1-1-1991

Entity Information And Entity Interaction In A Distributed Interactive Simulation: Military Standard (draft)

University of Central Florida Institute for Simulation and Training

INSTITUTE FOR SIMULATION AND TRAINING

[redacted]

January 25, 1991

ENTITY INFORMATION AND ENTITY INTERACTION IN A DISTRIBUTED INTERACTIVE SIMULATION

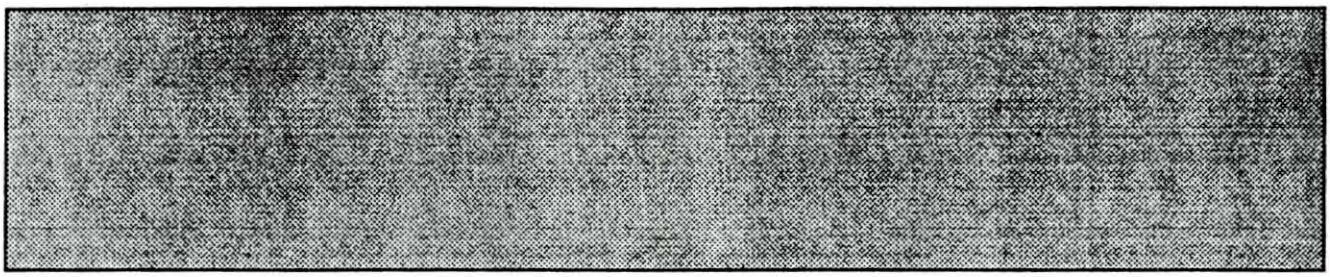
DRAFT

IST-PD-90-2

[redacted]

IST

INSTITUTE FOR SIMULATION AND TRAINING

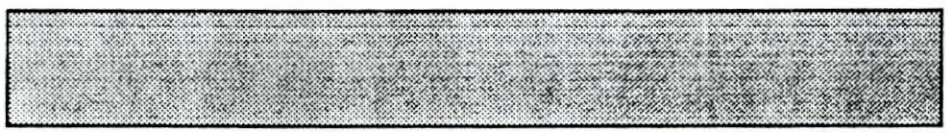


Contract Number N61339-89-C-0043
PM TRADE
DARPA

January 25, 1991

MILITARY STANDARD (DRAFT)

Entity Information and Entity Interaction in a Distributed Interactive Simulation



IST

Institute for Simulation and Training
12424 Research Parkway, Suite 300
Orlando FL 32826

University of Central Florida
Division of Sponsored Research

IST-PD-90-2
(Revised)

METRIC

DRAFT STANDARD
25 JANUARY 1991

MILITARY STANDARD

PROTOCOL DATA UNITS FOR

ENTITY INFORMATION AND ENTITY INTERACTION IN A

DISTRIBUTED INTERACTIVE SIMULATION

"NOTE: This draft, dated 25 January 1991, prepared by the Institute for Simulation and Training for PM TRADE, has not been approved and is subject to modification. DO NOT USE PRIOR TO APPROVAL. (Project _____)"

FOREWORD

1. This draft military standard has been prepared by the Institute for Simulation and Training for PM TRADE and DARPA based upon currently available technical information but it has not been approved for promulgation. It is subject to modification. However, pending its promulgation as a coordinated military standard, it may be used.

2. Beneficial comments (recommendations, additions, deletions) and any pertinent data which may be of use in improving this document should be addressed to: Dr. Bruce McDonald, Institute for Simulation and Training, 12424 Research Parkway, Suite 300, Orlando, FL 32826 by using the self-addressed Standardization Document Improvement Proposal Form appearing at the end of this document or by letter.

3. This standard contains requirements for a communication protocol between simulators in a Distributed Interactive Simulation (DIS). This protocol focuses on communication related to information about a simulated entity and the interactions that take place between simulated entities. Because of this fact, protocol related to other aspects of DIS are not specified as requirements. In particular, this standard does not specify protocol related to data collection, network management, or terrain changes. The protocol data units contained in this draft standard emphasize target engagements that take place on or above land. Detection of targets based on their emission of sound or electromagnetic energy is not currently supported in the protocol data units. The surface and subsurface ocean environment is also not currently represented. The system designer must recognize this fact and design software that will be flexible enough to incorporate new protocols and protocol data units as they become available. This flexibility must also allow for updates in the protocol that may result from technological advances and increased fidelity requirements.

4. Some protocol data units (PDUs) that are not required as part of this standard are recommended for use until a standard protocol is developed. These PDUs are included in section 6 of this document.

C O N T E N T S

PARAGRAPH		PAGE
1.	SCOPE	1
	1.1 Scope	1
	1.2 Application	1
2.	APPLICABLE DOCUMENTS	2
	2.1 Government documents	2
	2.1.1 Specifications, standards, and handbooks	2
	2.2 Non-Government publications	2
3.	DEFINITIONS	3
	3.1 Acronyms used in this standard	3
	3.2 Application layer	4
	3.3 Articulated Part	4
	3.4 Ballistic munition	4
	3.5 Binary Angle Measurement (BAM)	4
	3.6 Bit	5
	3.7 Broadcast	5
	3.8 Byte	5
	3.9 Connectionless service	5
	3.10 Data link layer	5
	3.11 Dead reckoning	5
	3.12 Distributed Interactive Simulation (DIS)	5
	3.13 Distributed Simulation	5
	3.14 Emitter	5
	3.15 Entity	5
	3.16 Entity coordinates	5
	3.17 Euler angles	6
	3.18 Exercise	6
	3.19 Fidelity	6
	3.20 Fields	6
	3.21 Guided munition	6
	3.22 Host or Host computer	6
	3.23 Layer	6
	3.24 Layering	6
	3.25 Local Area Network (LAN)	6
	3.26 Long-Haul network	7
	3.27 Multicast	7
	3.28 Network architecture	7
	3.29 Network management	7
	3.30 Node	7
	3.31 Octet	7
	3.32 Open system architecture	7

C O N T E N T S

PARAGRAPH		PAGE
3.33	OSI Reference Model	7
3.34	Protocol Data Unit (PDU)	7
3.35	Physical layer	7
3.36	Presentation layer	7
3.37	Protocol	7
3.38	Real-Time	8
3.39	Session layer	8
3.40	Simulated entity	8
3.41	Simulation exercise	8
3.42	Simulation management	8
3.43	Simulation manager computer	8
3.44	Transport layer	8
3.45	Wide-Area Network (WAN)	8
4.	GENERAL REQUIREMENTS	9
4.1	Introduction	9
4.2	Authority in charge of DIS exercises	9
4.2.1	Site manager	9
4.2.2	Simulation Manager Computer	9
4.3	DIS exercise	9
4.4	The issuing entity	9
4.5	Issuance of PDUs	9
4.6	Receipt of PDUs	9
4.7	Protocol data units for DIS	9
4.7.1	Protocol data unit header	10
4.7.1.1	Protocol version	10
4.7.1.2	DIS exercise identification	10
4.7.1.3	Types of PDUs	10
4.7.2	Entity Information	10
4.7.2.1	Entity State PDU	10
4.7.2.1.1	Information contained in the Entity State PDU	11
4.7.2.1.2	Issuance of the Entity State PDU	12
4.7.2.1.3	Receipt of the Entity State PDU	12
4.7.3	Weapons fire	12
4.7.3.1	Representation of weapons fire in DIS	12
4.7.3.2	Fire PDU	13
4.7.3.2.1	Information Contained in the Fire PDU	13
4.7.3.2.2	Issuance of the Fire PDU	13
4.7.3.2.3	Single rounds and bursts of fire	14
4.7.3.2.4	Receipt of the Fire PDU	14

C O N T E N T S

PARAGRAPH	PAGE
4.7.3.3 Detonation PDU	14
4.7.3.3.1 Information contained in the Detonation PDU	14
4.7.3.3.2 Issuance of the Detonation PDU	15
4.7.3.3.3 Inclusion of the Entity Identifier	15
4.7.3.3.4 Termination of the existence of munition entities	15
4.7.3.3.5 Sky shots	15
4.7.3.3.6 Receipt of the Detonation PDU	15
4.7.4 Dead reckoning	15
4.7.4.1 Dead reckoning and the issuing entity	16
4.7.4.2 Dead reckoning and the receiving entity	16
4.7.4.3 Dead reckoning algorithms	16
4.7.5 Update Threshold Control	16
4.7.5.1 Threshold values	16
4.7.5.2 Changing update threshold values	17
4.7.5.3 State information for Update Threshold Control	18
4.7.5.3.1 The requesting host computer	18
4.7.5.3.2 The responding host computer	20
4.7.5.4 Representation of Update Threshold Control in DIS	23
4.7.5.5 Update Threshold Request PDU	24
4.7.5.5.1 Information contained in the Update Threshold Request PDU	24
4.7.5.5.2 Issuance of the Update Threshold Request PDU	24
4.7.5.5.3 Receipt of the Update Threshold Request PDU	24
4.7.5.6 Update Threshold Response PDU	24
4.7.5.6.1 Information contained in the Update Threshold Response PDU	25
4.7.5.6.2 Issuance of the Update Threshold Response PDU	25
4.7.5.6.3 Receipt of the Update Threshold Response PDU	25
4.7.6 Logistics Support	25
4.7.6.1 Procedure for Logistics Support	25
4.7.6.2 State information for resupply	26
4.7.6.2.1 The receiving host computer	26
4.7.6.2.2 The supplying host computer	29

C O N T E N T S

PARAGRAPH	PAGE
4.7.6.3 State information for repair	31
4.7.6.3.1 The receiving host computer	31
4.7.6.3.2 The repairing host computer	33
4.7.6.4 Service Request PDU	35
4.7.6.4.1 Information contained in the Service Request PDU	35
4.7.6.4.2 Issuance of the Service Request PDU	35
4.7.6.4.3 Receipt of the Service Request PDU	35
4.7.6.5 Resupply Offer PDU	36
4.7.6.5.1 Information contained in the Resupply Offer PDU	36
4.7.6.5.2 Issuance of the Resupply Offer PDU	36
4.7.6.5.3 Receipt of the Resupply Offer PDU	36
4.7.6.6 Resupply Received PDU	36
4.7.6.6.1 Information contained in the Resupply Received PDU	36
4.7.6.6.2 Issuance of the Resupply Received PDU	37
4.7.6.6.3 Receipt of the Resupply Received PDU	37
4.7.6.7 Resupply Cancel PDU	37
4.7.6.7.1 Information contained in the Resupply Cancel PDU	37
4.7.6.7.2 Issuance of the Resupply Cancel PDU	37
4.7.6.7.3 Receipt of the Resupply Cancel PDU	37
4.7.6.8 Repair Complete PDU	37
4.7.6.8.1 Information contained in the Repair Complete PDU	37
4.7.6.8.2 Issuance of the Repair Complete PDU	37
4.7.6.8.3 Receipt of the Repair Complete PDU	38
4.7.6.9 Repair Response PDU	38
4.7.6.9.1 Information contained in the Repair Response PDU	38
4.7.6.9.2 Issuance of the Repair Response PDU	38
4.7.6.9.3 Receipt of the Repair Response PDU	38
4.7.6.9.4 Cancellation of repair service	38
4.7.7 Collisions	38
4.7.7.1 Collision PDU	38
4.7.7.1.1 Information contained in the Collision PDU	39
4.7.7.1.2 Issuance of the Collision PDU	39
4.7.7.1.3 Receipt of the Collision PDU	39
4.7.8 Electronic Interaction	39

C O N T E N T S

PARAGRAPH		PAGE
	4.7.8.1 Radar PDU	39
	4.7.8.1.1 Information contained in the Radar PDU	39
	4.7.8.1.2 Issuance of the Radar PDU	40
	4.7.8.1.3 Receipt of the Radar PDU	41
5.	DETAILED REQUIREMENTS	42
	5.1 Representation of data	42
	5.1.1 Byte ordering	42
	5.1.2 Enumeration representation	42
	5.1.3 Number representation	42
	5.1.3.1 Floating point	42
	5.1.3.2 Integers	42
	5.2 Basic data types and records	43
	5.2.1 Angle Representation	43
	5.2.2 Angular Velocity Vector Record	43
	5.2.3 Articulated Parts Records	43
	5.2.3.1 Articulated Parts Record	43
	5.2.3.1.1 Change indicator	44
	5.2.3.1.2 Part ID	44
	5.2.3.1.3 Number of parameters	44
	5.2.3.1.4 Parts Parameter Record	44
	5.2.3.1.4.1 Parameter type	44
	5.2.3.1.4.2 Parameter value	44
	5.2.4 Boolean	45
	5.2.5 Burst Descriptor Record	45
	5.2.5.1 Munition	45
	5.2.5.2 Warhead	45
	5.2.5.3 Fuze	45
	5.2.5.4 Quantity and rate	45
	5.2.6 Date/Time	46
	5.2.7 Entity Capabilities Record	46
	5.2.8 Entity Identifier Record	46
	5.2.8.1 Simulation Address Record	47
	5.2.8.1.1 Site Identifier	47
	5.2.8.1.2 Host Identifier	47
	5.2.8.2 Entity Identification Number	47
	5.2.9 Entity Marking Record	48
	5.2.10 Entity Type Record	48
	5.2.10.1 Kind	49
	5.2.10.2 Domain	49

