BEYOND VISUAL RANGE EXTENSIONS TO DIS

Ken Doris
Grace Mak-Cheng
Grumman Aerospace and Electronics

ABSTRACT

The successful 1992 I/ITSEC demonstration of DIS was a significant milestone in the development of the DIS protocols, proving that Version 1.0 of the standard is truly workable. Although the plans for the 1993 I/ITSEC demonstration focus on long-haul and live participant involvement, a vital ingredient to the eventual success of distributed simulation lies in the ability of subsequent versions of DIS to adequately support beyond visual range (BVR) encounters.

Simulation of BVR effects within the DIS context offers substantial increases to training effectiveness, tactics development, and improvements to the acquisition process. To achieve these goals we must overcome a new set of challenges. SIMNET, the predecessor to DIS, provided a solid background in the development of version 1.0 of DIS, but was limited to within visual range encounters. The BVR extensions found in DIS Version 2.0 can thus borrow little from the SIMNET legacy. New problems, such as sensor simulation, EW data base correlation, and environmental effects must be addressed.

This paper provides insight into the key issues associated with extending DIS to encompass the beyond visual range arena. In addition, it describes series of rapid prototype implementations of the Emitter, Transmitter, and Signal PDUs, starting with a joint Grumman/NTSC experiment held on the last day of the 1992 I/ITSEC show. The "lessons learned" from these implementations are discussed along with suggestions and guidelines for future development of BVR PDUs and associated data bases.

ABOUT THE AUTHORS

Kenneth Doris is a Technical Advisor in Grumman's Combat Systems organization. He is currently directing several research projects, including one devoted to DIS investigation. He is an active member of both the Communications Architecture and the Emissions subgroups of DIS. Last fall Mr. Doris lead the Grumman team at the 14th I/ITSEC DIS demonstration held in San Antonio. He holds a Bachelor of Electrical Engineering from Rensselaer Polytechnic Institute and has twenty-five years experience in Simulation and C^3I, specializing in computer architecture and software engineering.

Grace Mak-Cheng is an Engineering Specialist in Grumman's Combat Systems organization. She is the Principal Investigator for an Independent Research and Development project devoted to DIS and networked simulation and actively participates in several DIS working groups . Ms. Mak-Cheng holds a Masters in Computer Science from New York Institute of Technology and a Bachelor of Electrical Engineering from New York Polytechnic Institute of Technology. She is an expert in the use of structured analysis and structured design software methodologies, especially as applied to embedded systems.

BEYOND VISUAL RANGE EXTENSIONS TO DIS

Ken Doris
Grace Mak-Cheng
Grumman Aerospace and Electronics

INTRODUCTION

DIS Version 1.0 (IEEE-93-1278) and its predecessor SIMNET have proven their worth in simulation of direct fire encounters, those in which the participants are within visual range of their opponents. Such encounters are the "end game" of any conflict and thus are vital to success. To reach that end game in a favorable position however, requires victory in conflicts which are beyond visual range (BVR). Consider the progress of Desert Storm. Its opening round was a series of attacks on Iraqi radar sites, denying the enemy many of its BVR "eyes". All Air Force and Navy strikes that followed were typically supported by EW jamming aircraft to further blind the enemy's electronic sensors. Iraq's main retaliatory tactic, the use of SCUD missiles, was thwarted primarily through the BVR success of the Patriot missile system.

Desert Storm cannot be taken as a model for all future conflicts since it was almost exclusively a land war. In the air and on/under the sea, Allied forces acted with virtual impunity - we cannot expect that luxury in the future. We must ensure that we are as well prepared to fight in those arenas as we were for the armored battles on land. SIMNET and DIS 1.0 have no provisions to support such conflicts. DIS Version 2.0 and beyond must provide these additional elements.

This paper explores some of the key issues associated with adding the BVR extensions to DIS. In addition, it describes series of rapid prototype implementations of the Emission, Transmitter, and Signal PDUs, starting with a joint Grumman/NTSC experiment held on the last day of the 1992 I/ITSEC show. The "lessons learned" from these implementations are discussed along with suggestions and guidelines for future development.

