1-1-1994

Standard For Distributed Interactive Simulation Application Protocols: Version 3.0 Working Draft

University of Central Florida Institute for Simulation and Training

INSTITUTE FOR SIMULATION AND TRAINING

MARCH 7, 1994

STANDARD FOR DISTRIBUTED INTERACTIVE SIMULATION APPLICATION PROTOCOLS

VERSION 3.0
WORKING DRAFT

IST DOCUMENTATION

IST-CR-94-18

IST

INSTITUTE FOR SIMULATION AND TRAINING

Contract Number N61339-91-C-0091
STRICOM
DMSO
March 7, 1994

Standard for Distributed Interactive Simulation-- Application Protocols

Version 3.0 Working Draft

Institute for Simulation and Training
3280 Progress Drive
Orlando FL 32826

University of Central Florida
Division of Sponsored Research

IST-CR-94-18

DRAFT STANDARD
7 Mar 1994

STANDARD FOR INTERACTIVE SIMULATION

PROTOCOLS FOR

DISTRIBUTED INTERACTIVE SIMULATION APPLICATIONS

VERSION 3.0

(WORKING DRAFT)

"NOTE: This draft, dated 7 Mar 1994, prepared by the Institute for Simulation and Training for STRICOM, has not been approved and is subject to modification. DO NOT USE PRIOR TO APPROVAL."

Foreward

The following collection of protocols is intended to be an addition to the Standard for Distributed Interactive Simulation--Application Protocols verion 2.0 Fourth Draft.

The contents of this draft are incomplete and subject to change reflecting its working status.

C O N T E N T

4.4.9 Aggregate Protocol for DIS. The following paragraphs shall establish the content of the Aggregate Protocol PDUs and the procedure for using the Aggregate Protocol in a DIS exercise.

4.4.9.1 Aggregate Approach. The Aggregate Protocol provides a mechanism for grouping multiple entities and communicating information about these groups of entities. It also provides a mechanism for participants in a DIS exercise to request that the entities that make up an aggregate join or leave the DIS exercise.

4.4.9.1.1 Scope. This standard provides a mechanism for SAFORs and constructive wargames to interface to DIS exercises by describing aggregates. The coordination of the simulation of constituent entities, the assignment of de-aggregated entities to simulation applications is outside the scope of this standard.

4.4.9.1.2 Requirement. The Aggregate Protocol is an optional part of the DIS Protocol. Participants in a DIS exercise do not have to process Aggregate Protocol PDUs beyond what is necessary to ignore the Aggregate Protocol PDUs.

4.4.9.1.3 Origination. The Aggregate Protocol PDUs may be transmitted and received by a Simulation Management application other than the application simulating its constituent entities. The Simulation Management application that issues and receives Aggregate PDUs is called the Aggregate Controller. Exercises may have multiple Aggregate Controllers. The Aggregate Controller may be at a different Site than its constituent entities.

4.4.9.1.4 Aggregate Controller. The Aggregate Controller is responsible for issuing and receiving Aggregate Protocol PDUs.

4.4.9.1.5 Timing. The Aggregate PDUs are issued every 5 seconds, or at the default interval that Entity State PDUs are issued.

4.4.9.2 Aggregate Protocol PDU Header. All Aggregate Protocol PDUs contain the standard PDU header (see 4.4.1 of document IST-CR-93-40). The Aggregate Protocol defines three PDU types and also uses the Acknowledge PDU of the Simulation Management Protocol. The Four PDUs in the Aggregate Protocol are:

(1) Aggregate Descriptor PDU
(2) De-Aggregate Request PDU
(3) Re-Aggregate Request PDU
(4) Acknowledge PDU (see 4.4.6.4.5 of document IST-CR-93-40)

4.4.9.3 Aggregate Definition. An aggregate is a group of entities. Entities may be grouped by type, capabilities, mission, command hierarchy, or arbitrarily. Aggregate may be formed for any reason. The purpose of an aggregation is to save bandwidth by sending fewer PDUs for the Aggregate than for individually represented units, or to save computational effort by providing

information on the group of entities that can be used by a receiver to sort Entity State PDUs, or to put hierarchical information about military or functional units or groups on the DIS network.

4.4.9.3.1 Issuance of the aggregate protocol PDUs. If the entities that make up an Aggregate are in compliance with the DIS standard, they will always issue Entity State PDUs. In practice however, many uses of the Aggregate Protocol, such as interfaces to SAFORs or Constructive Wargames, will only issue the Entity State PDUs for the subset of the Aggregates that are currently participating in DIS exercises.

4.4.9.3.2 Joining a DIS exercise. The process of joining a DIS exercise via the Aggregate Protocol is called de-aggregation. It implies that the constituent entities of an aggregate were not issuing or receiving DIS PDUs and after de-aggregation are fully compliant with the DIS protocols. The aggregate descriptor PDUs are issued before and after de-aggregation.

4.4.9.3.3 Leaving a DIS exercise. The process of leaving a DIS exercise via the Aggregate Protocol is called re-aggregation. It implies that the constituent entities of an aggregate were issuing or receiving DIS PDUs and after re-aggregation will not issue DIS PDUs. The aggregate descriptor PDUs are issued before and after re-aggregation.

4.4.9.4 Aggregation and De-Aggregation Protocol. Any entity can request to an Aggregate Controller that it de-aggregate by sending a De-Aggregate Request PDU. The Aggregate Controller receives the request and returns a De-Aggregate Acknowledge PDU. The constituent entities of the Aggregate then issue their Entity State PDUs. In the case of hierarchical aggregations, (ie. division, regiment, battalion etc.) a De-Aggregation Request must go to each subordinate aggregation desired to be de-aggregated. A De-Aggregation Request to the highest aggregation would yield, for example only the headquarters staff or units belonging to the division.

4.4.9.4.1 Aggregate Controllers can request to re-aggregate. The aggregate sends out a warning to all other participants that it is going to re-aggregate by issuing a Re-Aggregate Request PDU. Any entity can stop this process by sending a De-Aggregate Request PDU to the Aggregate Controller.

4.4.9.4.2 Issuance of De-Aggregate Request. In order to prevent "flooding" of the network by an ill-timed Re-Aggregate request, entities should send out De-Aggregate Requests at a random time 0 to 5 seconds after receipt of an undesired Re-Aggregate Request PDU. Entities may listen for other entities sending out De-Aggregate requests in order not to duplicate the request.

4.4.9.4.3 Receipt of De-Aggregate Request. Upon receipt of the De-Aggregate Request the receiving Aggregate Controller

communicates to its component entities that they should now issue Entity State PDUs. The required mechanism for this is outside the scope of this standard. The Simulation Management Protocol (4.4.6) may be used for this role.

4.4.9.4.4 Detection Range. The Detection range of the sending entity can be used by the Aggregate to help determine when it can re-aggregate. If a detection range value received by an Aggregate Controller is greater than the default detection range of the Aggregate Controller's re-aggregate function, then the new detection range can be used for re-aggregation calculations.

The responsibility for de-aggregating when an aggregate comes within interaction range of DIS entities always rests with the aggregate controller. The detection range is provided only as an aid. The de-aggregation protocol is not meant imply that DIS entities are responsible for "cueing" aggregate controllers when to de-aggregate, since the entire aggregate protocol is optional.

4.4.9.4.5 De-Aggregation Time. The De-Aggregation time is the amount of time, in seconds, that the requestor would like the aggregate to be de-aggregated. A value of zero (0) means follow default protocol for de-aggregating. A value of -1 means de-aggregate long enough to send constituent Entity State PDU once.

Applications that wish to see the contents of an aggregate once may also be able to use the Simulation Management protocol Data Query and Data PDUs.

4.4.9.5 Description of the De-Aggregation Process. This will describe the process of taking an aggregated unit from an aggregated state to a de-aggregated state. There are multiple paths through the protocol from an aggregated state to a de-aggregated state.

4.4.9.5.1 The unit starts aggregated. It may chose at any time to become de-aggregated. This can be accomplished without any interaction with other entities or management entities on the network.

4.4.9.5.2 The unit starts aggregated. The Aggregate Controller receives a de-aggregate request. It responds with an acknowledge PDU with the acknowledge flag set to "will comply". It then enters a de-aggregated state.