C O N T E N T S

PARAGRAPH		PAGE
5.2.10.3	Country	49
5.2.10.4	Category	49
5.2.10.5	Subcategory	49
5.2.10.6	Specific	49
5.2.10.7	Extra	49
5.2.11	Euler Angles Record	50
5.2.12	Event Identifier Record	50
5.2.13	Exercise Identifier	51
5.2.14	Organizational Unit Record	51
5.2.14.1	Force ID	51
5.2.14.2	Country ID	51
5.2.14.3	Service ID	51
5.2.14.4	Hierarchy	51
5.2.15	Radar system data type	52
5.2.16	Repair type	53
5.2.17	Service type	53
5.2.18	Supply Quantity Record	53
5.2.18.1	Supply type	53
5.2.18.2	Quantity	53
5.2.19	Terrain Database Identifier Record	53
5.2.20	Time stamping	54
5.2.20.1	Absolute timestamp	54
5.2.20.2	Relative timestamp	54
5.2.20.3	Scale	54
5.2.21	Vector Record	54
5.2.21.1	Entity coordinates	54
5.2.21.2	Linear acceleration vector	55
5.2.21.3	Linear velocity vector	55
5.2.22	World Coordinates Record	55
5.3	Protocol Data Units for Distributed Interactive Simulation	56
5.3.1	Introduction	56
5.3.2	List of DIS Protocol Data Units	56
5.3.3	Protocol Data Unit (PDU) header	58
5.3.4	Entity Information	60
5.3.4.1	Entity State PDU	60
5.3.5	Entity Interaction	64
5.3.5.1	Weapons fire	64
5.3.5.1.1	Fire PDU	64
5.3.5.1.2	Detonation PDU	68

C O N T E N T S

PARAGRAPH	PAGE
5.3.5.2 Update Threshold Control	72
5.3.5.2.1 Update Threshold Request PDU	72
5.3.5.2.2 Update Threshold Response PDU	74
5.3.5.3 Logistic Support	76
5.3.5.3.1 Service Request PDU	76
5.3.5.3.2 Resupply Offer PDU	78
5.3.5.3.3 Resupply Received PDU	80
5.3.5.3.4 Resupply Cancel PDU	82
5.3.5.3.5 Repair Complete PDU	83
5.3.5.3.6 Repair Response PDU	84
5.3.5.4 Collisions	86
5.3.5.4.1 Collision PDU	86
5.3.5.5 Electronic interaction	88
5.3.5.5.1 Radar PDU	88
6. NOTES	91
6.1 Intended use	91
6.2 Description of Distributed Interactive Simulation	91
6.2.1 Introduction	91
6.2.2 Definition	91
6.2.3 DIS application	91
6.2.4 Requirements for DIS	91
6.2.4.1 Entity Information	92
6.2.4.1.1 Types	92
6.2.4.1.2 Location and orientation	92
6.2.4.1.3 Appearances	92
6.2.4.2 Entity Interaction	92
6.2.4.2.1 Weapons fire	92
6.2.4.2.2 Update Threshold Control	92
6.2.4.2.3 Logistic Support	93
6.2.4.2.4 Collisions	93
6.2.4.2.5 Electronic Interaction	93
6.2.4.3 DIS Management	93
6.2.4.3.1 Network Management	93
6.2.4.3.2 Simulation Management	93
6.2.4.3.3 Performance Measures	94
6.2.4.4 Environment Information	94
6.2.4.4.1 Changes in the terrain	94
6.2.4.4.2 Weather conditions	94
6.2.4.4.3 Degrees of ambient illumination	94
6.2.4.4.4 Other environmental effects	94

C O N T E N T S

PARAGRAPH	PAGE
6.3 Interim Protocol	94
6.3.1 Activation and Deactivation of Simulated Entities	94
6.3.1.1 Activate Request PDU	95
6.3.1.1.1 Information contained in the Activate Request PDU	95
6.3.1.1.2 Issuance of the Activate Request PDU	96
6.3.1.1.3 Receipt of the Activate Request PDU	96
6.3.1.2 Activate Response PDU	96
6.3.1.2.1 Information contained in the Activate Response PDU	96
6.3.1.2.2 Issuance of the Activate Response PDU	96
6.3.1.2.3 Receipt of the Activate Response PDU	96
6.3.1.3 Deactivate Request PDU	97
6.3.1.3.1 Information contained in the Deactivate Request PDU	97
6.3.1.3.2 Issuance of the Deactivate Request PDU	97
6.3.1.3.3 Receipt of the Deactivate Request PDU	97
6.3.1.4 Deactivate Response PDU	97
6.3.1.4.1 Information contained in the Deactivate Response PDU	98
6.3.1.4.2 Issuance of the Deactivate Response PDU	98
6.3.1.4.3 Receipt of the Deactivate Response PDU	98
6.3.2 PDUs associated with activation and deactivation	98
6.3.2.1 Activate Request PDU	98
6.3.2.2 Activate Response PDU	104
6.3.2.3 Deactivate Request PDU	106
6.3.2.4 Deactivate Response PDU	107
6.4 Proposed protocol	108
6.4.1 Emitter Type Record	108
6.4.1.1 Emitter class	108
6.4.1.2 Database entry number	108
6.4.1.3 Mode number	108
6.4.2 Conditions for use of the Emitter PDU	108
6.4.3 Emitter PDU	109

C O N T E N T S

PARAGRAPH		PAGE
6.5	Examples of logistic support	112
6.5.1	Resupply Service	112
6.5.2	Repair Service	113
6.6	Example of Update Threshold Control	113
6.7	Distributed Interactive Simulation Protocol Data Units expressed in Ada code	115
6.7.1	Basic data types and records	115
6.7.1.1	Angles	115
6.7.1.2	Angular Velocity Vector Record	115
6.7.1.3	Articulated Part Record	115
6.7.1.4	Boolean	116
6.7.1.5	Burst Descriptor Record	116
6.7.1.6	Date/Time data type	116
6.7.1.7	Entity Capabilities Record	116
6.7.1.8	Entity Identifier Record	116
6.7.1.9	Entity Marking	117
6.7.1.10	Entity Type	117
6.7.1.11	Euler angles: Representation of entity orientation	118
6.7.1.12	Event Identifier Record	118
6.7.1.13	Exercise identifier	118
6.7.1.14	Organizational Unit Record	118
6.7.1.15	Radar system	119
6.7.1.16	Repair type	119
6.7.1.17	Service type	119
6.7.1.18	Supply Quantity Record	119
6.7.1.19	Terrain database identifier	120
6.7.1.20	Time stamp: Representation of time	120
6.7.1.21	Vector Record	120
6.7.1.22	World Coordinates Record	120
6.7.2	Distributed Interactive Simulation Protocol Data Units	121
6.7.2.1	PDU header	121
6.7.2.2	Entity State PDU	121
6.7.2.3	Fire PDU	122
6.7.2.4	Detonation PDU	122
6.7.2.5	Update Threshold Request PDU	122
6.7.2.6	Update Threshold Response PDU	123
6.7.2.7	Service Request PDU	123
6.7.2.8	Resupply Offer PDU	123
6.7.2.9	Resupply Received PDU	124
6.7.2.10	Resupply Cancel PDU	124

C O N T E N T S

PARAGRAPH		PAGE
6.7.2.11	Repair Complete PDU	124
6.7.2.12	Repair Response PDU	124
6.7.2.13	Collision PDU	125
6.7.2.14	Radar PDU	125
6.7.2.15	Interim PDUs	126
6.7.2.15.1	Activate Request PDU	126
6.7.2.15.2	Activate Response PDU	127
6.7.2.15.3	Deactivate Request PDU	127
6.7.2.15.4	Deactivate Response PDU	128
6.7.2.16	Recommended PDU: The Emitter PDU	128
Appendix A		
The Open Systems Interconnection Reference Model		130
10.1	Scope	130
30.1	OSI Reference Model	130
30.2	Layer functions in the OSI Reference Model	132
Appendix B		
Fuze, Warhead and Supply Types		133
10.1	Scope	133
30.1	Fuze and Warhead	133
30.2	Supply types	135
Appendix C		
Defined Repair Types		136
10.1	Scope	136
30.1	General repair codes	136
30.2	Subsystems	136
30.2.1	Propulsion systems.	136
30.2.2	Hull/Airframe/Body.	137
30.2.3	Interface to environment systems.	137
30.2.4	Weapon systems.	137
30.2.5	Fuel systems.	137
30.2.6	Electronic and communications systems.	138
30.2.7	Life support systems.	138
30.2.8	Hydraulics systems and actuators.	138
30.2.9	Auxiliary craft.	139
Appendix D		
Appearance Field Bit Definition		140
10.1	Scope	140
30.1	Platforms	140
30.1.1	Platforms of the land domain	140
30.1.2	Platforms of the Air Domain	141
30.1.3	Platforms of the Surface Domain	142
30.1.4	Platforms of the Sub-surface Domain	142
30.1.5	Platforms of the Space Domain	143

C O N T E N T S

PARAGRAPH	PAGE
30.2 Munitions	143
30.2.1 Guided munitions	143
30.3 Life forms	144
30.3.1 Life forms	144
30.4 Environmentals	144
30.4.1 Environmentals	144
30.5 Cultural features	145
30.5.1 Cultural features	145
Appendix E	
Organizational Unit	146
10.1 Scope	146
30.1 Hierarchy of units	146
30.1.1 Force ID	146
30.1.2 Country ID	146
30.1.3 Service ID	147
30.1.4 Hierarchy	147
30.1.4.1 Army	147
30.1.4.2 Air Force	148
30.1.4.3 Coast Guard	148
30.1.4.4 Marines	148
30.1.4.5 Navy	149
Appendix F	
Enumeration and Bit-encoded Values	150
10.1 Scope	150
30.1 Enumeration values and bit-encoded values for DIS required PDUs	150
30.1.1 Appearance field bit definitions	150
30.1.2 Country ID	150
30.1.3 Detonation result	154
30.1.4 Entity marking	155
30.1.5 Organizational unit values	155
30.1.6 PDU kinds	155
30.1.6.1 Reserved values	155
30.1.6.2 Defined values	155
30.1.6.3 Temporary	155
30.1.7 Radar mode	156
30.1.8 Radar system category	156
30.1.9 Radar system subcategory	156
30.1.10 Radar system ID	157
30.1.11 Repair types	157
30.1.12 Result of Repair	157
30.1.13 Service type	158
30.1.14 Update change result	158

C O N T E N T S

PARAGRAPH	PAGE
30.2 Enumeration values and bit-encoding for interim and proposed PDUs	158
30.2.1 Activate reason	158
30.2.2 Activate result	158
30.2.3 Deactivate reason	159
30.2.4 Deactivate reason	159
30.2.5 Emitter data base number	159
30.2.6 Emitter class	159
Appendix G	
Articulated Parts	161
10.1 Scope	161
30.1 Articulation parameters	161
30.1.1 Elevation	161
30.1.2 Azimuth	161
30.1.3 Extension	161
30.1.4 Position	161
30.1.5 Station	162
30.2 Guidelines for choosing parameters	162
30.3 Articulation Parameters	162
30.3.1 Parameter types	163
30.3.2 Specific attached and articulated parts	164
30.3.2.1 Attached part stations	164
30.3.2.2 Articulating parts	164
30.4 Example #1	170
30.4.1 Parameter types	170
30.5 Example #2	171
Appendix H1	
Entity Types	173
10.1 Scope.	173
30.1 Entity Kind.	173
30.1.1 Platform Kind.	173
30.1.1.1 Domain.	173
30.1.1.2 Country.	174
30.1.1.3 Specific Platform Definition.	174
30.1.2 Munition Kind.	174
30.1.2.1 Domain.	174
30.1.2.2 Country.	174
30.1.2.3 Specific Munition Definition.	175
30.1.2.4 Fuze and Warhead.	175
30.1.3 Life Form Kinds.	175
30.1.3.1 Domain.	175
30.1.3.2 Country.	175
30.1.4 Environmental Kinds.	178