BEYOND VISUAL RANGE SIMULATION ISSUES

The DIS community is in the process of developing the additional message types (called Protocol Data Units or PDUs) and databases to support simulation of electronic, thermal, and acoustic emissions. Along with these there is a corresponding effort to develop techniques of simulating the environmental conditions that affect them.

Electronic Emissions Simulation

In the area of Electronic Emissions the DIS working groups have been concentrating on the simulation of radar and EW sensors and radio communications. In its current state the latest DIS standard defines three types of PDUs in this field:

- Emissions PDU
- Transmitter PDU
- Signal PDU

Emissions PDU

The Emission PDU has been under development since 1991[1]. At that time a Radar PDU also existed. As a result of the of the work performed by the Emissions subgroup of DIS, the Radar PDU was deleted and its requirements incorporated into the Emissions PDU (EMPDU). It was originally hoped that this single PDU could be designed to handle all types of emissions (including radar, EW, radios, infrared and acoustics). It quickly became evident that this would be unmanageable, and its scope was limited to radar, EW (near-term) and acoustics (future). Even given that limitation, the EMPDU is probably one of the most complex and little understood of all DIS PDUs.

The basic premise of the EMPDU is that every emitter is associated with a given entity. A *Simulation Host* is responsible for the simulation of the entity and its emitters (the entity may have one or more emitters "on-board"). The EMPDU was therefore designed to represent multiple emitters for a single entity, and for each emitter, multiple beams, as shown in the simplified diagram below:

Emitting Entity ID
Number of Systems (N)
Emitter #1
Number of Beams (M_1)
Beam #1 Parametrics
⋮
Beam #M_1 Parametrics
Emitter #N
Number of Beams (M_2)
Beam #1 Parametrics
⋮
Beam #M_2 Parametrics

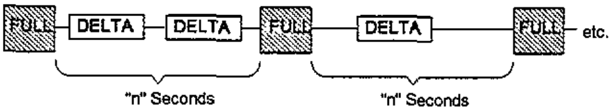
Simplified Emission PDU Diagram

That the PDU is of variable length is not by itself unusual for DIS messages, but in this particular case we find variable length records imbedded within other variable length records. This is not only difficult to comprehend at the design level, but also presents a formidable problem in software implementation.

During the early development of the EMPDU it was assumed that the entire PDU would be sent every time one or more parameters of any emitter and/or beam changed. It has recently been recognized that this would create a significant amount of traffic on the network (Emissions PDUs are among the largest in DIS).. A new set of rules was therefore established. Under these rules it is the responsibility of the host to:

- Issue a "Delta" form of the EMPDU whenever a change occurs to one or more of the emitters on the entity
- Issue a "Full" form of the PDU at a predetermined rate (e.g. every "n" seconds).

The Delta form of the PDU will contain data only for those emitter/beam combinations which have changed, while the Full form of the PDU will contain data for all emitter/beam combinations which are in the "on" state. Pictorially, the Emission PDU stream for a given entity might look like the following:



Typical Issuance of Emissions PDUs

Transmitter and Signal PDUs

These PDUs have been under development since 1991. At that time it was realized that the Emissions subgroup was overtaxed to handle this area and a separate subgroup was formed to independently attack this sphere of DIS simulation. The initial work of the Radio subgroup was based on earlier SIMNET experience and concentrated on the simulation of radios carried by land-based entities. Since that date the subgroup has developed this pair of PDUs (see simplified diagrams of the PDUs below) which seem to handle all of the requirements with the possible exception of data links (recently a separate subgroup of DIS has been formed to concentrate solely on the unique aspects of data links, with the possibility of either defining modifications to the existing PDUs or establishing new ones).