4.4.9.5.3 The unit starts aggregates. The Aggregate Controller receives a de-aggregate request. It responds with an acknowledge PDU with the acknowledge flag set to "will not comply". It remains aggregated.

4.4.9.6 Description of the Re-Aggregation Process. This will describe the process of taking an aggregated unit from a de-aggregated state to the aggregated state. There is one path

from the De-Aggregated State to the Aggregated state but, it can be interrupted.

4.4.9.6.1 The unit starts de-aggregated. The Aggregate Controller starts the re-aggregation process by issuing a Re-Aggregate Request PDU. It is now waiting to re-aggregate.

4.4.9.6.2 Aggregate Timing. After ten seconds in the Waiting to Re-aggregate State the unit enters the Aggregated State unless it receives and responds positively to a De-Aggregate Request.

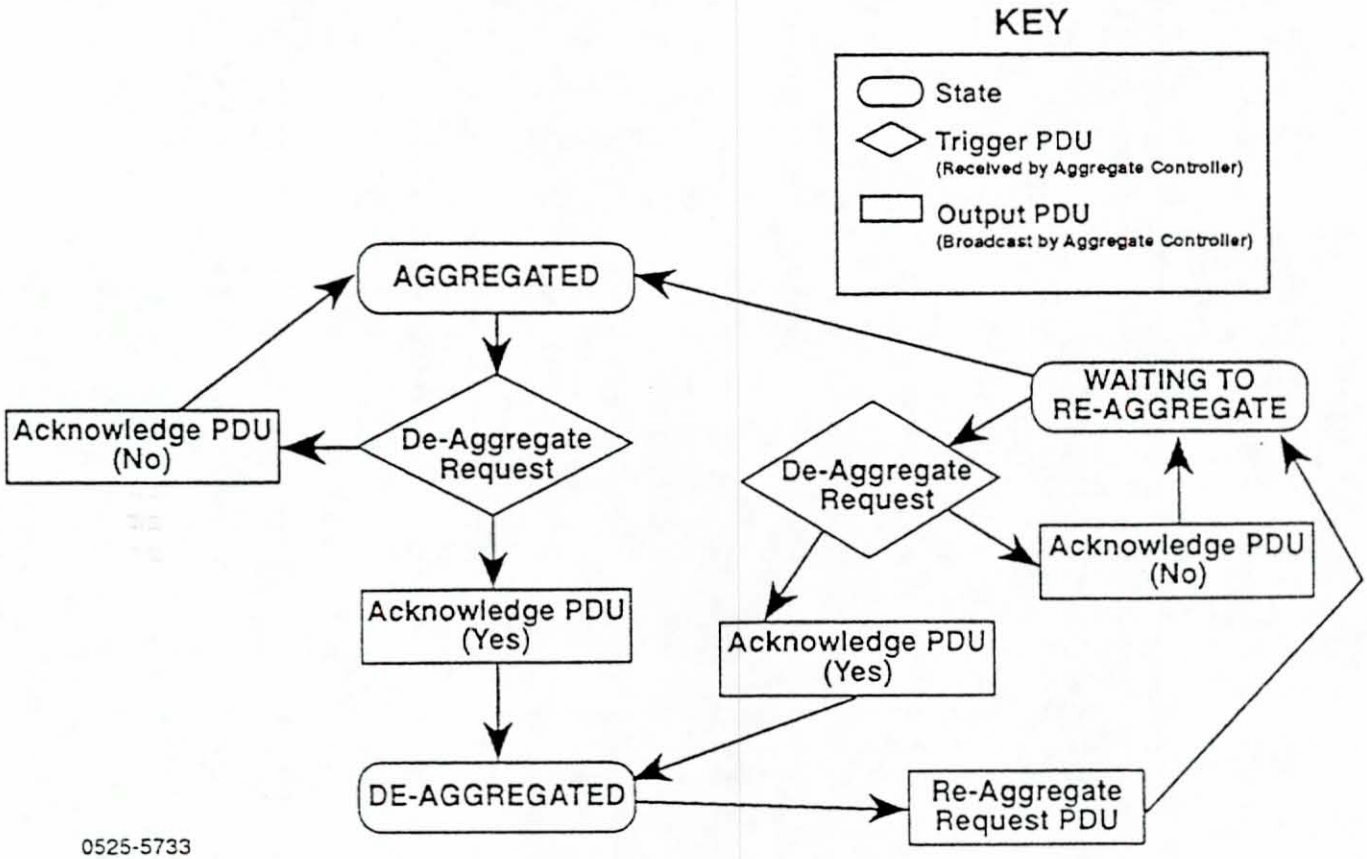

Figure 4 - 15

Aggregate Protocol State Diagram

4.4.10 Distributed Underwater Acoustic Emission Regeneration for DIS. The following paragraphs shall establish the content and procedure for the use of the Distributed Underwater Acoustic Emission Regeneration in a DIS exercise.

4.4.10.1 Distributed Underwater Acoustic Emission Regeneration Approach. The following assumptions shall be followed in DIS:

(1) Entities with underwater acoustic emitters(s) shall simulate their emitters(s) and shall output predefined, real-time operational parameters via the DIS communications network.

(2) Entities with receivers that can sense underwater acoustic emissions shall determine the received signal(s) to the fidelity level required by that particular receiving simulation application. This determination shall be accomplished by using the operational parameters provided in the Underwater Acoustic PDU along with information from stored databases which describe the emitter, the environments and its boundaries.

(3) Entities that emit intentionally shall determine if any acoustic echoes are received to the fidelity level required by that transmitting and receiving simulation application.

4.4.10.2 Underwater Acoustic PDU. The Underwater Acoustic (UA) PDU shall be used to communicate acoustic emissions, including active emissions and countermeasures, propulsion emissions, and other entity events that result in generation of underwater acoustic emissions.

4.4.10.2.1 Information Contained in the UA PDU. The UA PDU shall contain the following information:

(1) Standard PDU Header information.

(2) Identification of the emitting entity.

(3) Identification of the event.

(4) Identification of the type of update information in PDU.

A state update provides a full description of the emission systems(s) identified in the PDU. Only acoustically generating (UA emitting) systems and beams with acoustic energy will be included in the state update.

A change update provides a method to allow changes of UA emitter state to communicated between state updates.

(5) Number of UA emitter systems for which information is being provided in the PDU.

(6) Information for one or more UA emissions systems that the entity has. This information includes:

(a) Length of the system data.

(b) Number of beams (for each system) for which information is being provided in the PDU.

(c) Acoustic System which includes the acoustic emitter name, function of the system and acoustic ID number.

(d) Location of the UA emitter which is the location of the acoustic source.

(e) Information for one or more acoustic beams that the system has. This information includes:

(i) Length of the beam.

(ii) Beam ID # for each beam.

(iii) Operating mode of the acoustic source included off.

(iv) Shaft speed in revolutions per minute (RPM).

(v) Any acoustic emission masker status (on or off).

(vi) An additional Source Activity field.

(vii) UA fundamental parametric data which is essentially data that can vary for a specific system or can vary dynamically during system operation (even though this system's mode and beam functions are not changed). This data is also available to support applications of low-fidelity simulations which may not have the computational power to process high-fidelity regeneration models.

4.4.10.2.2 Issuance of the UA PDU. The UA PDU shall be issued in the following instances:

(1) Changed data update:

(a) Operational parameters contained in the UA PDU fields change.

(b) Main Beam Descriptors exceed specified threshold. The threshold may be established at exercise start or during an exercise. The mechanism to set the threshold is outside the scope of this standard.

(c) Location with respect to entity (e.g. a towed active emitter's) changes by TBD meters.

(d) The number of beams, operating Mode, or masker status changes.

(e) Shaft RAM changes by TBD.

(2) State update: a predetermined length of time has elapsed since issuing the last UA PDU. This value may be established at exercise start or during the exercise. The mechanism by which this value is established is outside the scope of the protocol. If no value is established, the default value shall be five seconds. A default tolerance of ±10% shall apply to this value. The tolerance value may be set by a mechanism outside the scope of this protocol.

4.4.10.2.3 Receipt of the UA PDU. Upon receipt of a UA PDU, the receiving entity shall determine if the emission is detectable and use the information in the UA PDU to appropriately influence emission detection equipment in the simulation.