C O N T E N T S

PARAGRAPH		PAGE
30.1.4.1	Domain.	178
30.1.4.2	Country.	178
30.1.4.3	Specific Environmental Definition.	178
30.1.5	Cultural Feature Kind.	180
30.1.5.1	Domain.	180
30.1.5.2	Country.	180
30.1.5.3	Specific Cultural Feature Definition.	180

Appendix H2

LIST OF FIGURES

FIGURE		PAGE
4-1	The behavior of the requesting host computer	19
4-2	The behavior of the responding host computer	22
4-3	The behavior of the receiver during resupply service	28
4-4	The behavior of the supplier during resupply service	31
4-5	The behavior of the receiver during repair service	33
4-6	The behavior of the repairing host computer during repair service	35
5-1	Byte ordering	43
5-2	Angular Velocity Vector Record	44
5-3	Parts Parameter Record	45
5-4	Articulated Parts Record	46
5-5	Burst Descriptor Record	47
5-6	Entity Capabilities Record	47
5-7	Simulation Address Record	48
5-8	Entity Identifier Record	49
5-9	Entity Marking Record	49
5-10	Entity Type Record	51
5-11	Euler Angles Record	51
5-12	Event Identifier Record	52
5-13	Organizational Unit Record	53
5-14	Radar system data type	54
5-15	Supplying Quantity Record	54
5-16	Terrain Database Identifier Record	55
5-17	Vector Record	56
5-18	World Coordinates Record	56
5-19	PDU header	59

LIST OF FIGURES

FIGURE		PAGE
5-20	Entity State PDU	62
5-21	Fire PDU	67
5-22	Detonation PDU	70
5-23	Update Threshold Request PDU	73
5-24	Update Threshold Response PDU	75
5-25	Service Request PDU.	77
5-26	Resupply Offer PDU	79
5-27	Resupply Received PDU.	81
5-28	Resupply Cancel PDU.	82
5-29	Repair Complete PDU.	83
5-30	Repair Response PDU.	85
5-31	Collision PDU.	87
5-32	Radar PDU.	90
6-1	Activate Request PDU	101
6-2	Activate Response PDU.	105
6-3	Deactivate Request PDU	106
6-4	Deactivate Response PDU.	107
6-5	Emitter Type Record.	108
6-6	Emitter PDU.	111
A-1	Open Systems Interconnection Reference Model	131

LIST OF TABLES

TABLE		PAGE
I	Functions of the OSI Layers.	132
II	Electromagnetic Spectrum: Emitter Classes.	160

1. SCOPE

1.1 Scope. This standard establishes the requirements for entity information and entity interaction protocol data units exchanged between simulators participating in a distributed interactive simulation. It is the first in a series of standards being developed to address the problem of interoperability among interconnected simulators. These standards encompass a portion of the application layer of a communications architecture as defined by the International Organization for Standardization's (ISO) Open Systems Interconnection (OSI) Reference Model (see Appendix A).

1.2 Application. When invoked in a specification or statement of work, these requirements will apply to simulation devices intended for participation in a Distributed Interactive Simulation (DIS). The contractor is responsible for invoking all the applicable requirements of this Military Standard on any and all subcontractors he may employ.

2. APPLICABLE DOCUMENTS

2.1 Government documents.

2.1.1 Specifications, standards, and handbooks.

MILITARY

MIL-STD-1777 - Internet Protocol Specification, Appendix A.

2.2 Non-Government publications.

STANDARDS

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION (ISO)

ISO 7498 - OSI Reference Model

INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS(IEEE)

IEEE 754-1985 - Standard for floating point numbers

NON-STANDARD PUBLICATIONS

SIMNET

BBN RPT. 7102 - The SIMNET Network and Protocols

INTEROPERABILITY MEETINGS

IST-CF-89-1 - Summary Report: The First Conference on Standards for the Interoperability of Defense Simulations

IST-CF-90-01 - Summary Report: The Second Conference on Standards for the Interoperability of Defense Simulations

IST-CR-90-13 - Summary Report: The Third Workshop on Standards for the Interoperability of Defense Simulations

ACCOMPANYING DOCUMENTS

IST-PD-90-1 - Rationale Document for Entity Interaction Protocol for Distributed Interactive Simulation

3. DEFINITIONS

3.1 Acronyms used in this standard. The acronyms used in this standard are defined as follows:

ANSI	-	American National Standards Institute
ASCII	-	American Standard Code for Information Interchange
ASW	-	Anti-Submarine Warfare
BAM	-	Binary Angle Measurement
CIG	-	Computer Image Generator
CWI	-	Continuous Wave Illuminations
C3	-	Command, Control and Communications
DARPA	-	Defense Advanced Research Projects Agency
DIS	-	Distributed Interactive Simulation
ECM	-	Electronic Countermeasures
EHF	-	Extremely High Frequency
EO	-	Electro-Optical
ESM	-	Electronic Warfare Support Measures
EW	-	Electronic Warfare
HF	-	High Frequency
IEEE	-	Institute of Electrical and Electronic Engineers
IFF	-	Identify Friend or Foe
IR	-	Infrared
ISO	-	International Organization for Standardization
IST	-	Institute for Simulation And Training
ITD	-	Interim Terrain Database

LAN	-	Local Area Network
LF	-	Low Frequency
OSI	-	Open Systems Interconnection
PDU	-	Protocol Data Unit
PM TRADE	-	Army Project Manager for Training Devices
PRF	-	Pulse Repetition Frequency
SAFOR	-	Semi-Automated Forces
SAM	-	Surface To Air Missile
SHF	-	Super High Frequency
SIMAN	-	Simulation Management Protocol
UHF	-	Ultra High Frequency
UTSS	-	Universal Threat Simulation System
VLF	-	Very Low Frequency
WAN	-	Wide-Area Network

3.2 Application layer. The layer of the ISO reference model which provides the means for user application processes to access and use the network's communications resources.

3.3 Articulated Part. A visible part of a simulated entity that is able to move relative to the entity itself.

3.4 Ballistic munition. Any munition that follows a predetermined ballistic trajectory.

3.5 Binary Angle Measurement (BAM). A method of angle measurement which uses the binary number system. Two types of angles are used in this standard:

a. 32-bit BAMs- 32-bit BAMs are used to express the orientation of an entity. Each unit of BAM is equal to $360/2^{32}$ degrees or $2*pi/2^{32}$ radians.

b. 16-bit BAMs- 16-bit BAMs are used to express the orientation of articulated parts. Each unit of BAM is equal to $360/2^{16}$ degrees or $2*pi/2^{16}$ radians.

3.6 Bit. The smallest unit of information in the binary system of notation.

3.7 Broadcast. An addressing mode in which a PDU of a single application process is sent to all nodes on a network.

3.8 Byte. A sequence of eight consecutive bits operated upon as a unit.

3.9 Connectionless service. A message service which provides a mode of information transfer between peer entities in which each data transfer is independent of and not coordinated with previous or subsequent transfers and in which no state information is maintained.

3.10 Data link layer. The layer of the ISO reference model which provides the functional and procedural means to transfer data between stations, and to detect and correct errors that may occur in the physical layer.

3.11 Dead reckoning. A methodology for the estimation of the position of an entity or object based on a previously known position and estimates of time and motion.

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

3.13 Distributed Simulation. See Distributed Interactive Simulation.

3.14 Emitter. A device that is able to discharge detectable electromagnetic energy.

3.15 Entity. An element of a simulated world (such as a vehicle) that is generated and controlled by one or more simulators. An entity may also be an element of the simulated world, such as cultural features including buildings and bridges, that may be subject to changes in appearance as a result of the simulation exercise.

3.16 Entity coordinates. Location with respect to a simulated entity is described by an entity coordinate system. This right-handed Cartesian coordinate system has its origin at the center of the bounding volume of the entity. The positive direction of each axis extends out the front (x-axis), the right side (y-axis), and downward (z-axis).

3.17 Euler angles. A set of three angles used to describe the orientation of an entity as a set of three successive rotations about three different orthogonal axes (x, y, and z), using the right hand rule: first counterclockwise about z by angle ψ (psi), then counterclockwise about the new y by angle θ (theta), then counterclockwise about the newest x by angle ϕ (phi). Angles ψ and ϕ range between $\pm\pi$, while angle θ ranges only $\pm\pi/2$ radians. These angles specify the successive rotations needed to transform from the world coordinate system to the entity coordinate system.

3.18 Exercise. See Simulation Exercise.

3.19 Fidelity. A measure of the realism of a simulation.

3.20 Fields. A series of contiguous bits treated as an instance of a particular data type that may be part of a higher level data structure.

3.21 Guided munition. Any munition that responds to steering commands after launch and does not follow a predetermined ballistic trajectory.

3.22 Host or Host computer. A computer attached to a network.

3.23 Layer. A well-defined, logical and hierarchical subdivision of a network architecture. Each layer can use the services of the next lower layer, plus its own functions, to create services which are made available to the next higher layer.

3.24 Layering. A structuring technique in which the communication functions are partitioned into a hierarchical set of layers. Each layer performs a related subset of functions required to communicate with another system.

3.25 Local Area Network (LAN). A communications network designed for a moderate size geographic area and characterized by moderate to high data transmission rates, low delay, and low bit error rates.

3.26 Long-Haul network. See Wide Area Network.

3.27 Multicast. An addressing mode in which a PDU of a single application process is sent to a subset of the nodes on a network.

3.28 Network architecture. The organization of system components, functions that these components perform, and their interrelationships; a specification which defines how a system is to be organized. This specification defines functional modularity as well as protocols and interfaces which allow communication and cooperation among those modules.

3.29 Network management. The collection of administrative structures, policies and procedures which collectively provide for the management of the organization and operation of the network as a whole.

3.30 Node. A device which communicates via a Local Area Network or a Wide Area Network. A point in a network where one or more communications lines terminate.

3.31 Octet. A sequence of eight bits, usually operated upon as a unit.

3.32 Open system architecture. A communications architecture standard developed by ISO.

3.33 OSI Reference Model. A model that organizes the data communication concept into seven layers and defines the functionality of each layer. OSI Reference Model for Network Architecture was created by ISO and is used as a reference to compare network architectures.

3.34 Protocol Data Unit (PDU). A portion of the application level data that is passed on a network between application processes.

3.35 Physical layer. The layer of the ISO reference model which provides the mechanical, electrical, functional, and procedural characteristics to access the transmission medium.

3.36 Presentation layer. The layer of the ISO reference model which frees the application processes from concern with differences in data representation.

3.37 Protocol. A formal set of conventions or rules governing the format, timing, and error control to facilitate message exchange between communicating processes.

3.38 Real-Time. An event or data transfer which must be accomplished within an allotted amount of time or when the accomplishment of the action has either no or diminishing value.

3.39 Session layer. The layer of the ISO reference model which provides the mechanisms for organizing and structuring the interaction between two entities.

3.40 Simulated entity. See Entity.

3.41 Simulation exercise. An activity which uses computers to train individuals in certain skills or to evaluate operational equipment within a given scenario.

3.42 Simulation management. A mechanism that provides centralized control of the simulation exercise. Functions of simulation management include: Start, Restart, Maintenance, and Shutdown of the exercise. Other functions provided include introduction of late players and the collection and distribution of certain types of data.

3.43 Simulation manager computer. The simulation manager computer is a host computer that exists on each LAN participating in a DIS. This host computer is responsible for performing simulation management functions for that LAN. Specific requirements for the simulation management computer are not completely specified by this standard although certain functions are specified as the responsibility of the simulation manager computer.

3.44 Transport layer. The layer of the ISO model which accomplishes the transparent transfer of data over an established communications connection, providing an end-to-end service with high data integrity.

3.45 Wide-Area Network (WAN). A communications network designed for large geographic areas. Sometimes called Long-Haul Network.

4. GENERAL REQUIREMENTS

4.1 Introduction. This section contains requirements concerning the content and use of PDUs in DIS exercises.

4.2 Authority in charge of DIS exercises. As of August, 1990, no organization has been assigned the duty of serving as the authority in charge of DIS exercises. Until such an organization is assigned, the Army Project Manager for Training Devices (PM TRADE) shall serve as a temporary authority. This authority shall perform administrative functions such as maintaining and distributing databases, allocating site identifiers and configuration control of this standard.

4.2.1 Site manager. At each designated DIS site, a site manager shall oversee DIS operations at that site.

4.2.2 Simulation Manager Computer. The Simulation Manager Computer shall be a host computer existing on each LAN. This computer shall be responsible for establishing the exercise for the LAN to which it belongs. The simulation manager computer for each LAN is designated by the site manager for that LAN.