Emitting Entity ID
Radio ID
Radio Type
Transmit State
Antenna Location
Antenna Pattern
Transmitter Parametrics
Crypto System & Key ID
Number of Modulation Params (n)
Modulation Parameter #1
⋮
Modulation Parameter #n

Simplified Transmitter PDU Diagram

Emitting Entity ID
Radio ID
Encoding Scheme
Sample Rate
Length of Data (# of bits)
Number of Samples
Data Byte #1
⋮
Data Byte #n

Simplified Signal PDU Diagram

The concept of operation for the Transmitter and Signal PDUs is rather straightforward - each transmission from a radio is represented by at least two Transmitter PDUs (one for transmitter "on", the second for "off") acting as bookends around a series of Signal PDUs. Additional Transmit PDUs may be issued if any of the parameters of the transmitter change (e.g. antenna orientation) or after a fixed time interval has expired. The number and size of the Signal PDUs vary depending on the type and length of data being sent.

In practice this concept is somewhat more difficult to implement, especially for voice. In transmitting voice the sending host must break the voice stream up into a series of individual Signal PDUs while the receiving host must reassemble them There is a trade-off that must be made between the number of packets, the processing burden placed on both the sender and receiver, speech intelligibility and transport delay. Theoretically the least burden would be to send one very large packet that contains all the speech. This would also result in excellent intelligibility, but would result in unacceptable delay at the receiving end. A compromise must be made in how the voice is packetized - maintaining reasonable delays and intelligibility along with moderate processing burdens. Experimentation with these factors is an ongoing effort in the DIS community.

Acoustic Sensor Simulation

This field is perhaps the most complex field to simulate and has no analogy in SIMNET or DIS 1.0. Acoustic are tightly linked to environment, especially in the ocean milieu. Unlike EM propagation that varies based on only a small subset of environmental conditions , underwater acoustic simulation must evaluate multiple conditions. A reasonably high fidelity acoustic model needs to consider the following environmental effects:

- Sea State
- Water temperature gradient (surface to bottom)
- Water salinity gradient (surface to bottom)
- bottom depth contour
- biological noise sources (snapping shrimp, dolphins, etc.)

In contrast a similar level of fidelity for Electromagnetic propagation needs only to look at rainfall rate and line of sight interference due to terrain. To draw an analogy with EM we would need to consider wind speed and direction at each given altitude, temperature gradients, as well as terrain surface roughness and time of day

The representation of acoustic emissions also requires simulation of such factors as the number of engines, rpm for each, cavitation of props (hull speed vs. prop rpm), and generation of knuckles as ships maneuver. The Emissions subgroup of DIS is only just starting to attack these problems.

Infrared Sensor Simulation

Desert Storm also highlighted one of our fortes in modern battle - our ability to "own the night". SIMNET and DIS Version 1.0 have only a rudimentary ability to show thermal imaging, based purely on simple rules to modify the normal daylight visual scenes produced by the image generators.. True simulation will require simulation of additional factors such as:

- Background thermal levels based on time of day and surface type
- Vehicle thermal level based on # engines, time on, time since off, color/surface type
- The effects of flares and IR jammers

Laser Designation Simulation

A key feature of our success in Desert Storm was the use of "smart weapons". Here again the ability to simulate these encounters in Simnet and DIS 1.0 is lacking. Our success in the Gulf War cannot be taken as predictive of the future.- The next aggressor may possess effective countermeasures. The key is to test our tactics against a more sophisticated adversary in this arena and to develop new tactics (and possibly weapon systems) to overcome such countermeasures

Data Base Issues

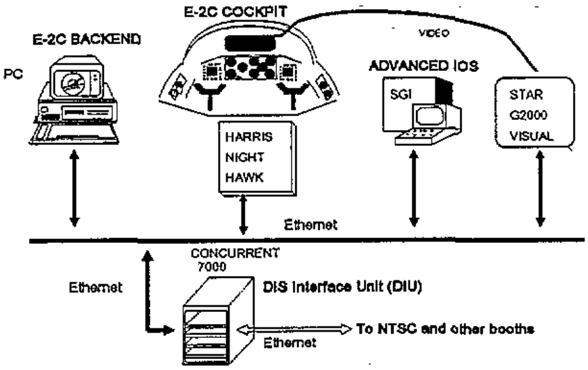
Environmental databases are being worked on by the individual (Air, Land, and Sea) subgroups of DIS, but are in very early stages.. Emitter databases are one of the tasks for the Emissions subgroup. Many exist, but all differ as they were developed for different purposes.. It is the charter of the Emissions Subgroup to define a standard set for DIS, and this work is ongoiing.