4.4.11 Destructible Entity Protocol for DIS. Information associated with dynamic changes of entities within the simulation environment shall be communicated through the use of several PDUs. These are the:

(a) Creation / Modification PDU

(b) Deletion PDU

(c) Request ID PDU

(d) Reply ID

(e) Request Object

(f) Reply Object

(g) Areal Feature PDU

4.4.11.1 Creation / Modification PDU. This PDU shall be used to communicate the addition or change(s) of a destructible entity in a simulated environment

Creation - The instance of a new object in the simulated environment

Modification - the change in appearance of a Creation object in the simulation environment

4.4.11.1.1 Information contained in the Creation / Modification PDU. The PDU shall contain the following information:

(a) A sequential number of the Destructible Entity PDU's which is incremented each time a Destructible Entity PDU is issued and contains the host and exercise ID

(b) An identification of the object to be created or modified

(c) The world coordinates of the object location

(d) The orientation of the object

(e) The height of the object (Minimum and Maximum)

(f) The length of the object (Minimum and Maximum)

(g) The width of the object (Minimum and Maximum)

(h) The object material type

(i) The appearance of the object

4.4.11.1.2 Issuance of the Creation / Modification PDU. This PDU shall be issued when the simulation exercise requires the creation or modification of destructible entities.

The Creation PDU is issued when a new destructible object is required in the simulation exercise. The PDU is issued by the creating host. A network responsible host collects and maintains management information about Destructible PDU's on the local network.

The Modification PDU is issued by a simulation entity when it performs some action that results in an appearance change to an existing destructible entity.

4.4.11.1.3 Receipt of the Creation/Modification PDU. Upon receipt of this PDU the receiving host computer will take appropriate action based on the PDU type.

If the PDU type is a Creation the receiving host computer will create the object based upon the Object Type ID number. It will place the object at the world coordinate location with the correct orientation. The height and length as well as the material composition are provided to support the creation of the object. The receiving host computer will store the information of the creation in a table. The minimum information stored in a table shall be the sequence number, PDU header, and Object Type, and Timestamp.

If the PDU type is a Modification the receiving host computer will replace the identified object with a modified version based upon the appearance number as specified in Appendix D. 30.5 of DIS standard. The receiving host computer will store the information of the modification in a table. The minimum information stored in a table shall be the Sequence Number, PDU Header, Object Type, and Timestamp.

4.4.11.2 Deletion PDU. This PDU is used to communicate the deletion of an object in the simulated environment.

Deletion - the removal of a destructible object from the simulation environment. This includes those created by the Areal Feature PDU.

4.4.11.2.1 Information contained in the Deletion PDU. The PDU will contain the following information.

A sequential numbering of the Destructible Entity PDU's which is incremented each time a Destructible Entity PDU is issued.

(a) An identification the object to be deleted.

(b) The world coordinates of the object location.

4.4.11.2.2 Issuance of the Deletion PDU. This PDU is issued when the simulation exercise requires the deletion of destructible entities. The deletion PDU is issued by a simulation entity when it performs some action that results in the removal of a destructible entity.

4.4.11.2.3 Receipt of the Creation\modification/Deletion PDU. Upon receipt of this PDU the receiving host computer will remove the identified entity from the host simulation environment. The receiving host computer will store the information of the deletion in a table. The minimum information stored in table shall be the sequence number, PDU header, and time stamp.

4.4.11.3 Request ID. The Request ID PDU is an exercise maintenance tool that permits simulation host computers to query the current status of destructible object(s). This PDU permits simulations to Request the status of dynamic state(s) in an exercise.

4.4.11.3.1 Information contained in the Request ID PDU. The Request PDU shall contain a matrix of maximum sequence numbers, by issuing host, of the last destructible entity received.

4.4.11.3.2 Issuance of the Request ID PDU. The Request ID PDU shall be issued by any entity that is returning to the exercise after an absence, or is conducting a periodic checking of the simulation environment to ensure the correctness of its simulation environment. The Request ID shall transmit a matrix of its maximum sequence number by issuing host of the last received destructible entity.

4.4.11.3.3 Receipt of the Request ID PDU. Upon receipt of the Request ID PDU a responsible host computer shall send a Reply with its matrix of maximum destructible entity sequence numbers by issuing host.

4.4.11.4 Reply ID. The Reply ID PDU is an exercise maintenance tool that permits simulation responsible host computers to reply on the current status of dynamic state(s) in an exercise.

4.4.11.4.1 Information contained in the Reply ID PDU. The Reply ID PDU shall contain the sending responsible host \325s matrix of maximum destructible entity sequence numbers by host.

4.4.11.4.2 Issuance of the Reply ID PDU. The Reply ID PDU shall be issued by a responsible host which is responding to a Request ID with information about maximum sequence number for each host issuing destructible objects.

4.4.11.4.3 Receipt of the Reply ID PDU. Upon receipt of the Reply ID the receiving host computer shall compare the matrix of maximum sequence numbers by host to its table information to determine the destructible objects that are missing and required.

4.4.11.5 Request Object. The Request Object PDU is a simulation exercise maintenance tool that permits simulation host computers to request needed destructible objects. This PDU permits simulators to Request an Object\050s\051 in an exercise. This will allow additional simulators to be added to an exercise at any simulation time or allow for the recovery from host computer or communication failures by providing a method to update exercise information on destructible entity status.

4.4.11.5.1 Information contained in the Request Object PDU. The PDU will contain the following information:

(a) Sequence Minimum and Maximum Numbers

(b) Time Minimum and Maximum Values

(c)Geographic Bounds Minimum and Maximum

4.4.11.5.2 Issuance of the Request Object PDU. The Request Object PDU shall be issued by any host that is being added to the exercise after the simulation has started or is returning to the exercise after an absence or has determined through a Request PDU that destructible entity(ies) are needed.

4.4.11.5.3 Receipt of the Request Object PDU. Upon receipt of the Request Object PDU the responsible host computer shall send a Reply Object PDU with the requested matrix of destructible entity(ies) to the requesting host computer. This is done by selecting all of the destructible entity(ies) that match an "And" all of the conditions in the Request Object PDU.

4.4.11.6 Reply Object PDU. The Reply Object PDU is a exercise maintenance tool that permits simulation responsible host computers to send the requested destructible or allow for the recovery from host computer or communication failures by providing a method to update exercise information on destructible entity status.

4.4.11.6.1 Information contained in the Reply Object PDU. The Reply Object PDU shall contain the requested destructible entity PDU's; Creation, Modification, Deletion.

4.4.11.6.2 Issuance of the Reply Object PDU. The Reply Object PDU shall be issued by a responsible host entity which is transmitting update information entity status.

4.4.11.6.3 Receipt of the Reply Object PDU. upon receipt of the Reply Object PDU the receiving host computer shall review the destructible entity(ies) to determine which require processing to update the current destructible status, update the host maximum sequence number, and process the appropriate PDU(s)

4.4.11.7 Areal Feature PDU. This PDU is used to cerate or modify an areal feature in a simulation environment.

(a) Creation - The instance of a new feature in the simulation environment.

(b) Modification - The change in appearance of an object in the simulation environment.

4.4.11.7.1 Information contained in the Areal feature PDU. The PDU will contain the following information:

A sequential numbering of the Destructible Entity PDU's which is incremented each time a Destructible Entity PDU is issued and contains the host and exercise ID.

(a) An identification the feature to be created or modified.

(b) The World Coordinates of the feature location.

(c) The orientation of the feature.

(d) The number of segments and location of the segments

(e) The number of and offset for the cross-sections located at each of the segment points

(f) The material type of the feature

(g) The appearance of the feature

4.4.11.7.2 Issuance of the Areal Feature. This PDU is issued when the simulation exercise requires the creation or modification of dynamic areal feature.

The creation PDU is issued when a new areal feature is required in the simulation exercise. The PDU is issued by the creating host. A network responsible host collects and maintains management information about Destructible PDU's on the Local network.

The Modification PDU is issued by a simulation entity when it performs some action that results in an appearance change to an existing areal feature.

4.4.11.7.3 Receipt of the Areal Feature PDU. Upon receipt of this PDU the receiving host computer will take appropriate action based on the PDU type.