4.3 DIS exercise. A DIS exercise shall be a set of interacting host computers which shall share one identifying number called the Exercise Identifier.

4.4 The issuing entity. In a DIS exercise, host computers may represent one or more entities. The host computer shall issue PDUs for each of the entities that it simulates. It is, therefore, not the entity that issues PDUs but the host computer that represents that entity. For reference, it is easier to say that the entity is issuing the PDU rather than saying that the host computer that is simulating the entity is issuing the PDU. Therefore, the statement "entity issues" shall mean "the host computer representing the entity issues."

4.5 Issuance of PDUs. PDUs shall be issued according to the guidelines specified in the paragraphs that follow.

4.6 Receipt of PDUs. Upon receipt of PDUs, host computers shall act as specified in the paragraphs that follow. Unless otherwise stated, the actions described in the paragraphs to follow that refer to the receiving entity shall apply to the entity to which the PDU is addressed.

4.7 Protocol data units for DIS. The following paragraphs shall establish the content and the procedure for use of PDUs in a DIS exercise.

4.7.1 Protocol data unit header. A protocol data unit (PDU) header shall be the first part of each protocol data unit. The header shall specify the identification number associated with the DIS exercise, the protocol version, and the type of protocol data unit that follows (see also 5.3.3).

4.7.1.1 Protocol version. The PDU header shall specify the version of DIS protocol to which the PDU pertains.

4.7.1.2 DIS exercise identification. Each DIS exercise shall be distinguished from other exercises by the use of an Exercise Identifier. This identifier shall be assigned by the organization that establishes the exercise. An identifier shall be assigned that is currently not in use on the network on which the exercise is communicated.

4.7.1.3 Types of PDUs. There shall be 13 PDUs required by this standard. PDUs in this standard shall be defined as follows:

- Entity State PDU
- Fire PDU
- Detonation PDU
- Update Threshold Request PDU
- Update Threshold Response PDU
- Service Request PDU
- Resupply Offer PDU
- Resupply Received PDU
- Resupply Cancel PDU
- Repair Complete PDU
- Repair Response PDU
- Collision PDU
- Radar PDU

In addition to these PDUs, other PDUs may be added to this standard as the standard develops. Type definitions have been reserved for this type of protocol growth. Type definitions in the standard have also been reserved for implementation of experimental PDUs.

4.7.2 Entity Information. Information associated with the appearance and location of an entity shall be communicated in a DIS exercise through the use of an Entity State PDU (ES PDU) (see also 5.3.4).

4.7.2.1 Entity State PDU. The Entity State PDU (ES PDU) shall communicate information about an entity's state. This includes state information that is necessary for the receiving host computers to represent the issuing entity in the host computer's own simulation.

4.7.2.1.1 Information contained in the Entity State PDU. The ES PDU shall contain the following information:

a. The identification of the entity that issued the PDU.

b. The issuing entity's specific entity type.

c. The time at which the data in the PDU is valid.

d. Information about the location of the entity in the simulated world and its orientation. This information includes:

- The location with respect to the world
- The velocity (the rate at which its location is changing)
- Acceleration (the rate at which its velocity is changing)
- Orientation
- The angular velocity (or the rate at which its orientation is changing)
- The dead reckoning parameters that should be employed when extrapolating the position of this entity. Values for this field are currently undefined.

e. The information required for representation of the entity's visual appearance. This information includes:

- The appearance of the entity (for example, normal, smoking, on fire, producing a dust cloud, and so forth.)
- Markings
- The number and orientation of articulated parts
- The presence of attached parts or stores

f. The capabilities of the entity. Defined capabilities include:

- Resupply
- Repair

4.7.2.1.2 Issuance of the Entity State PDU. The ES PDU shall be issued by an entity when one or more of the following conditions exist:

a. The discrepancy between an entity's actual state (as determined by its own high fidelity model) and its dead reckoned state (state using specified dead reckoning algorithms) exceeds a predetermined threshold. (see 4.7.4 and 4.7.5.1 concerning dead reckoning and threshold values). This threshold includes changes in position/orientation information and articulated parts information.

b. A change in the entity's appearance occurs. This change may include being on fire or smoking.

c. A predetermined length of time has elapsed since the issuing of the last ES PDU. If no value is established, the default value for this time shall be five seconds. This value shall be established at exercise start by the simulation manager computer (see 3.43), or may be changed by the simulation manager computer during the exercise.

4.7.2.1.3 Receipt of the Entity State PDU. Upon receipt of the ES PDU, a host computer shall use the information contained therein to model the position and appearance of the entity that issued the PDU.

4.7.3 Weapons fire. Information associated with the representation of weapons effects in a DIS exercise shall be accomplished through the use of two PDUs: The Fire PDU and the Detonation PDU (see also 5.3.5.1).

4.7.3.1 Representation of weapons fire in DIS. Representation of weapons fire in a DIS exercise shall consist of the following sequence of events:

a. An entity fires a weapon. The firing of a weapon shall be communicated through the use of a Fire PDU.

b. The munition launched shall be modeled by the firing entity's host computer. If the munition is a guided munition, it shall be assigned a unique identification number in the Fire PDU and the host computer modeling its behavior shall issue ES PDUs for the munition according to the procedures for the use of the ES PDU (see 4.7.2). The munition, therefore, is represented as an entity. If the munition is not a guided munition

(for example, a ballistic munition), the field that contains the munition identification number in the Fire PDU shall contain the value zero and no ES PDUs shall be issued for the munition.

c. The impact or detonation of a munition represents the end of its path and the end of the existence of a munition entity. This event shall be represented by a Detonation PDU. A Detonation PDU shall also be used to terminate sky shots (see 4.7.3.3.5) that do not impact or detonate.

4.7.3.2 Fire PDU. The Fire PDU shall be used to communicate information associated with the firing of a weapon.

4.7.3.2.1 Information Contained in the Fire PDU. The Fire PDU shall contain the following information:

a. The identification of the entity issuing the PDU.

b. The identification of the intended target entity if known to the host computer, zero otherwise.

c. The identification of guided munitions (munitions not guided have an identification value of zero).

d. The identification of the specific event marked by the firing of an entity's weapon.

e. The time at which the data in the PDU is valid.

f. The information required for representation of the path and impact of the munition. This information includes:

- The location from which the munition was launched
- The type of munition fired
- The warhead of the munition (if applicable, zero otherwise)
- The fuze employed by the munition (if applicable, zero otherwise)
- The quantity and rate at which it was fired
- The initial velocity of munition fired
- The range the firing entity's fire control system has assumed for computing the fire control solution

4.7.3.2.2 Issuance of the Fire PDU. The Fire PDU shall be issued by an entity at the moment it fires a weapon.

4.7.3.2.3 Single rounds and bursts of fire. If the firing of the weapon represents a single round, the quantity field of the Fire PDU shall contain the value one, and the rate field shall contain the value zero. Otherwise the fields shall contain the quantity of munition fired and the rate at which it was fired, respectively.

4.7.3.2.4 Receipt of the Fire PDU. Upon receipt of a Fire PDU, a host computer shall use the information therein to represent any necessary visual and aural effects produced by the firing of the weapon, whether it be a muzzle flash, noise, or smoke.

4.7.3.3 Detonation PDU. The Detonation PDU shall be used to communicate information associated with the impact or detonation of a munition.

4.7.3.3.1 Information contained in the Detonation PDU. The Detonation PDU shall contain the following information:

a. The identification of the entity issuing the PDU.

b. The identification of the target entity if it impacted an entity, zero otherwise.

c. The identification of guided munitions (munitions not guided have an identification value of zero).

d. The identification of the event. This number shall be the same as the event identification number assigned to the corresponding Fire PDU.

e. The time at which the data in the PDU is valid.

f. The information required for representation of the impact or detonation of the munition. This information includes:

- The location with respect to the world
- The type of munition fired
- The warhead of the munition (if applicable, zero otherwise)
- The fuze employed by the munition (if applicable, zero otherwise)
- The quantity and rate at which it was fired
- The velocity just before impact
- The energy of detonation
- The directionality of the detonation
- The momentum
- The location of detonation with respect to

the target entity
- The detonation result (detonation, impact, no detonation or impact)

4.7.3.3.2 Issuance of the Detonation PDU. The Detonation PDU shall be issued by a host computer at the moment that a munition being modeled by that host computer impacts or detonates. If the munition neither impacts nor detonates (sky shot) the controlling host computer shall issue a Detonation PDU to indicate when it has ceased to model the munition.

4.7.3.3.3 Inclusion of the Entity Identifier. If the impact or detonation is known to have affected only a specific entity (for example, by impacting it), the firing entity shall communicate the Entity Identifier of the affected entity in the Detonation PDU. The location of the impact or detonation in the entity coordinates of the affected entity shall also be included. Otherwise, all fields of the Entity Identifier shall contain zeros. In both cases, the location of the impact or detonation in world coordinates shall be communicated.

4.7.3.3.4 Termination of the existence of munition entities. The Detonation PDU shall indicate the termination of the existence of a munition entity. Upon receipt of the Detonation PDU, host computers shall stop modeling the munition.

4.7.3.3.5 Sky shots. If the munition neither impacts nor detonates (sky shot), the controlling host computer shall issue a Detonation PDU to indicate when it has ceased to model the munition. For example, if the munition has exceeded a certain range, the controlling host computer shall issue a Detonation PDU and shall cease to model that munition. In the case of a sky shot, the location field of the Detonation PDU shall contain the location of the munition entity when the controlling host stopped modeling the munition.

4.7.3.3.6 Receipt of the Detonation PDU. Upon the receipt of a Detonation PDU, a host computer shall use the information therein to represent the visual and aural effects that may be produced by the detonation or impact of the munition. The receiving host computer shall also use the information to determine damage that may have been received as a result of the detonation.

4.7.4 Dead reckoning. A method of estimation called dead reckoning shall be employed to limit the rate at which ES PDUs are issued. By estimating the position/orientation of other entities, it is not necessary for an entity to receive a report about every change in position/orientation that occurs in the entities it is dead reckoning. Only when a change in

position/orientation differs a certain amount from the dead reckoned position/orientation is a new update of position required.

4.7.4.1 Dead reckoning and the issuing entity. Each host computer shall maintain a high fidelity model of itself (representing its actual position) and a lower fidelity, dead reckoned, model of itself. Certain thresholds shall be established as criteria for determining if the entity's actual position/orientation has varied from its dead reckoned position/orientation. When the entity's actual position/ orientation has varied from the dead reckoned position/ orientation by more than the threshold value, the entity shall issue an ES PDU to communicate to other host computers its actual position. The entity shall also use the same information communicated to other host computers to update its own dead reckoning model of itself.

4.7.4.2 Dead reckoning and the receiving entity. Each host computer shall also maintain a dead reckoned model of the position of entities that are of interest (within sight or range). An entity shall also dead reckon the orientation of other entities when specified by the dead reckoning model in use. The dead reckoned position/orientation of other entities shall be used to display their positions in an entity's visual or sensor displays. When the entity receives a new update from one of the entities it is dead reckoning, it shall correct its dead reckoning model and base its estimations on the most recent position, velocity, and acceleration information. Smoothing techniques may be used to eliminate jumps that may occur in a visual display when the dead reckoned position of an entity is corrected to the most recently communicated position.

4.7.4.3 Dead reckoning algorithms. Proposed dead reckoning algorithms for use with this standard are included in the Rationale Document that pertains to this standard (see 2.2). To allow parameters associated with dead reckoning, a field has been set aside in the Entity State PDU for dead reckoning parameters.

4.7.5 Update Threshold Control. Information associated with control of the threshold value that determines the issuing of ES PDUs shall be communicated in a DIS exercise through the use of two PDUs: the Update Threshold Request PDU, and the Update Threshold Response PDU (see also 5.3.5.2).

4.7.5.1 Threshold values. Threshold values in relation to the ES PDU shall be based on the distance that an entity's dead reckoned position varies from its actual position, or the angle the entity's dead reckoned orientation varies from its actual orientation. Position thresholds are in the x, y and z

directions. Orientation thresholds are angles about each of the entity's three axes of rotation. The smaller or tighter the threshold value the more often a host computer is likely to issue an ES PDU. The initial threshold values shall be established at the start of the DIS exercise.