RAPID PROTYPING OF BVR PDUS

Over the past year Grumman has been involved in a number of rapid prototyping experiments with the Emissions and Radio PDUs, some via its own internal IRAD and others under contract to Navair/NTSC. The following paragraphs describe several of these activities.

Experimental test of Emitter PDU at 1992 I/ITSEC

From April of 1992 through August of 1992 a series of monthly meetings was held by the University of Central Florida's Institute for Simulation and Training (UCF/IST) to discuss the upcoming DIS demonstration at I/ITSEC. During the July meeting NTSC announced that it's portion of the demonstration would include an emulation of a SLQ-32 Radar Warning Receiver (RWR) and that they were interested in testing the use of the Emissions PDU to drive the RWR. Grumman decided to take advantage of the I/ITSEC opportunity and opened discussions with NTSC to jointly conduct a preliminary test of the PDU per the June 1992 format. A diagram of the Grumman I/ITSEC hardware configuration used during these tests is presented below.



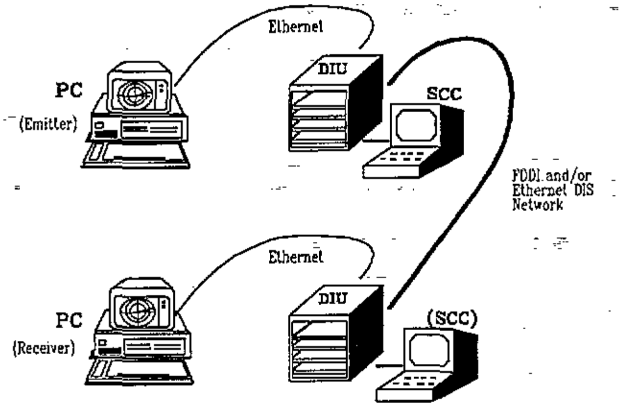
Grumman I/ITSEC DIS Demonstration Hardware Configuration

The format of the I/ITSEC DIS demonstration was rigidly set by UCF/IST and left only the last 3 hours on the last day (Wednesday, November 4) for what was termed "experimental PDU tests". It was during this time that Grumman and NTSC exercised the Emissions PDU for the first time.

The test consisted of Grumman's simulator acting as a hostile radar, issuing Emissions PDUs at a 1 Hz rate. To avoid security problems, the parameters selected for the radar were kept generic, and only the *Mode, Frequency and PRF* fields were evaluated by the NTSC simulation. The results of the test were successful, with the emulated SLQ-32 display correctly tracking the "hostile" E-2C as it moved around the gaming area. Unfortunately the selection of the last time slot at the show left too little time to extend the experiment to other variations of the Emission PDU (e.g. multiple emitters on the same platform, multiple beams per emitter, etc.). It is also interesting to note that the use of the Emission PDUs had no adverse effects on the other simulators on the network at that time.

Laboratory tests of Emitter, Transmit and Signal PDUs

Subsequent to the San Antonio demonstration Grumman extended its investigation of the Emissions PDU to include implementation and test of the September 1992 version of the Emissions PDU. This stage of the effort took place in Grumman's Great River, N.Y. facility, in a simplified version of the full testbed utilized at I/ITSEC. A block diagram of the Grumman DIS testbed hardware configuration is shown below.