If the PDU type is a Creation the receiving host computer will create the object based upon the Object Type ID number. It will place the object at the world coordinate location with the

correct orientation. The number of segments is provided as a counter for the iteration of a matrix for each of the segments. The receiving host computer will store the information of the creation. As a minimum the information stored should contain the PDU header, Areal, Feature type, and timestamp.

If the PDU type is a Modification the receiving host computer will identified with a modified version based upon the appearance number as specified in 30.5.1 of the DIS Standard. The receiving host computer will store the information of the modification. As a minimum the information stored should contain the PDU Header, Areal Feature Type, and Timestamp.

4.4.12 Dynamic Environment Protocol for DIS. The following paragraphs define the protocol for the simulation of dynamic environmental events in DIS. The Dynamic Environment Protocol (DEP) supports the simulation of atmospheric and oceanographic conditions that change during the life of a distributed simulation session.

4.4.12.1 Dynamic Environmental Conditions Simulation Approach. The following assumptions shall be followed for the DEP in DIS:

(1) The application responsible for a particular environmental condition shall output Environmental PDUs to represent the current state of the environmental condition.

(2) Receiving applications which are affected by the environmental conditions shall use the parametric physical description data in the Environmental PDU to drive the appropriate effects models within their simulations.

(3) A DIS entity shall be associated with every environmental entity.

(4) The broadcast of Environmental PDUs will be at: entity instantiation, entity termination, or periodically. The period between updates of the Environmental PDU will be at a considerably lower update rate (1/30 Hz). The low update rate is due to the slowly changing nature of the environmental conditions that will be handled by the Environmental PDU.

4.4.12.2 Issuance of the Environmental PDU. The environmental PDU shall be issued by the simulation application responsible for the environmental entity when one or more of the following conditions exist:

(1) A change in the environmental entity's state occurs. Parameter changes requiring a re-issuance include: Size, Shape, and Intensity of the entity.

(2) A predetermined length of time has elapsed since the issuing of the last Environmental PDU. This value may be established at session start or during execution of the session. The mechanism by which this value is established is outside the scope of the protocol. If no value is established, the default value shall be 30 seconds. A default tolerance of 10% shall apply to this value. The tolerance value may be set by a mechanism outside the scope of this protocol.

The Environmental PDU shall be issued using a real-time best effort, multicast communications service. The PDU shall be sent to all simulation applications participating in the same session.

4.4.12.3 Receipt of the Environmental PDU. Upon receipt of the Environmental PDU, a simulation application shall determine that the PDU contains newer information than that currently being used to model the received entity. If so, the simulation application shall use the information contained therein to model the entity described in the PDU. Otherwise, the PDU shall be discarded. If a predetermined length of time has elapsed since the receipt of he last Environmental PDU representing that given entity, the simulation application shall delete that entity instance from the simulation. The length of time shall be established at session start, or may be changed during the session. If no value is established the default value shall be 2.4 times he environmental PDU update interval described in 4.4.12.2(2). If the intensity parameter of the Environmental PDU is set to zero, the instance of the entity shall be deleted from the application.

4.4.12.4 Physical Parameters. This variable length field will contain the parametric description of the environmental entity. The parameters that shall be found in this field include: Size, Shape, and Intensity.

4.4.12.4.1 Size. This field shall contain the dimensions of the environmental entity in meters referenced to the entity coordinate system. Each of the X, Y, Z components shall be represented by 32-bit floating point numbers. Each of these components shall dilate the volume defined in the Shape field. A two dimensional entity (e.g. sea state) would be represented by having one of its dimensions set to zero.

4.4.12.4.2 Shape. This field shall contain an 8 bit enumerated value pointing to a "unit" shape defined in the DIS Enumeration Document or a session specific enumeration library as appropriate. These shapes shall be defined with three orthogonal axes each with a length of 0 or 1. The final volume (or area in the case of a 2-dimensional entity) shall be the unit shape as dilated by the size parameters contained in the Environmental PDU.

4.4.12.4.3 Intensity. This field shall contain an 8 bit unsigned integer value which shall contain the strength or intensity of the environmental entity. The contents of this field shall be used by the environmental effects models in each simulation application. The allowable units represented by this parameter shall be as defined by the enumeration document or in the session specific library. A value of zero (0) shall be used in this field to terminate the entity and shall be the value in the last Environmental PDU broadcast.

4.4.13 Entity Handover Protocol for DIS. The following paragraphs define the protocol that hall be use to pass the control of simulation entities from one simulation application to another. The Entity Handover Protocol (EHP) supports the dynamic allocation and control of simulation entities between simulation applications in a distributed simulation session. The EHP will use 3 PDUs: the Transfer Control Request PDU, Transfer Control PDU, and the Transfer Control Acknowledge PDU.

It is necessary for the handover of simulation entities from one simulation application to another to be as reliable as possible. For reliability, the use of a Request-Transfer-Acknowledge process is used in the EHP. Each step of the process requires the use of a timer. The value of the timer shall be established at session start and may be changed during the session. The mechanism used to set and change the timer values is outside the scope of this protocol. If no values are established, the default values for the handover timers shall be as follows:

(1) Request response timer - 5 seconds

(2) Transfer acknowledge timer - 1 second.

4.4.13.1 Procedure for Entity Handover. The transfer of entities from one simulation application to another in DIS shall be accomplished through a series of request and response messages between two simulation applications. The request for handover shall be initiated by the application that desires control over an entity that is under the control of another application. Handover of control of the entity does not occur until the acknowledgement of the transfer is complete.

4.4.13.2 State Information for Entity Handover. The following paragraphs describe the different states and transitions for entity handover.

4.4.13.2.1 Requesting Application. The requesting application may be in one of three states:

(1) Ready State - A requesting application is in the ready state when it is not in either the requesting state or receiving state.

(2) Requesting State - A requesting application is in the requesting state when it has a transfer control request pending and has not received a transfer control in response.

(3) Received State - A requesting application is in the received state when it has received the entity data but has not acknowledged receipt of the entity.

The behavior of the requesting entity during entity handover is shown in figure 4-.

Transition	Condition and Actions
Request Control	When the conditions for entity handover are met, the requesting application shall issue a Request Entity Control PDU (see X.X.X.X). The application shall proceed from the Ready state to the Requesting state and the Request Response Timer shall be set.
Transfer Control	When the requesting application has received the Transfer Control PDU and has not yet issued the Acknowledge Control Transfer PDU.
Repeat Request	When the Acknowledge control transfer timer expires, the Request Entity Control PDU shaw be reissued and the control transfer timer shall be reset.

4.4.13.2.2 Providing Application. The providing application may be in one of two states:

(1) Ready State - A providing application is in the ready state when it is not in either the control transfer state.

(2) Control Transfer State - A requesting application is in the control transfer state when it has received a transfer control request, is transferring control and is awaiting an acknowledgement of control assumption.

The behavior of the providing application is shown in figure X.X.

Transition	Condition and Actions
Offer Control	When a Transfer Control Request PDU is received, a Transfer Control PDU shall be issued, the transfer acknowledge timer shall be set and the application shall proceed from the ready state to the control transfer state.
Transfer Control	When a Transfer Control Acknowledge PDU is received the application shall cease to control the entity, stop issuing Entity State PDUs for the entity, continue to

dead reckon the entity, and shall proceed to the Ready State.

Transfer Canceled — When a Transfer Control PDU has been issued and the transfer acknowledge timer has expired without receipt of a Transfer Control Acknowledge PDU the application shall proceed from the control transfer state to the ready state, and the application in control of the entity shall not change.

4.4.13.3 Issuance of the Transfer Control Request PDU. The Transfer Control Request PDU shall be issued by an application when control of another application's entity is desired. The application shall proceed from the Ready state to the Requesting state and the Request Response Timer shall be set.

The Transfer Control Request PDU shall be issued using a best effort multicast communication service.

4.4.13.4 Receipt of the Transfer Control Request PDU. Upon Receipt of the Transfer Control Request PDU, the receiving application shall:

(a) Issue a Transfer Control PDU

(b) Set the Transfer Acknowledge timer

(c) Proceed from the ready state to the control transfer state.

4.4.13.5 Issuance of the Transfer Control PDU. The Transfer Control PDU shall be issued by an application upon receipt of a Transfer Control Request PDU and when control of the entity by the requesting application is desired. The application shall proceed from the Ready state to the Transfer Control state and the Transfer Acknowledge Timer shall be set.