4.7.5.2 Changing update threshold values. In a DIS exercise, it is sometimes desirable for a host computer to maintain a more accurate state model of certain entities. To maintain this model, it is necessary to obtain a more accurate estimation of the other entity's position/orientation. This goal may be achieved by reducing the threshold values used to determine when ES PDUs are issued. Reduced threshold values would cause an entity to report changes in its actual position from its dead reckoned position sooner. In most cases, this would increase the rate at which ES PDUs are issued by the entity. At other times, it may be necessary for all host computers to increase the threshold values in order to produce a lower update rate and reduce network traffic. Exercise-wide permanent threshold changes shall be the responsibility of the simulation manager computer.

4.7.5.3 State information for Update Threshold Control. The following paragraphs describe the different states and transitions for Update Threshold Control. An example of Update Threshold Control is given in 6.6.

4.7.5.3.1 The requesting host computer. The requesting host computer may be in one of two states:

Ready State	A requesting host computer is in the Ready State when it is not in the Requesting State.
Requesting State	A requesting host computer is in the Requesting State when it has requested new threshold values and has not received a reply to its request.

The behavior of the requesting host computer is shown on Figure 4-1.

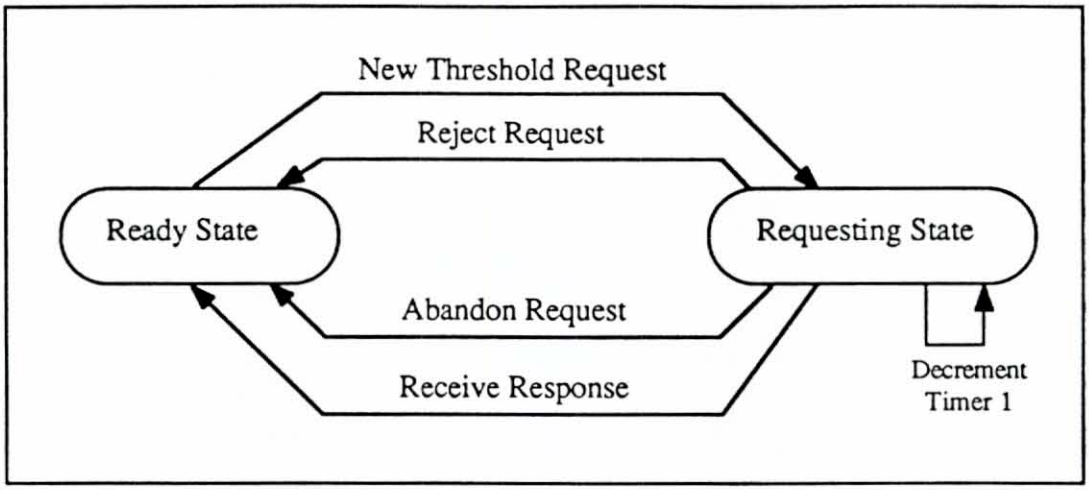


FIGURE 4-1. Requestor behavior during Update Rate Control.

Transition	Condition and Actions
New Threshold Request	When a request for change has been issued using the Update Threshold Request PDU (see 4.7.5.5), the entity shall proceed from the Ready State to the Requesting State. Timer 1 is set for five seconds.
Reject Request	When the entity has received an Update Threshold Response PDU (see 4.7.5.6) in response to its request rejecting the requested change, the entity shall proceed from the Requesting State to the Ready State.
Abandon Request	When Timer 1 expires, the request shall be abandoned and the entity shall proceed from the Requesting State to the Ready State.
Receive Response	When the entity has received an Update Threshold Response PDU in response to its request, the entity shall proceed from the Requesting State to the Ready State.

4.7.5.3.2 The responding host computer. The responding host computer may be in one of three states:

Default Threshold State	A responding host computer is in the Default Threshold State if it has not been requested to change the threshold values established by the simulation manager computer.
Altered Threshold State	A responding host computer is in the Altered Threshold State if it has received a request for smaller thresholds and is implementing the request.
Hold Altered Threshold State	A responding host computer is in the Hold Altered Threshold State if the time for an altered threshold has expired and the host computer has not received another request to implement a threshold value less than the default. The responding host computer shall not remain in this state for more than five seconds.

The behavior of the responding host computer is shown on Figure 4-2.

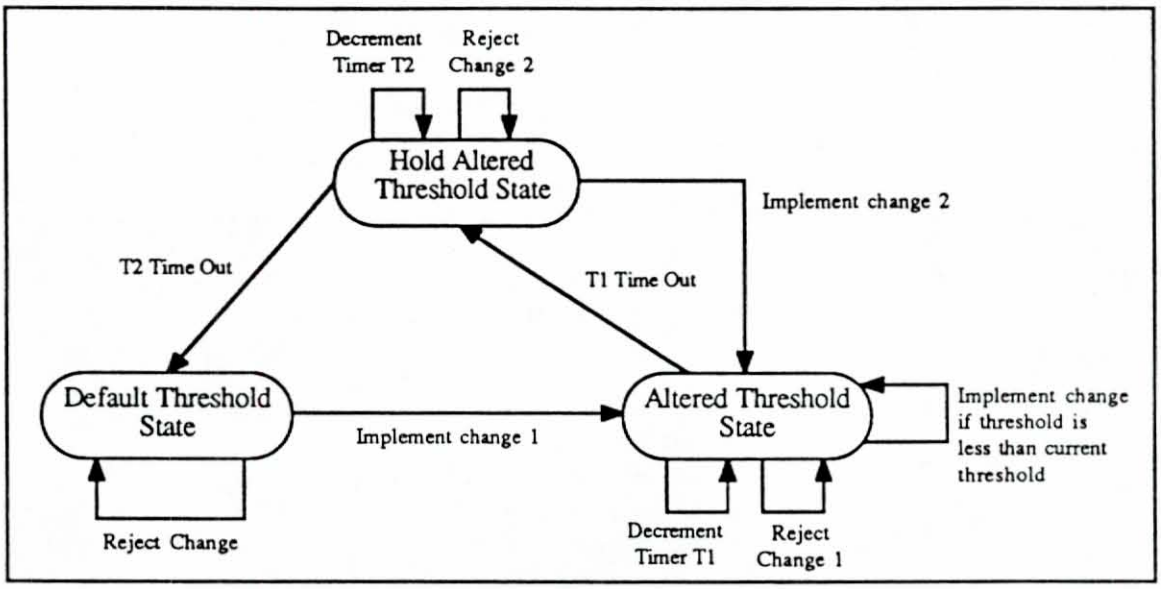


FIGURE 4-2. Responder behavior during Update Rate Control.

Transition	Condition and Actions
Implement Change 1	When an Update Threshold Request PDU (4.7.5.5) is received, if the requested threshold value is less than the default value, the entity shall implement the new thresholds and shall proceed from the Default Threshold State to the Altered Threshold State. Timer 1 shall be set to the minimum of time specified in the Update Threshold Request PDU and three minutes.
Reject Change 1	When an Update Threshold Request PDU is received, if the requested threshold value is rejected, the entity shall issue an Update Threshold Response PDU and shall remain in the Altered Threshold State.
Implement Change 2	When an Update Threshold Request PDU is received, if the requested threshold value is less than the default threshold value, the entity shall implement the new thresholds and shall proceed from the Hold Altered Threshold State to the Altered Threshold State. Timer 1 is set to the minimum of time specified in the Update Threshold Request PDU and three minutes.
Reject Change 2	When an Update Threshold Request PDU is received, if the requested threshold value is rejected, the entity shall issue an Update Threshold Response PDU and shall remain in the Hold Altered Threshold State.
Timer 1 Expires	When the entity is in the Altered Threshold State and Timer 1 expires, the entity shall proceed to the Hold Altered Threshold State. Timer 2 shall be set to five seconds.
Timer 2 Expires	When the entity is in the Hold Altered Threshold State and Timer 2 expires, the entity shall proceed to the Default Threshold State.

4.7.5.4 Representation of Update Threshold Control in DIS. Representation of Update Threshold Control shall consist of the following sequence of events:

a. An entity desires a more accurate estimation of another entity's position and/or orientation. The first entity shall be called the Requestor. The second entity shall be called the Responder.

b. The Requestor shall issue an Update Threshold Request PDU, addressed to the Responder, requesting a change in threshold values. The Requestor then proceeds from the Ready State to the Requesting State. The length of time over which the new thresholds are to remain in effect shall also be specified. This time shall not exceed three minutes. The Requestor may continue to issue requests until a response is received from the Responder.

c. The Responder, upon reception of the Update Threshold Request PDU, shall issue an Update Threshold Response PDU to indicate to the Requestor whether or not it is able to implement the new thresholds. If the Responder intends to change its thresholds, it shall proceed from the Ready State to the Altered Threshold State and shall change its update thresholds. The Responder shall maintain these new thresholds until the period of time specified by the Update Threshold Request PDU or three minutes has expired. If the Responder does not intend to change its thresholds because tighter thresholds are already implemented, the responder shall specify the time remaining in the use of current thresholds in the Update Threshold Response PDU. The Requestor shall not issue another request to this particular Responder until the remaining time expires.

d. When the Requestor receives the Update Threshold Response PDU from the Responder, the Requestor shall proceed from the Requesting State to the Ready State.

e. When the specified time period has expired, the Responder proceeds from the Altered Threshold State to the Hold Altered Threshold State. The Responder shall remain in this state for five seconds or until any incoming requests to maintain thresholds smaller than the default threshold values are received. If five

seconds have expired and no request is received, the Responder shall proceed to the Default Threshold State. If another request is received and accepted prior to the end of the five second time period, the Responder shall return to the Altered Threshold State.

4.7.5.5 Update Threshold Request PDU. The Update Threshold Request PDU shall be used when a host computer wishes to request a change in the thresholds governing the rate at which ES PDUs are issued.

4.7.5.5.1 Information contained in the Update Threshold Request PDU. The Update Threshold Request PDU shall contain the following information:

a. The identification of the entity that issued the PDU.

b. The identification of the entity that is being requested to change its update threshold values.

c. The new threshold values:

- Linear thresholds shall be specified as length in meters relative to the entity's coordinate system.

- Rotational thresholds shall be measured in BAMs.

d. The duration of the change.

4.7.5.5.2 Issuance of the Update Threshold Request PDU. The Update Threshold Request PDU shall be issued by an entity at the time a change in the update thresholds is desired.

4.7.5.5.3 Receipt of the Update Threshold Request PDU. Upon receipt of an Update Threshold Request PDU, the entity shall respond immediately with an Update Threshold Response PDU.

4.7.5.6 Update Threshold Response PDU. The Update Threshold Response PDU shall be used to communicate a response to the receipt of an Update Threshold Request PDU.

4.7.5.6.1 Information contained in the Update Threshold Response PDU. The Update Threshold Response PDU shall contain the following information:

a. The identification of the entity issuing the PDU.

b. The identification of the entity to which the PDU is addressed.

c. The response to the request for the update change.

d. If the request is not implemented, then the length of time remaining for the current threshold value shall be given. If the request is implemented, the length of time specified in the request shall be given.

4.7.5.6.2 Issuance of the Update Threshold Response PDU. The Update Threshold Response PDU shall be issued by an entity receiving an Update Threshold Request PDU.

4.7.5.6.3 Receipt of the Update Threshold Response PDU. Upon receipt of an Update Threshold Response PDU, the receiving host computer shall note the response of the sending host computer and expect thresholds to change or remain the same based on that response.

4.7.6 Logistics Support. Information associated with the representation of logistics support in a DIS exercise shall be communicated through the use of several PDUs: Service Request PDU, Resupply Offer PDU, Resupply Received PDU, Resupply Cancel PDU, Repair Complete PDU, and Repair Response PDU (see also 5.3.5.3 and 6.5).

4.7.6.1 Procedure for Logistics Support. Logistics support in DIS shall be accomplished through a series of request and response messages between two entities. Two types of service have been defined for DIS: resupply and repair.

4.7.6.2 State information for resupply. The following paragraphs describe the different states and transitions for resupply service. An example of the resupply function is given in 6.5.1.

4.7.6.2.1 The receiving host computer. The receiving host computer may be in one of three states:

Ready State	A receiving host computer is in the Ready State when it is able to receive an offer for supplies from an entity with supplying capabilities[1].
Requesting State	A receiving host computer is in the Requesting State when it has requested supplies and has not received a reply to its request.
Receiving State	A receiving host computer is in the Receiving State when it has been offered supplies and is in the process of receiving them.

The behavior of the receiver during resupply service is shown on Figure 4-3.

[1]The fact that an entity is able to provide resupply or repair service is indicated in the capabilities field in its ES PDUs.