Grumman's DIS Testbed

In this demonstration the two PCs act as two separate simulators linked over a DIS network. One PC acts as an entity carrying an emitter, while the other acts as an entity carrying a generic radar warning receiver. The case tested exercised one of the more complex cases supported by the Emissions PDU - that of radar emitters with a rotating beam. The beam rotates throughout a 360^{o} arc, illuminating the receiving entity only once per rotation. The receiving system therefore must perform the following steps to correctly simulate the physics of the radiation:

1. Read the *Emitting Entity ID* field of the Emission PDU to determine which platform carries the emitter
2. Access that platform's latest Entity State PDU (stored locally) and read its *Position, Orientation* and *Time Stamp* fields to determine

the position and orientation of the platform at the time the Entity State PDU was issued

3. Dead Reckon the platform's position and orientation to correspond to the current time
4. Read the Emissions PDU's *Location with Respect to Entity*, *Beam Az Center, Beam Az Sweep, Beam El Center, Beam El Sweep* and *Beam Sweep Sync* fields to determine the position of the beam at the time the PDU was issued.
5. Dead Reckon the beam's position to correspond to current time.
6. Compare own ship position to determine if the receiver is being illuminated at the current time.

These steps must be performed at the local update rate of the receiver's simulation host computer and represent a fairly high computational load, giving further credence to the necessity of using a powerful DIU as the DIS interface at each host.

The results of the laboratory test of the September 1992 version of the Emissions PDU were successful. The receiving "simulator" correctly tracked the "radar beam" of the transmitting "simulator" as it moved throughout the gaming area.

Radio Voice PDU Implementation

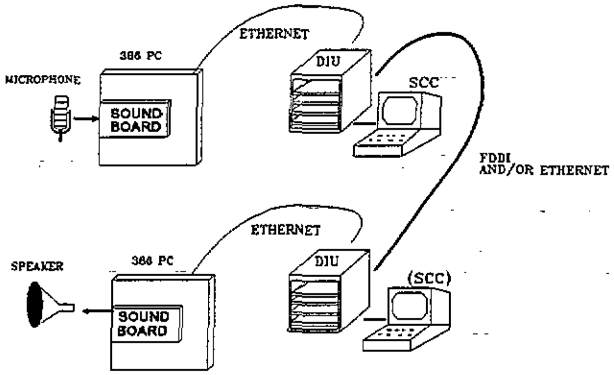
The simulation of voice radio traffic across computer networks is not new to DIS. Within SIMNET a SINCGARS radio simulation was implemented at Fort Knox, Kentucky and Fort Monmouth, New Jersey as early as 1989[2]. The SIMNET voice simulation differed from that required for DIS in one significant area - the method of voice encoding/decoding. It utilized a special board (developed by a SIMNET contractor) called a SIMVAD (SIMNET Voice Analog Digital), built specifically for that application. The board allowed for two types of encoding schemes, Adaptive Pulse-Code with Hybrid Quantization (APCHQ) or Continuously Variable Slope-Delta (CVSD). APCHQ is used for simulation of high quality speech at minimum bit rates, but incurs a high computational load. CVSD is lower quality and less compute-intensive and was found to be best at simulating the SINCGARS radio.

The method of voice encoding/decoding selected for DIS is called *u*-law Pulse Code Modulation (or *u*-law PCM). This technique was developed by Bell Laboratories and is used as the standard for digitizing voice on the Public Switched Telephone Network (PSTN) in all of North America. It is supported by a large number of commercial add-in boards for both PCs and workstations, and is of sufficiently high quality to support all voice applications.

For this investigation a goal was set to evaluate low cost commercial audio hardware to determine its suitability for DIS . After surveying a wide variety of commercial equipment, the hardware selected was the SoundBlaster, manufactured by Creative Labs Inc. This product is a PC compatible add-in board and is one of the most popular on the market. It supports a variety of sampling rates, and interfaces to standard microphones and speakers.

The major drawback to the board (and to all the low cost boards surveyed) is that was not designed for networked real-time operation as required by DIS. Thus our effort in this area concentrated on developing software that would allow the board to operate in a DIS environment.

