architecture

As stated earlier, the emerging Standard and the preceding workshops were primarily concerned with interoperability and Open Systems. The Open Systems Interconnection (OSI) Reference Model will be used throughout this section for discussion of communication architecture.

The interoperability requirements for communication have been well defined in the OSI reference model. However, DIS needs certain services not currently offered in available OSI protocols and so research must be done to develop them. These required services are described below.

1) Guaranteed Service for Real-Time Simulation. The requirements for DIS are based mainly on the needs of Real-Time Simulation, which requires information on a "timely" basis so that the representation and tracking of objects in the simulation can be accomplished as they are occurring. This requirement calls for a communication architecture that can deliver a message in a timely manner.

2) Multicasting Capabilities. In DIS it is sometimes necessary to send messages to a subset of nodes on the network. If a message is to be sent to all entities, it is sent using broadcast. If the message is to be sent to a specific group (as would be the case if more than one exercise taking place on the same network), the communications method used is termed multicasting.

These services are not currently provided in the OSI model.

3) Appropriate Security Levels. Security is an important requirement for DIS, but many problems remain unresolved. Some of these problems are related to how classified information may be securely transmitted. Others deal with how to keep the entire network secure. The current belief is that commercially available encryption software will be adequate for security requirements, but the real-time performance of this software may be too slow. Research on real-time performance of encryption software is required.

4) Connectionless Service. A connectionless service transmits data by simply sending the data out onto the network and addressing it to the entity(s) that require it. There is no need to establish a connection between simulation entities before transmitting data. This is a requirement for multicast service and is not currently provided in the OSI model.

Emitter PDU

The EMITTER PDU would be issued by the simulator for any platform possessing emitting capability. It is issued when the platform changes its velocity vector or changes the mode of one of its emitters. It is assumed that all simulators requiring emitter information have a database containing information about the operating parameters of emitters in each mode. An example database is the Universal Threat System for Simulators (UTSS).

When an EMITTER PDU is issued, it would include information about the state of all of its emitters for a particular database. Should an emitter from another database change modes, a separate EMITTER PDU would be issued.

Environment Information

For simulated entities to participate in the same exercise, they must all have access to the same environment information. Different types of information about the environment are necessary to make the exercise as realistic as possible. This information may include changes in the terrain, weather, and ambient illumination.

Changes in the Terrain. During the course of a real battle, changes in the terrain occur frequently. An explosion may create a crater or blow up a bridge. Ditches might be dug and defensive embankments may be built. In addition, cultural features such as bridges and buildings could be destroyed or built. All of this information must be available to the participants in a simulated battle just as it would be accessible in a real battle. Therefore, DIS must provide the necessary functions to support dynamic terrain.

Weather Conditions. Weather conditions affect real life battle scenarios. Similarly, they should have an effect on the simulated battle. Conditions such as wind, rain, snow, fog or clouds need representation in a simulated exercise. The wind and its effect on a cloud of smoke that affects visibility or chemicals that affect dismounted infantry need to be represented as well.

Proposed Simulation Management Protocol

IST proposes a Simulation Management Protocol (SIMAN) that could provide many of the services required by DIS.

SIMAN would perform the following functions:

1. Exercise setup
2. Exercise start/restart
3. Exercise maintenance
4. Exercise end

SIMAN would serve as a central database for the simulation. It would record the exercise for purposes of playback, restart and admittance of new entities to the exercise. SIMAN would also perform data logging functions such as updating its database on entity capabilities and changes in the terrain. Further research is required into the requirements for SIMAN functions and the most efficient means of providing these services.

Unmanned Forces

One type of entity that is represented in a simulated battle is the Unmanned Force or Semi-Automated Forces (SAFOR). As simulated entities in the exercise, unmanned forces have many of the same requirements as manned forces. The data messages (PDUs) communicated on the network are the same as those for manned simulators. Unmanned forces, however, have some unique informational and database requirements that other entities do not have. Further research is required before effective semi-automated forces can be added to DIS.

Issues Concerning Fidelity Measures

Fidelity Measures address the allowable delay between operator action and simulated response, as well as the required fidelity for representing the visual appearance or sensor imagery of an entity or the environment. Many fidelity measures issues have been resolved in previous research on individual operator training systems. The three most critical remaining DIS fidelity issues requiring research are delay, entity appearance at long ranges, and depiction of environmental appearances.