The Transfer Control PDU shall be issued using a best effort multicast communication service.

4.4.13.6 Receipt of the Transfer Control PDU. Upon Receipt of the Transfer Control PDU, the receiving application shall:

(a) Issue a Transfer Control Acknowledge PDU

(b) Proceed to the ready state

(c) Issue an Entity State PDU for the entity now under its control.

4.4.13.7 Issuance of the Transfer Control Acknowledge PDU. The Transfer Control Acknowledge PDU shall be issued by an

application upon successful receipt of the Transfer Control PDU and only when transfer of entity control is desired by the controlling application. The application shall proceed from the

(a) Received state to the Ready State

(b) Delete the original entity from the application

(c) Create a new entity using the new entity ID with the original entity's state (as modified by the time transpired since the receipt of the Transfer Control PDU).

The Transfer Control Acknowledge PDU shall be issued using a best effort multicast communication service.

4.4.13.8 Receipt of the Transfer Control Acknowledge PDU. Upon Receipt of the Transfer Control Acknowledge PDU, all receiving applications shall:

(a) Delete the instance of the entity with the original Entity ID.

(b) Create an instance of the entity with the old entity's attributes and identified by new entity ID.

The application that relinquished control of the entity identified in the original entity ID field shall:

(a) Proceed from the Control Transfer state to the Ready state

(b) delete the entity from the its application

(c) replace the entity with the replacement entity from the application that has issued the Transfer Control Acknowledge PDU.

5.3.34 Acoustic System Record. Information about a particular UA emitter shall consist of three fields: Acoustic Name, Function and Acoustic ID Number. Enumeration values and bit-encoded values are given in document IST-CR-93-46 unless otherwise stated. These fields are described below:

(1) Acoustic Name - This field shall specify the UA emitter name for a particular emitter. This field shall be represented by a 16-bit enumeration.

(2) Function - This field shall specify the function for a particular UA emitter. Typical functions include active sonar transmissions and propulsion components. This field is intended to help receiving entities determine if the UA emission is of interest to the systems simulated by that entity. This field shall be represented by an 8-bit enumeration.

(3) Acoustic Identification Number - This field shall specify the UA emitter identification number relative to a specific system. This field shall be represented by an 8-bit unsigned integer. This field allows the differentiation of multiple systems on an entity, even if in some instances two or more of the systems may be identical UA emitter types.

The Acoustic System Record is represented in figure 5-58.

Acoustic Name	16-bit Enumeration
Function	8-bit Enumeration
Acoustic ID Number	8-bit Unsigned Integer

Figure 5-58 Acoustic System Record

5.3.35 Entity Aggregate Descriptor Record. Each Entity Aggregate Descriptor Record describes the location, orientation and shape of one Aggregate. An Aggregate Descriptor field contains a list of the Unique Entity IDs of its component entities and a list of the Unique Aggregate ID's of its subordinate Aggregates. The Aggregate Descriptor field should only list Entity ID's for entities that are issuing Entity State PDUs.

An Aggregate Descriptor Field with zero (0) entities and subordinate aggregates means the aggregate is on the lowest hierarchical level and is not issuing DIS PDUs (has not been de-aggregated).

Entities can move from one aggregate to another if their entity ID's are listed in a different descriptor field. Entity ID's may

only be listed in one Aggregate Descriptor at a time. Aggregate can change hierarchy by changing the descriptor field. Multiple level aggregates should always describe all levels of the Aggregate, even if one or more of the lower level aggregates have been de-aggregated.

(a) Unique Aggregate ID - The unique aggregate ID is assigned by the Aggregate Controller. It is unique in the same manner entity ID's are unique. Aggregate ID's may not be a duplicate of unique Entity ID's.

(b) Aggregate Shape Field - This field describes the shape or formation of the entities within the aggregate. The formations are aligned on the orientation vector. The first bit describes the whether or not the entities are on the earth's surface. Bits 1 to 15 describe the shape of the aggregate. Figure 7 describes the contents of bits 1 to 15.

(c) Number of Entities in Aggregate - Number of entities or subordinate entities in aggregate field. An aggregate can contain either constituent entities or subordinate aggregates, or both. A non-zero value in one of those fields represents the number of entities or subordinate aggregates within the Aggregate Descriptor. If M and m are both non-zero the list of Unique Entity ID's (1 to M) would be followed by the list of Unique Aggregate ID's (1 to m).

The Entity Aggregate Descriptor Record is represented in figure 5-59.

Unique Aggregate ID	16-bit Site ID 16-bit Application ID 16-bit Group ID
Size of Descriptor (in bytes)	16-bit unsigned integer
Aggregate Shape	16-bit enumeration
Number of Entities in Aggregate (M)	16-bit unsigned integer
Number of Aggregate ID's (m)	16-bit unsigned integer
Aggregate Hierarchical Name	16-bit enumeration
Length of Rectangular Solid Enclosing Aggregate Along X Velocity Axis	32-bit floating point
Length of Rectangular Solid Enclosing Aggregate Along Y Velocity Axis	32-bit floating point
Length of Rectangular Solid Enclosing Aggregate Along Z Velocity Axis	32-bit floating point
Position of Aggregate Center	X: 64-bit floating point Y: 64-bit floating point Z: 64-bit floating point
Aggregate Orientation	Psi: 32-bit floating point Theta: 32-bit floating point Phi: 32-bit floating point

Figure 5-59
Entity Aggregate Descriptor Record

Aggregate Velocity	X: 32-bit floating point Y: 32-bit floating point Z: 32-bit floating point
Unique Entity ID #1	16-bit Site ID 16-bit Application ID 16-bit Entity ID
.	.
Unique Entity ID # M	16-bit Site ID 16-bit Application ID 16-bit Entity ID
Unique Aggregate ID # 1	16-bit Site ID 16-bit Application ID 16-bit Entity ID
.	.
Unique Aggregate ID # m	16-bit Site ID 16-bit Application ID 16-bit Entity ID

Figure 5-59
Entity Aggregate Descriptor Record Cont.

5.3.36 UA Fundamental Parameter Data Record. The UA Fundamental Parameter Data Record contains regeneration parameters that are variable throughout a scenario dependent on the actions of the participants in the simulation. This record also provides basic parametric data that may be used to support low-fidelity simulations which may not have the processing capability to model a high-fidelity regeneration of emission beams. This record shall consist of ten fields. These fields are described below:

(1) Pulse Type - This field shall specify the type of pulse (usually from an active sonar or descriptive of a short duration UA emission) of the UA emission. This field shall be represented by a 16-bit enumeration.

(2) Scan pattern - This field shall specify the scan pattern employed. This field shall be represented by a 16-bit enumeration.

(3) Pulse Frequency - This field shall specify the center of frequency of the UA emission in hertz. Frequency modulation for a particular emitter and mode shall be derived from Pulse type and Scan Pattern. This field shall be represented by a 32-bit floating point number.

(4) Source Level - This field shall specify the source level for the emission in decibels relative to one micro Pascal. This field shall be represented by a 32-bit floating point number.

(5) Pulse Duration - This field shall specify the average pulse duration of the UA emission in milli-seconds. This field shall be represented by a 32-bit floating point number.

(6) Pulse Repetition Frequency (PRF) - This field specifies the average PRF of the UA emission in hertz. This field shall be represented by a 32-bit floating point number.

(7) Main Beam Bearing - This field shall specify the azimuthal bearing in degrees of the main beam (as opposed to any sidelobe) in the emitter coordinate system. This field shall be represented by a 32-bit floating point number. Omnidirectional beams have a bearing of zero.

(8) Main Beam D/E - This field shall specify the depression/elevation (D/E) and angle m degrees of the main beam (as opposed to any sidelobe) in the emitter coordinate system. Positive angles are as shown in the figure. This field shall be represented by a 32-bit floating point number. D/E omnidirectional beams have D/E of zero.

(9) Main Beam Width - This field shall specify the width in degrees of the main beam at the half power points. Omnidirectional beams have a width of 360. This field shall be represented by a 32-bit floating point number.

(10) Main Beam Height - This field shall specify the height in degrees of the main beam at the half power points. Omnidirectional beams have a height of 360. This field shall be represented by a 32-bit floating point number.

