THE STANDARDIZATION OF PROTOCOL DATA UNITS FOR INTEROPERABILITY OF DEFENSE SIMULATIONS

L. Bruce McDonald, Ph.D.
Christina Pinon
Robert Glasgow
Karen Danisas

Institute For Simulation And Training
University of Central Florida
Orlando, Florida

ABSTRACT

The SIMNET program has been a pioneer in "multi-interconnected-simulator" training for the Army. As the benefits of this training have been demonstrated, the Department of Defense (DoD) community has recognized a need for more training of this type and therefore wants to interconnect the current inventory of simulators in a manner similar to SIMNET. In an attempt to answer this need, a standardization process was begun to allow greater interoperability of defense simulations. This project is funded by the Defense Advanced Research Projects Agency (DARPA) and administered by the Army Project Manager for Training Devices (PMTRADE). The current mission at the Institute for Simulation and Training (IST) is to develop a standard for Protocol Data Units (PDU) on the application layer of the communications software. The Simulation PDU's of the SIMNET protocol were considered as a baseline for this effort. This paper discusses the approach taken for development of the Draft Standard, lists and describes the recommended PDU's, and discusses other requirements for interoperability of defense simulations.

INTRODUCTION

As operational equipment has become increasingly complex and more expensive to operate, and as test/training range space has decreased, the U. S. Military has found simulation to be a cost-effective means of training personnel and an aid in the design, development, and testing of materiel. Thus far, these simulations have largely been designed to simulate one unit of operational equipment operating against one or more threats. These types of simulations are ideal for evaluating the operation of equipment or training personnel to operate the equipment as a single entity. However, during the Iranian Rescue Mission and the Grenada Invasion, the military learned that coordinated actions between team members in different services and different units of operational equipment are required in order to achieve the mission objectives.

One alternative for providing a teamwork training and evaluation environment is to build a large simulation computer with multiple operator stations. A less expensive solution is to take the current inventory of individual operator trainers and developmental simulators and link them together.

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, Requirements, Protocol Data Units and Areas for Further Research.

HISTORY

This standards effort is part of a larger project funded by the Defense Advanced Research Projects Agency (DARPA) and administered by the Army Project Manager for Training Devices (PMTRADE). 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 50 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. A third workshop was conducted in August 1990 in which industry and government provided feedback on the proposed standard. These comments are being incorporated into the standard and the final draft standard will be submitted in December 1990 for approval by the three military services. After approval by the services, the standard will be submitted for approval as a DoD standard.

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
- Environment Information

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.

<u>Types</u>. 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.

<u>Location and Orientation</u>. 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.

<u>Appearances</u>. 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.

<u>Weapons Fire</u>. 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.

<u>Update Rate Control</u>. 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.

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.

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 some kind of 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.

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, entity interaction, and environment information).

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

Table I
List of DIS Protocol Data Units

I. Entity Information

- A. Entity Appearance PDU

II. Entity Interaction

- A. Weapons Fire
  1. Fire PDU
  2. Detonation PDU
- B. Update Rate Control
  1. Update Request PDU
  2. Update Response PDU
- C. Logistics Support
  1. Service Request PDU
  2. Service Cancel PDU
  3. Service Complete PDU
  4. Resupply Offer PDU
  5. Repair Offer PDU
- D. Collisions
  1. Collision PDU
- E. Electronic Interaction
  1. Emitter PDU

III. Environment Information

- A. Activation
  1. Activate Request PDU
  2. Activate Response PDU
- C. Deactivation
  1. Deactivate Request PDU
  2. Deactivate Response PDU

Entity Appearance 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 APPEARANCE 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 Appearance 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 Appearance PDU. The contents of each of these fields is described below:

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.

Entity Dynamic 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 Acceleration. Describes an entity's linear acceleration in millimeters per second squared.