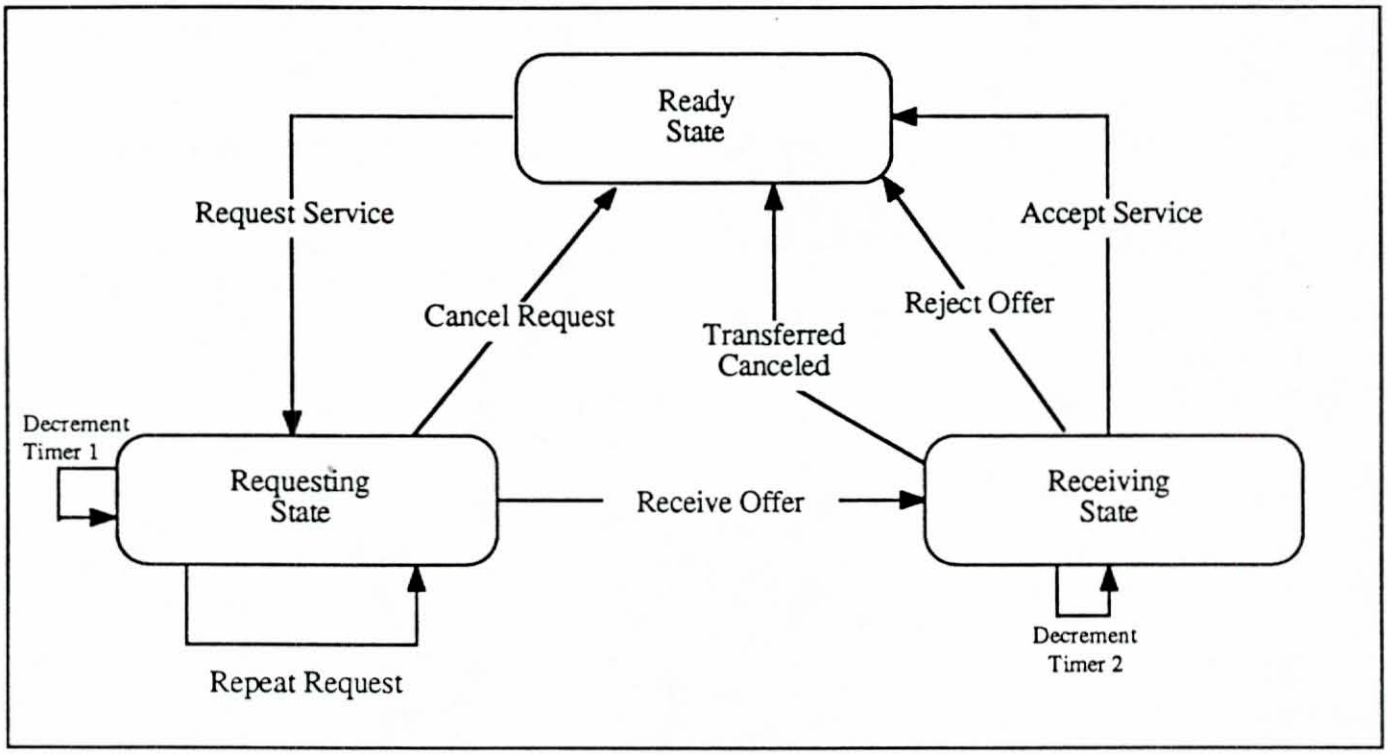


FIGURE 4-3. Receiver behavior during resupply.

Transition	Condition and Actions
Request Service	When conditions for resupply are met, the entity shall issue a Service Request PDU (see 4.7.6.4). The entity shall proceed from the Ready State to the Requesting State and Timer 1 shall be set for five seconds.
Cancel Request	When conditions for resupply are no longer met or when a Resupply Cancel PDU (4.7.6.7) is received from a supplier, Timer 1 shall be canceled and the entity shall proceed from the Requesting State to the Ready State.
Repeat Request	When Timer 1 expires, the Service Request PDU shall be re-issued and the timer shall be reset for five seconds.
Receive Offer	When a Resupply Offer PDU (see 4.7.6.5) is received, Timer 1 shall be canceled and Timer 2 shall be set to the period of time required for receiving some portion of the offered supplies (this time shall not exceed one minute) and the entity shall proceed from the Requesting State to the Receiving State.
Reject Offer	When conditions for resupply are no longer met, a Resupply Cancel PDU shall be issued, Timer 2 shall be canceled, and the entity shall proceed from the Receiving State to the Ready State.
Accept Service	When Timer 2 expires, the count of supplies on board shall be incremented and a Resupply Received PDU (see 4.7.6.6) shall be issued. The entity shall then proceed from the Receiving State to the Ready State.
Transfer Canceled	When a Resupply Cancel PDU (see 4.7.6.7) is received, the ongoing transfer is canceled and the entity proceeds from the Receiving State to the Ready State. No supplies (from the canceled transfer) are transferred.

4.7.6.2.2 The supplying host computer. The supplying host computer may be in one of two states:

Ready State	A supplying host computer is in the Ready State when it is able to receive a request for supplies and is able to offer supplies to a receiver.
Offering State	A supplier in the Offering State has made an offer of supplies and is waiting for the receiver to indicate the quantity of supplies it has taken.

The behavior of the supplier during resupply service is shown on Figure 4-4.

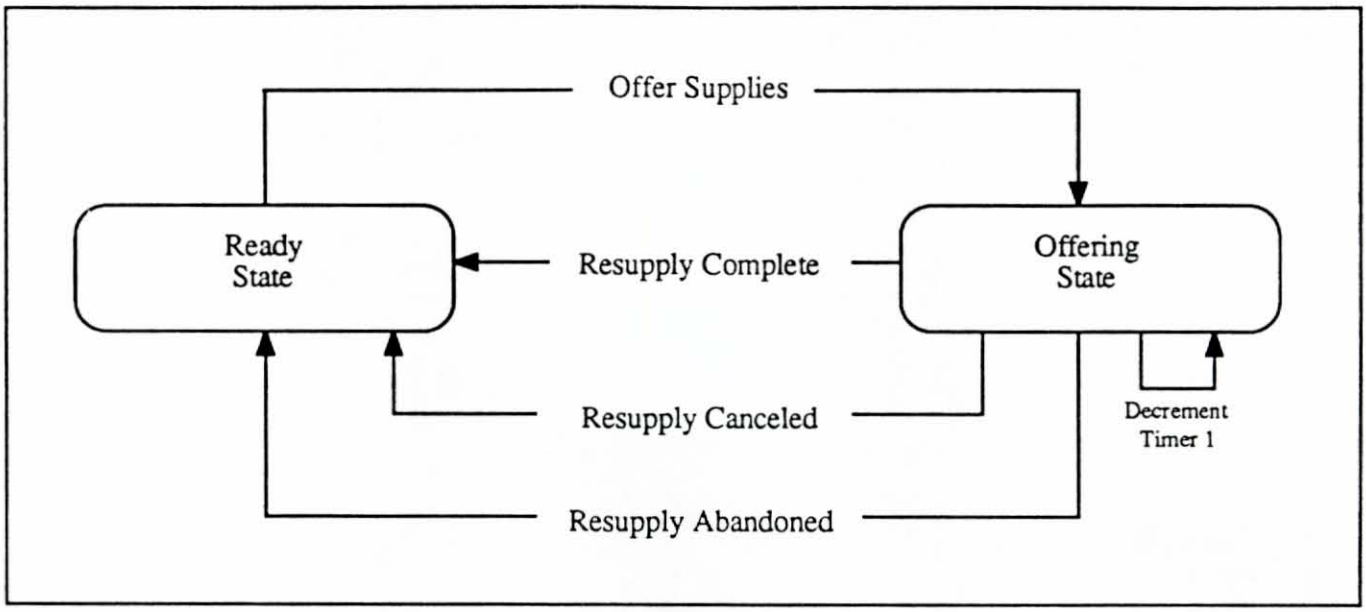


FIGURE 4-4. Supplier behavior during resupply.

Transition	Condition and Actions
Offer Supplies	When a Service Request PDU (see 4.7.6.4) is received, a Resupply Offer PDU shall be issued, Timer 1 shall be set for one minute, and the entity shall proceed from the Ready State to the Offering State.
Resupply Complete	When a Resupply Received PDU (see 4.7.6.6) is received, Timer 1 shall be canceled. The count of munitions on board shall be decremented and the entity shall proceed from the Offering State to the Ready State.
Resupply Canceled	When a Resupply Cancel PDU (see 4.7.6.7) is received, Timer 1 shall be canceled, and the entity shall proceed from the Offering State to the Ready State.
Resupply Abandoned	When Timer 1 expires, the transfer shall be abandoned, and the entity shall proceed from the Offering State to the Ready State.

4.7.6.3 State information for repair. The following paragraphs describe the different states and transitions for repair. An example of repair service is given in 6.5.2.

4.7.6.3.1 The receiving host computer. The receiving host computer may be in one of two states:

Ready State	A receiving host computer is in the Ready State when it is able to receive repairs from an entity with repair capabilities.
Requesting/ Receiving State	A receiving host computer is in the Requesting/Receiving State when it has requested repairs and has not received a response to its request.

The behavior of the receiver during repair service is shown on Figure 4-5.

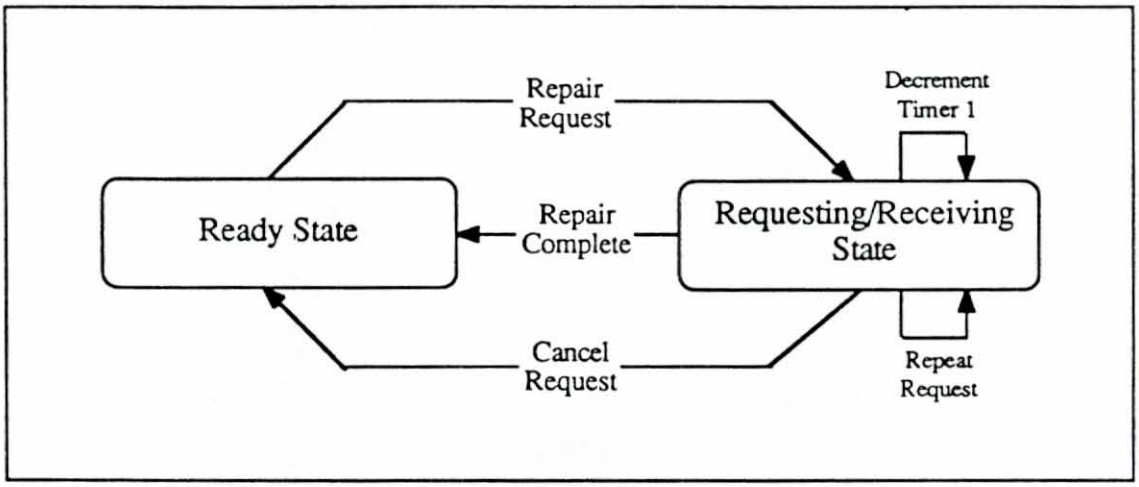


FIGURE 4-5. Receiver behavior during repair service.

Transition	Condition and Actions
Repair Request	When conditions for repair service are met, the entity shall issue a Service Request PDU (see 4.7.6.4). Timer 1 shall be set for five seconds. The entity shall proceed from the Ready State to the Requesting/Receiving State.
Repair Complete	When a Repair Complete PDU (see 4.7.6.8) is received, the entity shall issue a Repair Response PDU (see 4.7.6.9), Timer 1 shall be canceled, and the entity shall proceed from the Requesting/ Receiving State to the Ready State.
Cancel Request	When conditions for repair service are no longer met, the receiver shall cease to issue Service Request PDUs, Timer 1 shall be canceled and the entity shall proceed from the Requesting/Receiving State to the Ready State.
Repeat Request	When Timer 1 expires, the Service Request PDU shall be re-issued and the timer reset for five seconds. The entity shall remain in the Requesting/Receiving State.

4.7.6.3.2 The repairing host computer. The repairing host computer may be in one of two states:

Ready State	A repairing host computer is in the Ready State when it is able to offer repairs to a receiver.
Offering State	A repairing host is in the Offering State when it has received a request for repairs and is responding to the request.

The behavior of the repairing host computer during repair service is shown on Figure 4-6.