A pair of boards was acquired and installed in the PCs in Grumman's DIS laboratory testbed, as illustrated below.



DIS Testbed Configuration for Voice PDU Experiments

The software developed provided for the following functions:

1. At the "simulator" originating the voice traffic:
   - PC Reading of microphone on/off switch, and issuing switch transition messages to the DIU across the local Ethernet connection.

- DIU Issuing of DIS Transmit PDUs upon microphone switch on/off transitions.
- PC Real-time memory buffering of *incoming digitized voice originating from* the SoundBlaster.
- PC packetizing of voice into Ethernet frames for transmission from PC to DIU.
- DIU unpacking of Ethernet frames and repacking into Signal PDU for transmission across the DIS network.

2. At the simulator receiving the voice traffic:
    - DIU reading of incoming Transmit PDUs (here additional logic would normally exist to determine if the local radio receiver was set to the correct communications channel, but was omitted within the limited scope of the experiment).
    - DIU unpacking of incoming Signal PDUs from the DIS network and repackaging for transmission to the PC across the local Ethernet connection.
    - PC unpacking of incoming Ethernet voice frames from the local DIU and buffering in memory.
    - PC outputs of voice memory buffers to SoundBlaster for subsequent output to loudspeaker.

The results of the experiment were a limited success in that voice was sent and received per the DIS standard, however the quality of the speech was poor and problems were encountered in processing all but extremely short messages. These shortcomings are disucssed in more detail in the "Lessons Learned" section.

Radio Data Link Implementation

For this implementation the following data links were considered as potential candidates for *implementation:*

- Link-11
- Link-4A
- Link-16 (JTIDS)

Of these Link-11 and Link-4A are carried by all E-2Cs and F-14s in service, while Link-16 is just becoming available on a subset of the aircraft thus far. Of Link-11 and Link-4A the latter was *chosen for implementation since it was felt to be* most representative of the interaction common to the two aircraft. An additional factor in its favor is that unlike Link-11, Link-4A is not encrypted (encryption is one of the controversial areas to be investigated by the new Data Link Subgroup).

The implementation of Link-4A utilized the same *laboratory DIS testbed hardware configuration* shown earlier. In this case the following software was developed to provide the functionality:

1. At the "simulator" originating the data link traffic:
    - PC reading of operator target hook commands and issuing track data messages to the DIU across the local Ethernet connection.
    - DIU unpacking of Ethernet frames at the DIU and repacking into Transmit and Signal PDUs for transmission across the DIS network.
2. At the simulator receiving the data link traffic:
    - DIU reading of incoming Transmit PDUs (here additional logic would normally exist to determine if the local LInk-4A receiver was *the destination specified,* but was omitted within the limited scope of the experiment).
    - DIU unpacking of incoming Signal PDUs from the DIS network and repackaging for transmission to the PC across the local Ethernet connection.
    - PC unpacking of incoming Ethernet frames from the local DIU and buffering in memory.
    - PC reformatting of target position data and output to the simulated radar display.

The resulting demonstration confirmed that Link-4A target data can be transmitted across a DIS link using the current form of the Transmit and Signal PDUs. The format followed in the demonstration was as follows:

1. At the transmitting "simulator" (PC) the radar operator hooks a target on the simulated radar display and selects the datalink message icon at the top of the screen, causing the "simulator" to send a Link-4A message to the receiving "simulator".
2. At the receiving "simulator" the radar scope display reflects the target the transmitting "simulator" has on its radar display. The operator can then hook that target and view its parameters.

LESSONS LEARNED

The implementation of the Emitter, Transmit, and Signal PDUs was not as straightforward as one *might expect upon reading the DIS standard.* For the Emitter PDU one of the major problems encountered was that ofimplementing the requirement to handle multiple targets within multiple beams within multiple emitters all in a single PDU. In our implementation for this project we went to the simplest solution by sizing our buffers to a reasonable maximum size. This is fine for entities with few (e.g. 5 or less) emitters, but future DIS exercises may require more elegant solutions (e.g. the use of multiple pointers to *individual arrays in memory) to avoid tying up* huge amounts of memory simply to handle a potential worst case.