The Fundamental Parameter Data Record is shown in Figure 5-60.

Pulse Type	16-bit Enumeration
Scan Pattern	16-bit Enumeration
Pulse Frequency	32-bit Floating Point
Source Level	32-bit Floating Point
Pulse Duration	32-bit Floating Point
Pulse Repetition Frequency	32-bit Floating Point
Main Beam Bearing	32-bit Floating Point
Main Beam D/E	32-bit Floating Point
Main Beam Width	32-bit Floating Point
Main Beam Height	32-bit Floating Point

Figure 5-60 UA Fundamental Parameter Data Record

5.4.9 Aggregate Protocol Family. (see also 4.4.9)

5.4.9.1 Aggregate Descriptor PDU. The Aggregate Descriptor PDU consists of a header, and zero or more Aggregate Descriptors fields.

(1) Aggregate Descriptor PDU Header - This field shall contain data common to the standard PDU header information (see 5.3.24 of document IST-CR-93-40).

(2) Number of Aggregate Descriptors - This field shall contain the number of Aggregate Descriptor fields.

The Aggregate Descriptor PDU is represented in figure 5-61.

PDU Header	PDU Header Record
Originating Entity ID	Site: 16-bit unsigned integer Application: 16-bit unsigned integer Entity: 16-bit unsigned integer
Number of Aggregate Descriptors (N)	32-bit unsigned integer
Aggregate Descriptor 1	Entity Aggregate Descriptor Record
.	.
Aggregate Descriptor N	Entity Aggregate Descriptor Record

Figure 5-61
Aggregate Descriptor PDU

5.4.9.2 De-Aggregate Request PDU. The Aggregate Descriptor PDU consists of a header, and zero or more Aggregate Descriptors fields.

(1) Aggregate Descriptor PDU Header - This field shall contain data common to the standard PDU header information (see 5.3.24 of document IST-CR-93-40).

(2) Number of Aggregate Descriptors - This field shall contain the number of Aggregate Descriptor fields.

The De-Aggregate Request PDU is represented in figure 5-62.

PDU Header	PDU Header Record
Originating Entity ID	Site: 16-bit unsigned integer Application: 16-bit unsigned integer Entity: 16-bit unsigned integer
Unique Aggregate ID of Receiving Aggregate Descriptor	16-bit unsigned integer
Padding	32 bits unused
Detection Range of Sending Entity	64-bit floating point

Figure 5-62
De-Aggregate Request PDU

5.4.9.3 Re-Aggregate Request PDU. This PDU is sent by the Aggregate Controller before it re-aggregates and stops sending Entity State PDUs for its component entities.

5.4.9.3.1 Re-Aggregation Timing. The time of re-aggregation must be at least 10 seconds after the time the PDU is issued. This is to enable a receiving entity or an exercise controller to stop the re-aggregation. This is done by sending a De-Aggregate Request PDU in the manner described above.

The Re-Aggregate Request PDU is represented in figure 5-63.

PDU Header	PDU Header Record
Originating Entity ID	Site: 16-bit unsigned integer Application: 16-bit unsigned integer Entity: 16-bit unsigned integer
Unique Aggregate ID of Sending Aggregate Descriptor	16-bit unsigned integer
Simulation Time When Re-Aggregate Will Occur	Hours: 32-bit integer Time Past The Hour: 32-bit unsigned integer

Figure 5-63
Re-Aggregate Request PDU

5.4.10 Distributed Underwater Acoustic Regeneration Protocol Family. (see 4.4.10)

5.4.10.1 Underwater Acoustic PDU. Information about acoustic emissions shall be communicated using an Underwater Acoustic (UA) PDU. The UA PDU shall contain the following fields:

(1) PDU Header - These fields shall identify the protocol version number, the exercise identifier, the protocol data unit type and shall include a timestamp and PDU data length field.

(2) Emitting Entity ID - This field shall identify the entity that is the source of UA emission. The field shall be represented by an Entity Identifier Record (see 5.3.18 of document IST-CR-93-40).

(3) Event Identification - This field shall contain a number generated by the issuing simulator to associate related events. This field shall be represented by an Event Identifier Record (see 5.3.18 of document IST-CR-93-40).

(4) State Update Indicator - This field shall used to indicate if the data in the PDU represents a state update or just data that has changed since issuance of the last UA PDU (relative to the identified entity and UA emission systems(s)). This field shall be represented by an 8-bit enumeration. A value of zero (0) shall indicate a state while a value of (1) shall indicate a changed data update.

(5) Number of Systems - This field shall specify the number of UA emission systems being described in the current PDU. One, several or all of the UA emission systems on a particular entity may be described in a single UA PDU[1]. This field shall be represented by an 8-bit unsigned integer. The following information shall be provided for each system:

(a) System Data Length - This field shall specify the length of this systems data in 32-bit words including the System Data Length Field. This field shall be represented by an 8-bit unsigned integer.

(b) Number of Beams - This field shall specify the number of beams being described in the current PDU

---

[1]It should be noted that caution must be exercised to ensure that the number of systems being described does not cause the PDU length to exceed the maximum PDU length of 255 thirty-two-bit words.

for the system being described. This field shall be represented by an 8-bit unsigned integer.

(c) Acoustic System - This field shall specify information about a particular emitter's system. This field shall be represented by an Acoustic System Record (see 5.4.10.1.1).

(d) Location - This shall specify the location of the acoustic source the emitting entity's coordinate system. This location shall be the origin of the UA emission coordinate system which shall be parallel to the entity coordinate system. This field shall be represented by an Entity Coordinate Vector Record (see ****).

(i) Beam Data Length - This field shall specify the length of this beams data in 32-bit words including the beam data length field. This field shall be represented by an 8-bit unsigned integer.

(ii) Beam ID Number - This field shall specify a unique emission database number assigned to differentiate between otherwise similar or identical UA emitter beams within a UA emitter system. This field shall be represented by an 8-bit unsigned integer.

(iii) Operating Mode - This field shall specify a system and perhaps beam operating mode including off. This field shall be represented by an 16-bit enumeration.

(iv) Shaft RPM - This field shall specify a propeller shaft rotation rate in revolutions per minute. This field shall be represented by an 16-bit unsigned integer.

(v) Masker (on/off) - This field shall be used to indicate if an entity is employing an active acoustic masker. This field shall be 8-bit enumeration.

(vi) Additional Source Activity (SEDB/ANDB) - This field shall specify an index number for additional acoustic sources such as special effects (ie. torpedo tube flooding) and additional narrow band sources. This index shall be used to identify specific additional source activity data as a Special Effects Database (SEDB) and an Additional Narrow Band

Database (ANBD). This field shall be represented by a 32-bit unsigned integer.

(vii) Fundamental Data Parameters - This field shall specify dynamic parameters of the UA emitter. This shall be represented by a UA Fundamental Parameter Data Record (see 5.3.36).

The maximum number of Systems (N_{max}) and beams in system i ($M_{i\text{-}max}$) is a function of the maximum PDU size specified in the Standard for Interactive Simulation - Communication Architecture for Distributed Interactive Simulation, and must adhere to the following relationship:

$$160\ N_{max} + \sum_{i=1}^{N_{max}} 384\ M_i \leq \text{MAX PDU size} - 224$$

The Underwater Acoustic PDU is represented in figure 5-64.