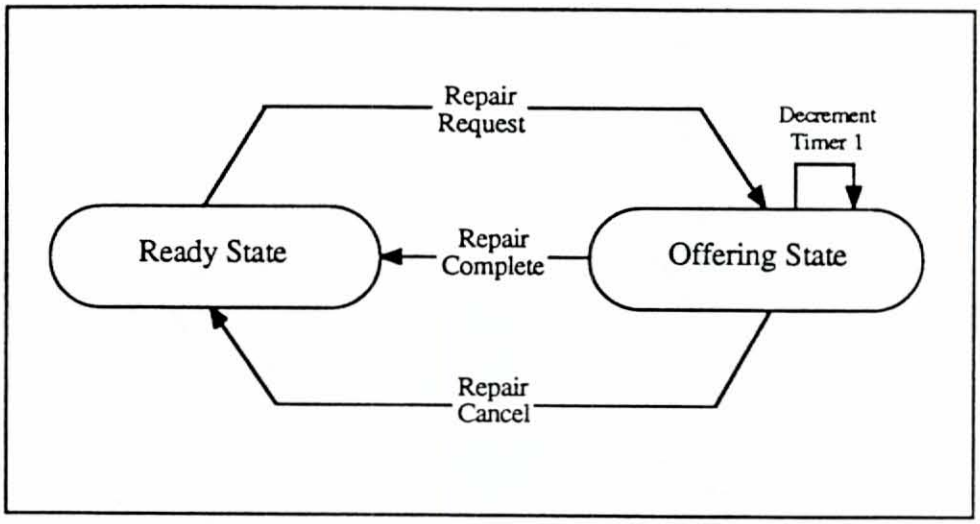


FIGURE 4-6. Repairing host computer behavior during repair service.

Transition	Conditions and Actions
Repair Request	When the repairing host computer has received a Service Request PDU (see 4.7.6.4) and has identified itself as the intended repairer, Timer 1 shall be set to 12 seconds. Every time a Service Request PDU is Received from the receiver, Timer 1 shall be reset to 12 seconds. The entity shall then proceed from the Ready State to the Offering State.
Repair Complete	When the repair is complete, the entity shall issue a Repair Complete PDU (see 4.7.6.8), cancel Timer 1, and shall proceed from the Offering State to the Ready State.
Repair Canceled	When Timer 1 has expired, the entity shall proceed from the Offering State to the Ready State.

4.7.6.4 Service Request PDU. The Service Request PDU shall be used to communicate information associated with one entity requesting a service from another.

4.7.6.4.1 Information contained in the Service Request PDU. The Service Request PDU shall contain the following information:

a. The identification of the entity issuing the PDU.

b. The identification of the entity that is able to provide the service required by the requesting entity.

c. The type of service being requested. Services defined are:

- Resupply
- Repair

d. The number and types of supplies if the service required is resupply.

4.7.6.4.2 Issuance of the Service Request PDU. The Service Request PDU shall be issued by an entity requesting logistics support when appropriate conditions[2] exist.

4.7.6.4.3 Receipt of the Service Request PDU. Upon receipt of a Service Request PDU, the entity that receives the PDU shall respond in one of the following ways:

a. If the service requested is resupply and the entity that receives the PDU is able to provide the needed supplies, the host computer shall issue a Resupply Offer PDU (see 4.7.6.5).

b. If the service requested is repair, the host computer shall simulate the needed repairs in the following manner: The repair process is allowed to proceed as long as the repairing host computer continues to receive Service Request PDUs. If Service Request PDUs cease to be received and are not seen for a period of 12 seconds, the repairing host computer shall assume that conditions for repair no longer exist and therefore shall abort the process.

---

[2]Appropriate conditions include internal conditions (such as certain crew actions in the simulator) as well as external conditions (such as conditions existing in the simulated world - entities not destroyed, or being within a certain distance).

c. If the service requested is resupply and the entity receiving the PDU is unable to provide the supplies requested, then the host computer shall issue a Resupply Cancel PDU (see 4.7.6.7).

4.7.6.5 Resupply Offer PDU. A Resupply Offer PDU shall be used to communicate the offer of supplies by a supplying entity to a receiving entity.

4.7.6.5.1 Information contained in the Resupply Offer PDU. The Resupply Offer PDU shall contain the following information:

a. The identification of the entity requesting resupply.

b. The identification of the supplying entity which issued the PDU.

c. The number of types of supplies that the supplier is able to provide.

d. The supply types available and the amount of each.

4.7.6.5.2 Issuance of the Resupply Offer PDU. The Resupply Offer PDU shall be issued by an identified supplying entity that has received a Service Request PDU (see 4.7.6.4) requesting resupply service.

4.7.6.5.3 Receipt of the Resupply Offer PDU. Upon receipt of a Resupply Offer PDU, the receiving entity proceeds from the Requesting State to the Receiving State. When receipt of the supplies is complete, the receiver shall respond by issuing a Resupply Received PDU.

4.7.6.6 Resupply Received PDU. A Resupply Received PDU shall be used to acknowledge the receipt of supplies by the receiving entity.

4.7.6.6.1 Information contained in the Resupply Received PDU. The Resupply Received PDU shall contain the following information:

a. The identification of the entity requesting resupply.

b. The identification of the supplying entity.

c. The number of types of supplies that the supplier is able to provide.

d. The supply types available and the amount of each taken by the receiver.

4.7.6.6.2 Issuance of the Resupply Received PDU. The Resupply Received PDU shall be issued by an identified receiving entity to indicate the supplies actually transferred from the supplier to the receiver.

4.7.6.6.3 Receipt of the Resupply Received PDU. Upon receipt of a Resupply Received PDU, the supplier decrements the number of supplies on board and proceeds to the Ready State.

4.7.6.7 Resupply Cancel PDU. The Resupply Cancel PDU shall be used to communicate the canceling of a resupply service provided through logistics support.

4.7.6.7.1 Information contained in the Resupply Cancel PDU. The Resupply Cancel PDU shall contain the following information:

a. The identification of the entity receiving supplies.

b. The identification of the entity supplying supplies.

4.7.6.7.2 Issuance of the Resupply Cancel PDU. The Resupply Cancel PDU shall be issued by either the receiver or supplier at any time during resupply to cancel the resupply service.

4.7.6.7.3 Receipt of the Resupply Cancel PDU. The entity receiving the Resupply Cancel PDU shall set its timers to zero and return to the Ready State. No supplies, for the canceled transaction, are transferred.

4.7.6.8 Repair Complete PDU. The Repair Complete PDU shall be used by the repairing host computer to communicate the repair that has been performed for the entity which requested repair service.

4.7.6.8.1 Information contained in the Repair Complete PDU. The Repair Complete PDU shall contain the following information:

a. The identification of the entity requesting repair service.

b. The identification of the entity providing the repair.

c. The repair performed by the repairing entity. Possible repairs are included in Appendix C.

4.7.6.8.2 Issuance of the Repair Complete PDU. The Repair Complete PDU shall be issued by a repairing host computer upon completion of a repair service requested by the receiving entity in a Service Request PDU (see 4.7.6.4).

4.7.6.8.3 Receipt of the Repair Complete PDU. Upon receipt of the Repair Complete PDU the receiving entity shall issue a Repair Response PDU (see 4.7.6.9), and proceed from the Requesting/Receiving State to the Ready State.

4.7.6.9 Repair Response PDU. A Repair Response PDU shall be used by the receiving entity to acknowledge the receipt of a Repair Complete PDU (see 4.7.6.8).

4.7.6.9.1 Information contained in the Repair Response PDU. The Repair Response PDU shall contain the following information:

a. The identification of the entity requesting repair service.

b. The identification of the entity providing the repair.

c. The result of the repair.

4.7.6.9.2 Issuance of the Repair Response PDU. The Repair Response PDU shall be issued by the entity receiving repair service upon receipt of a Repair Complete PDU from the repairing host computer.

4.7.6.9.3 Receipt of the Repair Response PDU. Upon the receipt of the Repair Complete PDU, the repairing entity shall note that the receiving entity has successfully received the repair.

4.7.6.9.4 Cancellation of repair service. If the receiving entity intends to cancel the repair service before the repairs are completed, it ceases to issue Service Request PDUs and returns to the ready state. The supplying entity that does not receive Service Request PDUs for twelve seconds abandons the repair service and returns to the ready state. If the supplying entity intends to cancel the repair service, it issues a Repair Complete PDU but lists the repair result as "no repairs performed." The receiving entity shall respond with the issue of a Repair Response PDU indicating the repair result as "service canceled by the supplier."

4.7.7 Collisions. Information associated with collisions between entities shall be communicated in a DIS exercise through the use of the Collision PDU (see also 5.3.5.4).

4.7.7.1 Collision PDU. The Collision PDU shall be used to communicate information about a collision between two simulated entities or between a simulated entity and another object in the simulated world (such as a cultural feature).

4.7.7.1.1 Information contained in the Collision PDU. The Collision PDU shall contain the following information:

a. The identification of the entity that issued the PDU.

b. The identification of the entity with which the issuing entity collided. If this ID number is unknown, the ID field shall contain zeros.

c. The time at which the data is valid.

d. The identification of the specific event marked by the collision of the entities.

e. Information for damage determination. This information, when available, shall be used by each entity to determine the extent of damage received during the collision. This information includes:

- The velocity vector of the issuing entity.
- The mass of the issuing entity.
- The location of impact in entity coordinates of the entity with which the issuing entity collided.

These fields shall all be used or all be zero.

4.7.7.1.2 Issuance of the Collision PDU. The Collision PDU shall be issued by an entity when a collision is detected between the issuing entity and an object or some other entity taking part in the simulation exercise.

4.7.7.1.3 Receipt of the Collision PDU. Upon receipt of the Collision PDU, the data contained therein shall be used to record the event and to determine the extent of the damage sustained in the collision.

4.7.8 Electronic Interaction. Information associated with electronic interactions shall be communicated in a DIS exercise through the use of the Radar PDU (see also 5.3.5.5).

4.7.8.1 Radar PDU. The Radar PDU shall be used to communicate information about radar use in a simulation exercise.

4.7.8.1.1 Information contained in the Radar PDU. The Radar PDU shall contain the following information:

a. The identification of the issuing entity.

b. The time at which the data in the PDU is valid.

c. The identification of the specific event marked by the use of an entity's radar.

d. The number of radar systems and the following information for each radar system:

- The location of the radar with respect to the entity.
- The type of radar system represented.
- The mode that the particular radar is in.
- The specific data for the radar as required. Currently the use of this field is optional. If the field is not used it shall contain zeros.
- The angles required to describe the volume covered by the radar scan.
- The average power being transmitted.
- The number of simulated entities that are illumined[3] by a particular radar emission. This list may be incomplete. Based on the number illumined, the following fields are also included:

    (1) Entity identifiers of the entities that are illumined by a particular radar.

    (2) Radar data specific to the applicable target ID.

4.7.8.1.2 Issuance of the Radar PDU. The Radar PDU shall be issued by an entity when one or more of the following conditions exist:

a. A radar system has changed modes.

b. A radar system has changed power.

---

[3]Entities illumined shall be entities that have been detected by the radar. This list may not include entities that have detected the radar scan but were not detected by the radar.

c. A radar system has had a change in the angles describing the volume of the scan.

d. A change in the entities illumined (if applicable) has occurred.

4.7.8.1.3 Receipt of the Radar PDU. Upon receipt of the Radar PDU, if the entity is indicated as being illumined or the entity determines that it is in the volume of the radar sweep, the receiving entity shall use the information to simulate any detection equipment that the host computer has.

5. DETAILED REQUIREMENTS

5.1 Representation of data.

5.1.1 Byte ordering. Byte ordering shall adhere to MIL-STD-1777 Internet Protocol Specification, Appendix A. This ordering is represented on Figure 5-1.

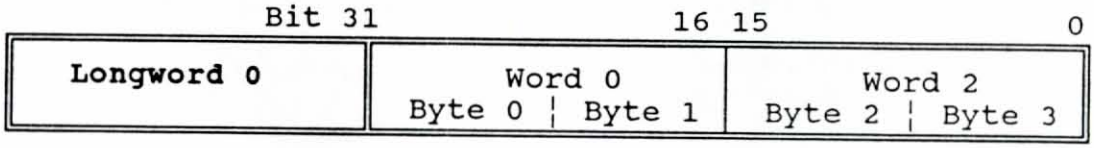


FIGURE 5-1. Byte ordering.

This ordering specifies:

- Most significant byte at lowest address
- Word is addressed by byte address of most significant byte

5.1.2 Enumeration representation. Enumeration types shall adhere to MIL-STD-1777 Internet Protocol Specification, Appendix A. Double and single precision enumerated types shall begin with zero in all cases for the first element of the type declaration.

5.1.3 Number representation. Numbers shall be represented as either floating point or integers. These numbers shall be expressed using the following format:

5.1.3.1 Floating point. Single and Double precision floating point numbers shall adhere to the IEEE 754-1985 Standard.

5.1.3.2 Integers. Integers shall be represented as signed or unsigned. Signed integers shall be represented in two's-complement form where the most significant bit shall designate the sign bit. This bit shall have a value of zero for positive numbers and one for negative numbers. Integers may have a size of 8, 16, or 32-bits.