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

ENTITY APPEARANCE PDU

FIELD SIZE (bits)	ENTITY APPEARANCE PDU FIELDS	
48	ENTITY ID	SITE ID - 16 bit uns int
		HOST - 16 bit uns int
		ENTITY - 16 bit uns int
16	PADDING	16 bits unused
32	ENTITY TYPE	32 bits uns int
96	ENTITY MARKING	Entity Marking - 12 element character string
32	ENTITY CAPABILTIES	ENTITY CAPABILITIES - 32 bits, boolean characters
32	TIME OF ISSUE	TIME STAMP - 32 bit uns int
32	ENTITY APPEARANCE	APPEARANCE - 32 bit uns int
96	ENTITY LOCATION	X COORDINATE - 32 bit signed integer
		Y COORDINATE - 32 bit signed integer
		Z COORDINATE - 32 bit signed integer
96	ENTITY VELOCITY	X COMPONENT - 32 bit signed integer
		Y COMPONENT - 32 bit signed integer
		Z COMPONENT - 32 bit signed integer
96	ENTITY ACCELERATION	X COMPONENT-32 bit signed integer
		Y COMPONENT-32 bit signed integer
		Z COMPONENT-32 bit signed integer
96	ENTITY ORIENTATION	YAW ANGLE - 32 bit BAM
		PITCH ANGLE - 32 bit BAM
		ROLL ANGLE - 32 bit BAM
96	ENTITY ANGULAR VELOCITY	YAW RATE - 32 bit signed integer
		PITCH RATE - 32 bit signed integer
		ROLL RATE - 32 bit signed integer
16+(N)(16) (N = Number of Articulated Parts)	ARTICULATED PARTS	#articulated parts*-16 bits uns int
		Part 1 AZIMUTH - 8 bit BAM Part 1 ELEVATION - 8 bit BAM
		Part 2 AZIMUTH - 8 bit BAM Part 2 ELEVATION - 8 bit BAM • • •
		Part N AZIMUTH - 8 bit BAM Part N ELEVATION - 8 bit BAM

0049-0958-0959

FIGURE 1 Entity Appearance PDU

* NOTE:

For even numbers of articulated parts, a 16 bit padding shall be added to maintain 32 bit boundaries.

10. Entity Angular Velocity: Describes the entity's angular velocity around its own axis.

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

Emitter PDU

The EMITTER PDU is 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 includes 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 will be issued.

The contents of each field in the Emitter PDU are listed in Figure 2 and described below.

1. Emitting Entity Identification: Identifies the entity that is issuing the EMITTER PDU.

2. Time of Emission: Describes the time at which the PDU was issued.

3. Location: Describes the location of the issuing entity.

4. Emitter Type: Indicates the number of emitters, the emitter class, the database entry number, and the mode number for each emitter. (Emitter Class is delineated in Table III.)

Emitter Class is included in the PDU so that a computer can quickly discard a PDU that contains information about sensors the platform does not have. For example, land vehicles have no way of sensing Very Low Frequency emissions from submarines, so the computer need not spend time searching for data in an emitter table.

EMITTER PDU

FIELD SIZE (bits)	EMITTER PDU FIELDS	
48	EMITTING ENTITY ID	SITE ID-16 bit uns int
		HOST-16 bit uns int
		ENTITY-16 bit uns int
16	PADDING	16 bits unused
32	ENTITY TYPE	32 bit uns int
32	TIME OF EMISSION	TIME STAMP-32 bit uns int
96	EMITTER LOCATION	X COORDINATE-32 bit signed integer
		Y COORDINATE-32 bit signed integer
		Z COORDINATE-32 bit signed integer
32+N(32) (N=# OF EMITTERS)	# OF EMITTERS	16 bits uns int
	DATABASE#	16 bits uns int
	DATABASE ACCESS INFORMATION (FOR EACH EMITTER)	CLASS-8 bit enum int
		DBASE ENTRY#-16 bit uns int
		MODE#: 8 bit uns int

0049-0971

FIGURE 2 Emitter PDU

TABLE III
Electromagnetic Spectrum: Emitter Classes

Emitter Class	Region	Center Wavelength
0	Other	---------
1	Acoustic	---------
2	Infrasonic	10^9 m
3	Sonic and VLF	10^6 m
4	Ultrasonic and LF	10^4 m
5	Radio and HF	100 m
6	Television and UHF	1 m
7	Radar and SHF	100 milli-m
8	Experimental and EHF	1 milli-m
9	Infrared	10-100 micro-m
10	Visible	1000 nano-m
11	Ultraviolet	10 nano-m
12	X Ray	100 pico-m
13	Gamma Ray	1 pico-m
14	Secondary Cosmic Ray	0.01 pico-m

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 3 and 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.

FIRE PDU

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

Figure 3 continued on next page

Figure 3 continued

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

0049-0960

FIGURE 3 Fire PDU

<u>Modeling the Trajectory of the Munition</u>. 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 Appearance and Emitter PDUs are transmitted for the munition throughout its flight.