UNDERWATER ACOUSTIC PDU		
FIELD SIZE	FIELD NAME	FIELD TYPE
96	PDU HEADER	Protocol Version - 8-bit enumeration
		Exercise ID - 8-bit unsigned integer
		PDU Type - 8-bit unsigned integer
		Padding - 8-bits unused
		Time Stamp - 32-bit unsigned integer
		Length - 16-bit unsigned integer
		Padding - 16-bits unused
48	EMITTING ENTITY ID	Site - 16-bit unsigned integer
		Application - 16-bit unsigned integer
		Entity - 16-bit unsigned integer
48	EVENT ID	Site - 16-bit unsigned integer
		Application - 16-bit unsigned integer
		Event - 16-bit unsigned integer
8	STATE UPDATE INDICATOR	8-bit enumeration
8	NUMBER OF SYSTEMS	8-bit unsigned integer
16	PADDING	16-bit unsigned integer
VARIES	SYSTEM DATA LENGTH	8-bit unsigned integer
	NUMBER OF BEAMS (M)	8-bit unsigned integer
	PADDING	16-bit unused
	ACOUSTIC SYSTEM	Acoustic Name - 16-bit enumeration
		Function - 8-bit enumeration
		Acoustic ID Number - 8-bit enumeration
	LOCATION (with respect to the entity)	X Component - 32-bit floating point
		Y Component - 32-bit floating point
		Z Component - 32-bit floating point

Figure 5-64 Underwater Acoustic PDU

UNDERWATER ACOUSTIC PDU CONT.		
FIELD SIZE	FIELD NAME	FIELD TYPE
	OPERATING MODE	16-bit enumeration
	SHAFT-RPM	16-bit unsigned integer
	MASKER (ON/OFF)	8-bit enumeration
	PADDING	8-bits unused
	ADDITIONAL SOURCE ACTIVITY (SEBD/ANDB)	32-bit unsigned integer
	FUNDAMENTAL DATA PARAMETERS	Pulse Type - 16-bit enumeration
		Scan Pattern - 16-bit enumeration
		Pulse Frequency - 32-bit floating point
		Source Level - 32-bit floating point
		Pulse Duration - 32-bit floating point
		Pulse Repetition Freq. 32-bit floating point
		Main Beam Bearing - 32-bit floating point
		Main Beam D/E - 32-bit floating point
		Main Beam Width - 32-bit floating point
		Main Beam Height - 32-bit floating point

Figure 5-64 Underwater Acoustic PDU Cont.

5.4.11 Destructible Entity Protocol Family.

5.4.11.1 Creation/Modification PDU. Addition or changes in the simulation environment shall be communicated by issuing a Creation/Modification PDU. The Creation/Modification PDU shall contain the following fields:

(a) PDU Header - These fields shall identify the protocol version number, the exercise identifier, and the protocol data unit type. The PDU Header shall be represented by the PDU Header described in 5.3.3 of document IST-CR-93-40.

(b) Sequence Number - This field shall specify the sequence number for this PDU. At the beginning of an exercise it will initialized to zero and sequential incremented for each new creation or modification.

(c) Object ID - This field shall identify the type of object to be created.

(d) Location in the World - This field shall be the location, in world coordinates, at which the object is to be created or modified. This field shall be represented by a World Coordinate Record.

(e) Orientation - This filed shall represent the orientation of the entity. This is necessary in order to ensure that the key features of the object are correctly oriented in each simulation.

(f) Height - These fields shall contain the minimum, and maximum Z coordinate values.

(g) Length - These fields shall contain the minimum and maximum Y coordinate values.

(h) Width - These fields shall contain the minimum and maximum X coordinate values.

(i) Material Composition - This field shall represent the material makeup of an entity object.

(j) Appearance - This field shall represent the modification to an object based upon DIS Appendix D, see 30.5.

5.4.11.2 Deletion PDU. Deletion of an object or feature in the simulation environment shall be communicated by issuing a Deletion PDU. The Deletion PDU shall contain the following:

(a) PDU Header - These fields shall identify the protocol version number, the exercise identifier, and the protocol data unit type. The PDU Header shall be represented by the PDU Header described in 5.3.3 of document IST-CR-93-40.

(b) Sequence Number - This field shall specify the sequence number for this PDU. At the beginning of an exercise it will initialized to zero and sequential incremented for each new creation or modification.

(c) Timestamp - This field shall specify the time at which the data in the PDU is valid. The field shall be represented by a Timestamp.

(d) Object ID - This field shall identify the type of object to be deleted.

(e) Location to the World - This field shall be the location, in world coordinates, at which the object is to be created or modified. This field shall be represented by a World Coordinated Record.

5.4.11.3 Request ID PDU. Requesting the maximum sequence number from a responsible host in the simulation environment shall be communicated by issuing a Request ID PDU. The Request ID PDU shall contain the following fields:

(a) PDU Header - These fields shall identify the protocol version number, the exercise identifier, and the protocol data unit type. The PDU Header shall be represented by the PDU Header described in 5.3.3 of document IST-CR-93-40.

5.4.11.4 Reply ID PDU. The Reply from a responsible host of a matrix of maximum sequence numbers of destructible entities in the simulation environment shall be communicated by issuing a Reply ID PDU. The Reply ID PDU shall contain the following fields:

(a) PDU Header - These fields shall identify the protocol version number, the exercise identifier, and the protocol data unit type. The PDU Header shall be represented by the PDU Header described in 5.3.3 of document IST-CR-93-40.

(b) Maximum Sequence Number Matrix - This field shall specify a matrix of maximum sequence number of destructible entities issued by simulation host. At the beginning of an exercise it will initialized to zero and incremented for each new creation or modification PDU issues by a host within its responsibility..

5.4.11.5 Request Object PDU. The Request of a matrix of destructible entities in the simulation environment shall be communicated by issuing a Request Object PDU. The Request Object PDU shall contain the following fields:

(a) PDU Header - These fields shall identify the protocol version number, the exercise identifier, and the protocol data unit type. The PDU Header shall be represented by the PDU Header described in 5.3.3 of document IST-CR-93-40.

(b) Sequence Minimum and Maximum Numbers - The bounding values for the sequence numbers of the original PDUs.

(c) Time Minimum and Maximum Values - The bounding values for the time stamps of the original PDUs.

(d) Geographic Bounds Minimum and Maximum - The minimum and maximum values of the geographic bounding box containing the query. This field shall be represented by two World Coordinate Record.

5.4.11.6 Reply Object PDU. The reply of requested objects from a responsible host shall be communicated by issuing a Reply Object PDU. The Reply Object PDU shall contain the following fields:

(a) PDU Header - These fields shall identify the protocol version number, the exercise identifier, and the protocol data unit type. The PDU Header shall be represented by the PDU Header described in 5.3.3 of document IST-CR-93-40.

(b) Maximum ID Number - This field shall specify the number of objects or areal features PDU's to be sent based upon a request.

(c) Count - This field shall contain a count of the number of Create/Modify PDUs contained in the array portion of the record.

(d) Retransmitted PDUs - This field shall contain a variable length array of the stored Create/Modify PDUs.

5.4.11.7 Areal Feature PDU. The creation or modification of a Areal Feature in a simulation exercise shall be communicated by issuing a Areal Feature PDU. The Areal Feature PDU shall contain the following fields:

(a) PDU Header - These fields shall identify the protocol version number, the exercise identifier, and the protocol data unit type. The PDU Header shall be represented by the PDU Header described in 5.3.3 of document IST-CR-93-40.

(b) Sequence Number - This field shall specify the sequence number for this PDU. At the beginning of an exercise it will be initialized to zero and sequentially incremented for each new creation or modification.

(c) Timestamp - This field shall specify the time at which the data in the PDU is valid. The field shall be represented by a Timestamp.

(d) Object ID - This field shall identify the type of feature to be created.

(e) Location in the World - This field shall be the bounding box location, in world coordinates, at which the feature is to be created or modified. This field shall be represented by a World Coordinate Record.

(f) Segment Count - This field shall represent the number of segments that represent the areal feature.

(g) Cross Section Count - This field shall represent the number of points in the cross section.

(h) Appearance - This field shall represent the modification to an areal feature based upon DIS Appendix D, see 30.5.

(i) Material Composition - This field shall represent the material makeup of the feature.

(j) Segment Coordinates - The origin of the segments in world coordinates. The number of segments equals the value in the Segment Count. This field shall be represented by a World Coordinate Record.

(k) Cross Section Coordinates - The offset values of the way points from the origin given in the Segment Coordinates record corresponding to the point. 32 bit float values.

5.4.12 Dynamic Environment Protocol Family.

5.4.12.1 Environmental PDU. Information about an environmental entity shall be communicated by issuing an Environmental PDU. The environmental PDU shall contain the following fields:

(a) PDU Header - This field shall contain data common to all DIS PDUs. The PDU header shall be represented by the PDU Header Record.

(b) Entity ID - This field shall identify the entity issuing the PDU. This field shall be represented by an Entity Identifier Record.

(c) Entity type - This field shall identify the entity type to be simulated by members of the session with the same Exercise ID field in the PDU Header Record.