Delay. The allowable delay between operator action and simulation response will depend on the criticality of the task being executed by the operator. One of the most time-critical tasks in distributed interactive simulation is tracking a target just prior to firing a weapon. Consequently, the smallest acceptable delay in a DIS will be that between the issuance of an appearance PDU by a target entity and the display of that entity's location on the engaging entity's display. Determination of acceptable delay will require empirical studies of operator performance under varying delay conditions.

Entity Appearance At Long Ranges. One shortcoming of current distributed interactive simulation is that the displays have insufficient resolution to accurately depict entities at long range, thereby preventing the engagement of these entities at a range specified in doctrine. This problem may be solved by using higher resolution displays or by color coding images too small to identify. Determining acceptable means of increasing target identification ranges will require empirical studies of operator performance with alternative modifications to the current approach.

Depiction of Environmental Appearance. The appearances of environmental entities such as smoke, fog, clouds, rain and snow need to be depicted in a manner realistic enough to achieve the training or equipment evaluation objectives. Each of these environmental entities effects visibility to a varying degree based on the density of the entity.

Five levels of density should be sufficient to meet the training and equipment evaluation objectives. Definition of how the visual system will depict the density of these environmental entities should be based on target detection range for each level of density. For example, "Fog with density level three shall produce a 50% target detection probability for the T-72 tank at ____ meters." Further research is required before these values can be stated.

Update Rate Control. The frequency at which one simulated platform must transmit an update of its location and orientation or its emitter status to another platform depends on what task the operator of the simulator is attempting to execute. If the operator of one platform is simply observing the other platform's motion for identification, the exact location of the platform is less critical and frequent updates are not required. However, if the operator is tracking the other platform in preparation for firing or two platforms are flying in close formation, the exact location is critical and a higher update rate is required. DIS needs a means of controlling platform location and orientation update rate in order to meet the requirements for some critical operator tasks without overloading the network while the operator is executing less critical tasks.

CONCLUSIONS

With the increased operating costs of military equipment and reduced budgets, increased use of simulation is needed to maintain readiness. Distributed Interactive Simulation will allow the armed forces to use the installed base of individual training devices to perform large scale team training exercises in a manner similar to SIMNET. The Simulation PDUs of the SIMNET protocol were considered as a baseline for this effort. The University of Central Florida's Institute for Simulation and Training has prepared a Standard (at the Protocol Data Unit level) which will allow dissimilar simulations to interoperate in a Distributed Interactive Simulation.

ABOUT THE AUTHORS

Dr. Bruce McDonald is the Program Manager on the Interoperability Standards program. He has a Ph.D. in Industrial Engineering and 23 years of experience in research and system analysis on training systems and operational equipment. For the last 14 years, he has concentrated on training systems and has completed the following tasks: functional descriptions and detailed specifications for training systems, requirements analysis of war game trainers and instructor station designs. He was program manager for a Catapult Launch Systems Trainer (Device 11F12) and the Principal Investigator on a study to develop Embedded Training capabilities for the AN/SPA-25G Radar system.

Christina Pinon Bouwens is the Program Engineer on the Standards program. She has an M.S. in Mathematics Science and has seven years of experience. She is currently working on communications protocols to support Distributed Interactive Simulation.

Dr. Ronald Hofer is the Chief Scientist at the Army's Project Manager for Training Devices. Dr. Hofer has more than 25 years' experience in simulation and training; he has spent 15 years in this field at PM TRADE. He is also chairman of the Communications Protocols working group which developed recommendations for this standard.

Gene Wiehagen is a System Engineer at the Army's Project Manager for Training Devices and is the BDS-D ATTD Manager in charge of all networking related projects. Mr. Wiehagen has a B.S. in Engineering aand is a member of the Steering Committee for the interoperability of defense simulations.

Karen Danisas has a B.S. in Physics, an M.S. in Simulation Engineering, and has six years of experience in the simulation industry. She is an engineer at the Army's Project Manager for Training Devices and is in charge of standards development for DIS.

Colonel James Shiflett is the Technical Director of the Defense Models and Simulations Office in the Office of The Secretary of Defense and was previously a program manager at the Defense Advanced Research Projects Agency where he was responsible for distributed interactive simulation technology development.