For the Transmit PDU the main problem encountered was when one or more of these PDUs were lost due to simulator overload. Since these PDUs are sent in broadcast mode there is no guarantee of their arrival at each potential recipient. We found, for example, that during heavy voice traffic our PC "simulator" would sometimes lose PDUs. This is annoying if the PDU lost is a voice Signal PDU causing clipped speech, but is disastrous when a Transmit PDU was missed since it is used to determine whether to process the Signal PDUs. Entire portions of speech can be lost in this manner. In the lab our solution was to experiment with adding artificial delays between the DIU and the PC, a rudimentary form of flow control. The ideal solution is total flow control across the network, but that is apparently in conflict with the nature of DIS.

The voice version of the Signal PDU implementation was probably the most challenging. Our goal was to explore the potential of implementing DIS voice via mass produced audio hardware designed for the PCs market. The Soundblaster board we choose did not conform completely to the proposed DIS standard, nor was it designed with such an application in mind, but we felt its low cost and board support in the PC environment would offset that drawback. As stated earlier the results of the experiment are mixed. It proved that acceptable speech can be sent using the DIS PDUs with a reasonably short delay as long as the speaker keeps his messages very short (e.g. less than 2 seconds in length) and relatively infrequent. Long sentences caused the receiving PCs buffers to overflow, often causing an exit from the DIS session. We found a cure for this by introducing a delay at the receiving DIU-to-PC interface, however this caused a dramatic degradation of speech intelligibility. We believe that the use of more powerful PCs, such as 66 MHz 486s, would significantly improve the situation. The increase in computational speed, perhaps combined with *play-out protocols* to smooth the jitter between sucessive voice packets[3,4]. would undoubtedly help, but we were unable to test these theories under the scope of this effort.

Another problem we observed affects both the *voice and datalink implementations.* The use of Ethernet 802.3 technology combined with the current format of the Signal PDU means that there is no guarantee that the received voice packets are in the same order that they were sent. This can be overcome by the use of a token-ring version of Ethernet or of FDDI, but these are considered too restrictive for DIS. A potential cure is to add a Signal PDU packet number to the existing set of fields and to locally reorder the packets based on this numbering scheme. Unfortunately the limited scope of our effort did not allow for experimentation with this concept to test its feasibility.

SUGGESTIONS FOR FUTURE BVR PDU AND DATA BASE DEVELOPMENT

One of the primary observations that can be made regarding the BVR PDUs is that they are fairly complex to implement. The current (2.03) version of the DIS standard provides only a top-level view of how to use them. Such PDUs require a separate "Users Guide" to aide the developers. The Emissions subgroup has started such a task but it will require a significant amount of work and time before it is sufficient for the DIS community.

Another observation is that these PDUs will become a major source of traffic on DIS networks. The use of delta vs. full PDU versions as described earlier may be required in all cases.

In the area of databases there is still much work to be accomplished. It appears that the establishment of "strawman" databases, at a declassified level, is the logical first step, but this is currently viewed as a low priority effort by the

DoD sponsors of DIS. The DoD customers must institute priorities in this area or it will continue to remain a background issue.

REFERENCES

1. Denver, J., "ECM/ECCM Implementation in the DIS Radar Environment", presented at the Fifth Workshop on Standards for the Interoperability of Defense Simulations, September 1991.

2. Gonzalez, J. "Electromagnetic Propagation Modeling for Distributed Simulation", Proceedings of the 13th Inter-service/Industry Training Systems Conference, November 1991.

3. Cohen, D., "Issues in Transnet Packetized Voice Communication", presented at the Fifth Symposium on Data Communications, 1977.

4. Gehl, T., "Packetized Voice for Simulated Command, Control, and Communications", Proceedings of the 13th Interservice/Industry Training Systems Conference, November 1991.