<u>Detonation of the Munitions</u> . 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 munitions's detonation or impact location. In this way, other entities are informed of the munition's detonation so they might produce the appropriate visual and aural effects and assess damage.

<u>Damage Determination</u>. Once the location and typé 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) <u>Guaranteed Service for Real-Time Simulation</u>. 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) <u>Multicasting Capabilities</u>. 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) <u>Appropriate Security Levels</u>. 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) <u>Connectionless Service</u>. 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.

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.

<u>Delay</u>. 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.

<u>Entity Appearance At Long Ranges</u>. 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.

<u>Depiction of Environmental Appearance</u>. 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.

CHANGES RECOMMENDED AT AUGUST WORKSHOP

Thirteen position papers were submitted to the Institute for Simulation and Training in response to the June release of the Draft Standard. At the August workshop, these papers were used as a springboard for discussion concerning issues that needed to be resolved for the January draft. As a result of the workshop, the following recommendations were made for incorporation into the draft standard.

General Changes

In general, the draft standard was accepted as a very important step toward interoperability of defense simulations. There were several general recommendations made:

1) An overall system design will be done as a top-down approach to Distributed Interactive Simulation. A summary of the design description will be given in the rationale document. There is a possibility that a handbook may be published at a later time to give the details.

2) A description of the communications requirements will be included as part of the standard. A subgroup called the Communication Architecture subgroup is working on the details for submission to the standards group. Some services being considered are:
    a) Multicast
    b) High Bandwidth (1 Mbps now, 10 Mbps in 1-2 yrs)
    c) Real Time Delivery
        - 100-250 ms end-to-end delay
        - minimal delay variability

d) Low Packet Loss
e) Wide Area Coverage
f) Security

3) Closer work with the services to determine the specific requirements for the system was recommended.

4) A testbed and eventually a means to validate the standard should be developed. IST is presently working on this very issue.

Other recommendations

Other recommendations being considered:

1) The use of the ACME RADAR PDU (used at AFHRL Williams AFB) as presented at the workshop as an interim solution until a database can be developed for the present EMITTER PDU.

2) The use of two PDUs to relay appearance information. A STATIC APPEARANCE PDU would be issued no more that once every minute. It would communicate information that does not change during the course of an exercise. It is also recommended that bit encoding not be used in the STATIC APPEARANCE PDU, but that a hierarchy of character strings be used. A DYNAMIC APPEARANCE PDU would be used to relay changing information. It would be issued under the same conditions that the APPEARANCE PDU as proposed in the first draft of the standard.

3) The decision to use Euler angles was confirmed during the workshop. A presentation was made arguing for the use of Quaternians to communicate orientation. The standard does not exclude the use of quaternians within a simulator in order to perform calculations. It will specify that Euler angles will be used to communicate platform orientation.

4) The decision to use fixed point numbers was overturned. The standard will use 64 bit floating point number where they naturally occur. Instances where floating point will not be used include expressing angles (which will still be done in BAMs) and counters.

5) Dead reckoning algorithms will be specified by the standard. These are being developed by a subgroup.

6) A more efficient way of expressing articulated parts is recommended. A subgroup is working on a viable alternative. In the interim, articulated parts will remain as is with 16 bit BAMs instead of 8 bits to express the orientation of the articulated part. Specifications accompanying articulated parts will be detailed further (such as default values and upper limits).

7) Repair and Resupply functions will be further described using state diagrams. Additional PDUs will be required to acknowledge receipt of supplies or repairs.

8) The DETONATION PDU will be modified to accommodate direct and indirect fire situations. In the case of direct fire, the entity that is hit (as determined by the firing entity) will be informed. The location of the hit will be given in the entity's coordinate system.

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 PDU's 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

Bruce McDonald-- Bruce is the Program Manager on the Interoperability Standards program. He has a Ph.D. in Industrial Engineering and 22 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-- Christina is the Program Engineer on the Standards program. She has a B.S. in Mathematics and has six years of experience in the mathematics field. She recently completed an M.S. in Mathematical Science at the University of Central Florida.

Robert Glasgow-- Bob has a B.S. in Industrial Engineering and has nine years of experience in the simulation industry. He is completing his M.S. in Industrial Engineering at the University of Central Florida.

Karen Danisas-- Karen has a B.S. in Physics and has five years of experience in the simulation industry. She has also completed her M.S. in Simulation Engineering at the University of Central Florida.