(d) Entity location - This field shall specify the entity's physical location in the simulated world. This field shall be represented by a World Coordinates Record.

(e) Entity linear velocity - This field shall specify the entity's linear velocity. This field shall be represented by a Linear Velocity Vector Record.

(f) Entity Orientation - This field shall specify the entity's orientation. This field shall be represented by the Euler Angles Record.

(g) Number of parameters - This field shall specify the number of parameters required to physically describe environmental entity. This field shall be represented by an 8-bit unsigned integer.

(h) Physical parameters - This field shall specify the parameter values for the representation of the environmental entity. Each physical parameter shall be represented by a Physical Parameter record.

The Environmental PDU is represented in figure 5-65.

ENVIRONMENTAL PDU		
FIELD SIZE	FIELD NAME	FIELD TYPE
96	PDU HEADER	Protocol Version - 8 bit enumeration
		Exercise ID - 8 bit unsigned integer
		PDU Type - 8 bit unsigned integer
		Padding - 8 bits unused
		Time Stamp - 32 bit unsigned integer
		Length - 16 Bit unsigned integer
		Padding - 16 bits unused
48	ENTITY ID	Site - 16 bit unsigned integer
		Application - 16 bit unsigned integer
		Entity - 16 bit unsigned integer
64	ENTITY TYPE	Entity Kind - 8 bit enumeration
		Domain - 8 bit enumeration
		Country - 8 bit enumeration
		Category - 8 bit enumeration
		Subcategory - 8 bit enumeration
		Specific - 8 bit enumeration
		Extra - 8 bit enumeration
192	ENTITY LOCATION	X - Component - 64 bit floating point
		Y - Component - 64 bit floating point
		Z - Component - 64 bit floating point

ENVIRONMENTAL PDU		
96	ENTITY LINEAR VELOCITY	X - Component - 32 bit floating point
		Y - Component - 32 bit floating point
		Z - Component - 32 bit floating point
192	ENTITY ORIENTATION	Psi - 64 bit floating point
		Theta - 64 bit floating point
		Phi - 64 bit floating point
8	NUMBER OF PARAMETERS	N - 8 bit unsigned integer
N * 48	PHYSICAL PARAMETERS	Parameter Type - 16 bit unsigned integer
		Parameter Value - 32 bit floating point

Figure 5-65 Environmental PDU

5.4.13 Entity Handover Protocol Family.

5.4.13.1 Transfer Control Request PDU. The Transfer Control Request PDU shall contain the following information:

(1) The identification of the application issuing the PDU.

(2) The identification of the entity that the issuing application wants to control.

The Transfer Control Request PDU is represented in figure 5-66.

TRANSFER CONTROL REQUEST PDU		
FIELD SIZE	FIELD NAME	FIELD TYPE
96	PDU HEADER	Protocol Version - 8 bit enumeration
		Exercise ID - 8 bit unsigned integer
		PDU Type - 8 bit unsigned integer
		Padding - 8 bits unused
		Time Stamp - 32 bit unsigned integer
		Length - 16 Bit unsigned integer
		Padding - 16 bits unused
48	ENTITY ID	Site - 16 bit unsigned integer
		Application - 16 bit unsigned integer
		Entity - 16 bit unsigned integer
32	REQUESTING APPLICATION	Site - 16 bit unsigned integer
		Application - 16 bit unsigned integer

Figure 5-66 Transfer Control Request PDU

5.4.13.2 Transfer Control PDU. The Transfer Control PDU shall contain the following public information (normally found in the Entity State PDU):

(1) The identification of the entity being transferred.

(2) Identification of the force to which the entity belongs

(3) The entity's specific entity type

(4) The issuing entity's alternate entity type for use with the Guise function

(5) Information about the location of the entity in the simulated world and its orientation. This information includes the entity's:

(a) Location in world coordinates

(b) Velocity

(c) Orientation in vehicular coordinates

(d) Dead reckoning parameters and algorithms

(6) The information required for the representation of the vehicle's representation. This information includes but is not limited to:

(a) Modifiers to the entity's representation

(b) Markings

(c) Articulated parts

(d) The presence of external parts or stores

(7) The capabilities of the entity. Defined capabilities include:

(a) Resupply

(b) Repair

The Transfer Control PDU shall contain the private information (typically only known by the controlling application) that is required to assume control of the entity without losing its internal states:

(1) Stores Status (Fuel level, Ammunition Level, etc.)

(2) Internal Damage or Casualties

(3) Plans (flight plans, search presets, etc.)

The Transfer Control PDU is represented in figure 5-67.

TRANSFER CONTROL PDU		
FIELD SIZE	FIELD NAME	FIELD TYPE
96	PDU HEADER	Protocol Version - 8 bit enumeration
		Exercise ID - 8 bit unsigned integer
		PDU Type - 8 bit unsigned integer
		Padding - 8 bits unused
		Time Stamp - 32 bit unsigned integer
		Length - 16 Bit unsigned integer
		Padding - 16 bits unused
48	ENTITY ID	Site - 16 bit unsigned integer
		Application - 16 bit unsigned integer
		Entity - 16 bit unsigned integer
32	REQUESTING APPLICATION	Site - 16 bit unsigned integer
		Application - 16 bit unsigned integer
8	FORCE ID	8 bit enumeration
8	# OF ARTICULATION PARAMETERS	N - 8 bit unsigned integer
16	# OF OTHER ENTITY STATE PARAMETERS	M - 16 bit unsigned integer
16	PADDING	Padding - 16 bits unused
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

FIELD SIZE (bits)	TRANSFER CONTROL PDU FIELDS (cont'd)	
64	ALTERNATIVE 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
96	ENTITY LINEAR VELOCITY	X - Component - 32 bit floating point
		Y - Component - 32 bit floating point
		Z - Component - 32 bit floating point
192	ENTITY LOCATION	X - Component - 64 bit floating point
		Y - Component - 64 bit floating point
		Z - Component - 64 bit floating point
96	ENTITY ORIENTATION	Psi - Component - 32 bit floating point
		Theta - Component - 32 bit floating point
		Phi - Component - 32 bit floating point
32	ENTITY APPEARANCE	32 bit record of enumerations
320	DEAD RECKONING PARAMETERS	Dead Reckoning Algorithm - 8 bit enumeration
		Other parameters - 120 bits unused
		Entity Linear Acceleration - 3 x 32 bit floating point
		Entity Angular Acceleration - 3 x 32 bit floating point
96	ENTITY MARKING	Character Set - 8 bit enumeration
		11 x 8 bit unsigned integer
32	CAPABILITIES	32 Boolean fields
n x 128	ARTICULATION PARAMETERS	Change - 16 bit unsigned integer
		ID - Attached to - 16 bit unsigned integer
		Parameter type - 32 bit parameter type record
		Parameter value - 64 bit variable type

FIELD SIZE (bits)	TRANSFER CONTROL PDU FIELDS (cont'd)	
m X 96	OTHER ENTITY STATE PARAMETERS	Parameter type - 32 bit parameter type record
		Parameter value - 64 bit variable type

Figure 5-67 Transfer Control PDU

5.4.13.3 Transfer Control Acknowledge PDU. The Transfer Control Acknowledge PDU shall contain the following information:

(1) The transferred entity's original Entity ID (see 5.3.8 of document IST-CR-93-40).

(2) The entity's new Entity ID (see 5.3.8 of document IST-CR-93-40).

The Transfer Control PDU is represented in figure 5-68.

TRANSFER CONTROL ACKNOWLEDGE PDU		
FIELD SIZE	FIELD NAME	FIELD TYPE
96	PDU HEADER	Protocol Version - 8 bit enumeration
		Exercise ID - 8 bit unsigned integer
		PDU Type - 8 bit unsigned integer
		Padding - 8 bits unused
		Time Stamp - 32 bit unsigned integer
		Length - 16 Bit unsigned integer
		Padding - 16 bits unused
48	OLD ENTITY ID	Site - 16 bit unsigned integer
		Application - 16 bit unsigned integer
		Entity - 16 bit unsigned integer
48	NEW ENTITY ID	Site - 16 bit unsigned integer
		Application - 16 bit unsigned integer
		Entity - 16 bit unsigned integer

Figure 5-68 Transfer Control Acknowledge PDU