5.2 Basic data types and records. This section specifies requirements for basic data types and records. Enumeration values and bit-encoded values are given in Appendix F unless otherwise stated.

5.2.1 Angle Representation. Angles shall be specified using one of two formats:

a. 32-bit integers expressed in binary angle measurement (BAM)
b. 16-bit integers expressed in binary angle measurement (BAM)

The 32-bit format shall be used to represent entity orientation. The 16-bit format shall be used to represent articulated parts orientation. Other uses for angle measure shall use which ever format is specified.

5.2.2 Angular Velocity Vector Record. The angular velocity of simulated entities shall be represented by the Angular Velocity Record. This record shall specify the rate at which an entity's orientation is changing. This rate shall be measured in BAMs per millisecond measured about each of the entity's own coordinate axes. The record shall consist of three fields. The first field shall represent velocity about the x-axis, the second about the y-axis, and the third about the z-axis (see 5.2.21.1). The positive direction of velocity is measured as counter-clockwise motion about each axis. The Angular Velocity Record is represented on
Figure 5-2.

Rate about x-axis	32-bit signed integer
Rate about y-axis	32-bit signed integer
Rate about z-axis	32-bit signed integer

FIGURE 5-2. Angular Velocity Vector Record.

5.2.3 Articulated Parts Records.

5.2.3.1 Articulated Parts Record. The specification of articulation parameters for movable parts and attached parts of an entity shall be represented by an Articulated Parts Record.

This record shall specify whether a change has occurred, the Part ID of the articulated part to which it is attached, number of parameters, and the type and value of each parameter (see Appendix G for further explanation and examples).

5.2.3.1.1 Change indicator. The change of any parameter for a particular articulated part shall be indicated by a change indicator field. This field shall be specified by a 16-bit unsigned integer. This field shall contain 1's if a parameter has changed and 0's if no change has occurred.

5.2.3.1.2 Part ID. The articulated part identification of an articulated part to which another articulated part is attached shall be specified by a 16-bit unsigned integer. This field shall contain the value zero if the articulated part is attached directly to the entity.

5.2.3.1.3 Number of parameters. The number of parameters for each articulated part shall be specified by an 8-bit unsigned integer.

5.2.3.1.4 Parts Parameter Record. The specification of parameters for each articulated part shall be represented with a Parts Parameter Record. The Parts Parameter record shall consist of the parameter type and the parameter value of one parameter for an articulated part.

5.2.3.1.4.1 Parameter type. The type of parameter represented shall be specified by a 32-bit unsigned integer.

5.2.3.1.4.2 Parameter value. The parameter value shall be specified by a 32-bit floating point number.

The Parts Parameter record is represented on Figure 5-3.

Parameter Type	32-bit unsigned integer
Parameter Value	32-bit floating point

FIGURE 5-3. Parts Parameter Record.

The Articulated Parts Record is represented on Figure 5-4.

Change Indicator	16-bit unsigned integer
Part ID of part to which attached	16-bit unsigned integer
# of parameters	16-bit unsigned integer
Parameter #1	Parts Parameter Record
Parameter #2	"
.	"
.	"
Parameter #n	"

FIGURE 5-4. Articulated Parts Record.

5.2.4 Boolean. A boolean data type shall be represented as a single bit representing a true-false value. This bit shall represent an enumeration type of one bit, where the value 0 is interpreted as false, and the value 1, as true.

5.2.5 Burst Descriptor Record. The firing of a round or a burst of ammunition shall be represented by a Burst Descriptor Record. This record shall specify the type of munition fired, the type of warhead, the type of fuze, the number of rounds fired, and the rate at which the rounds are fired in rounds per second. The fields of this record are described in the subparagraphs that follow.

5.2.5.1 Munition. The munition shall be specified by an Entity Type Record defined in 5.2.10.

5.2.5.2 Warhead. The warhead shall be specified by a 16-bit enumeration (see Appendix B).

5.2.5.3 Fuze. The fuze shall be specified by a 16-bit enumeration (see Appendix B).

5.2.5.4 Quantity and rate. Quantity and rate each shall be specified by 16-bit unsigned integers. Quantity shall represent the number of rounds fired in the burst, and rate shall

represent the rounds per second for the munition specified. For quantity equal to one, the rate field shall contain zeros. The Burst Descriptor Record is represented on Figure 5-5.

Munition	Entity Type 64-bit unsigned integer
Warhead	16-bit enumeration
Fuze	16-bit enumeration
Quantity	16-bit unsigned integer
Rate	16-bit unsigned integer

FIGURE 5-5. Burst Descriptor Record.

5.2.6 Date/Time. The date and time in the simulated world shall be represented using date/time data type. This data type shall specify the number of seconds elapsed since 00:00:00 GMT, January 1, 1970. This field shall be represented by a 32-bit unsigned integer.

5.2.7 Entity Capabilities Record. The capabilities of an entity shall be specified by an Entity Capabilities Record. This record shall be defined as a 32-bit record of boolean types. The values defined for this record are included on Figure 5-6.

1st bit	2nd bit	3rd bit	4th bit	5th - 32nd
Ammunition Supply Capability	Fuel Supply Capability	Misc. Supply Capability	Repair Capability	Additional Capabilities to be defined

FIGURE 5-6. Entity Capabilities Record.

5.2.8 Entity Identifier Record. The unique identification of each entity in an exercise shall be specified by an Entity Identifier Record. This identification number shall consist of a Simulation Address Record (see 5.2.8.1) and an entity

identification number. No entity shall be assigned an ID containing all zeros or all ones. The fields of this record are described in the subparagraphs that follow:

5.2.8.1 Simulation Address Record. An entity's simulation address shall be specified by a Simulation Address Record. A Simulation Address Record shall consist of the site identification number and the host identification number. These fields are described in 5.2.8.1.1 and 5.2.8.1.2. The Simulation Address Record is represented on Figure 5-7.

5.2.8.1.1 Site Identifier. The site identification number shall be assigned at the time a site is permitted to be a DIS site. This number is assigned by the authority that is managing DIS exercises and remains with that site for each exercise thereafter. This identifier shall be specified by a 16-bit unsigned integer.

5.2.8.1.2 Host Identifier. The host identification number shall be assigned by the authority managing a particular site. This number shall be registered with the authority that is managing DIS exercises and remains with that host for each exercise thereafter. This identifier shall be specified by a 16-bit unsigned integer.

Site Identifier	16-bit unsigned integer
Host Identifier	16-bit unsigned integer

FIGURE 5-7. Simulation Address Record.

5.2.8.2 Entity Identification Number. Each entity participating in an exercise shall have a unique entity number that is assigned by the host activating that entity. This number is unique to the host simulating the entity and is valid for the duration of the exercise for which the entity has been activated. No two entities being simulated by the same host computer shall have the same entity ID. No entity shall have an ID number of zero. Entity IDs assigned to munitions shall be assigned sequentially and shall not be identical to other entity IDs in use. The Munition ID shall be reused if all possible numbers have been exhausted. This number need not be registered or

retained for future exercises. The entity identification number shall be set to one for each exercise and incremented by one for each entity represented by a particular host. This number shall be represented using a 16-bit unsigned integer. The Entity Identifier Record is represented on Figure 5-8.

Simulation Address	Site Address: 16-bit unsigned integer Host Address: 16-bit unsigned integer
Entity Identification Number	16-bit unsigned integer

FIGURE 5-8. Entity Identifier Record.

5.2.9 Entity Marking Record. Entity markings shall be specified by the Entity Marking Record. This record shall specify the character set used in the marking and the string of characters to be interpreted for display. The character set shall be specified by an 8-bit enumeration. The string of characters shall be represented by an 11 element character string. This string shall begin with the most significant byte located at the lowest address. Characters not used shall contain zeros. The Entity Marking Record is represented on Figure 5-9.

Character Set	8-bit enumeration
1st Character	8-bit unsigned integer
2nd Character	8-bit unsigned integer
.	.
.	.
11th Character	8-bit unsigned integer

FIGURE 5-9. Entity Marking Record.

5.2.10 Entity Type Record. The type of entity in a DIS exercise shall be specified by an Entity Type Record. This record shall specify the kind of entity, the country associated with that entity, the domain, the specific identification of the entity, and any extra information necessary for describing the entity. Specific enumeration values for each of these fields are

contained in Appendix H1 and H2 unless otherwise stated. See Appendix H1 for more specific information on Entity Types. Fields not used shall contain the value zero. These fields are described below.

5.2.10.1 Kind. This field shall identify the kind of entity described by the Entity Type Record. This field shall be represented by an 8-bit enumeration.

5.2.10.2 Domain. This field shall specify the domain in which the entity operates (for example, sub-surface, surface, land) except for munition entities. For munitions entities this field shall specify the domain of the target (for example, the munition might be an air-to-surface, so the domain would be anti-surface). This field shall be represented by an 8-bit enumeration.

5.2.10.3 Country. This field shall specify the country to which the entity belongs. This field shall be represented by a 16-bit enumeration (see Appendix F).

5.2.10.4 Category. This field shall specify the main category that describes the entity. This field shall be represented by an 8-bit enumeration.

5.2.10.5 Subcategory. This field shall specify a particular subcategory to which an entity belongs based on the category field. This field shall be represented by an 8-bit enumeration.

5.2.10.6 Specific. This field shall specify specific information about an entity based on the subcategory field. This field shall be represented by an 8-bit enumeration.

5.2.10.7 Extra. This field shall specify extra information required to describe a particular entity. The contents of this field shall depend on the type of entity represented. This field shall be represented by an 8-bit enumeration. The Entity Type Record is represented on Figure 5-10.

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

FIGURE 5-10. Entity Type Record.

5.2.11 Euler Angles Record. Orientation of a simulated entity shall be specified by the Euler Angles Record. This record shall specify three angles as described in 3.17. These angles shall be in terms of the entity's coordinate system (see 5.2.21.1). The three angles shall each be specified by a 32-bit integer representing units of BAM. The Euler Angles Record is represented on Figure 5-11.

PSI	32-bit BAM
THETA	32-bit BAM
PHI	32-bit BAM

FIGURE 5-11. Euler Angles Record.

5.2.12 Event Identifier Record. Event identification shall be specified by the Event Identifier Record. This record shall consist of a Simulation Address Record and a 16-bit unsigned integer specifying the event number. The latter is uniquely assigned by the Host that initiates the sequence of events. The Event Identifier Record shall be set to one for each exercise and incremented by one for each event. In the case

where all possible values are exhausted, the numbers may be reused beginning again at one. The Event Identifier Record is represented on Figure 5-12.

Simulation Address Record	Site ID: 16-bit unsigned integer Host ID: 16-bit unsigned integer
Event ID	16-bit unsigned integer

FIGURE 5-12. Event Identifier Record.

5.2.13 Exercise Identifier. Exercise identification shall be specified by an 8-bit unsigned integer value. This value shall be unique to each exercise occurring simultaneously on the same communications medium.

5.2.14 Organizational Unit Record. The organizational unit to which a simulated entity belongs shall be specified by the Organizational Unit Record. This record shall contain fields specifying the force, country, and service branch to which the entity belongs, as defined in Appendix E. It shall specify a particular unit by using a hierarchy which depends on the information in the force, country, and service branch fields. These fields are described in the subparagraphs that follow. The Organizational Unit Record is represented on Figure 5-13.

5.2.14.1 Force ID. This field shall specify the highest level organizational component. This field shall be specified by an 8-bit enumeration (see Appendix E).

5.2.14.2 Country ID. This field shall identify the country or political affiliation to which the unit belongs. The Country ID shall be specified by a 16-bit enumeration (see Appendix F).

5.2.14.3 Service ID. This field shall specify the military service to which the unit belongs. The Service ID shall be specified by an 8-bit enumeration (see Appendix E).

5.2.14.4 Hierarchy. This field shall describe the unit number and the type of unit associated with each organization level applicable to the entity. Unit types shall be defined according to the service to which the unit belongs. Unit numbers shall be defined for up to eight unit types. Each level shall

consist of an 8-bit unsigned integer for the unit number at that level. Specific unit types for each service are defined in Appendix E.

Force	Force ID: 8-bit enumeration
Country	Country ID: 16-bit enumeration
Service	Service ID: 8-bit enumeration
Hierarchy	Eight levels are defined. The levels depend on the Service ID. Each level consists of: Unit Number: 8-bit unsigned integer

FIGURE 5-13. Organizational Unit Record.

5.2.15 Radar system data type. The radar system of an emitter shall be specified by a 32-bit unsigned integer. The following fields are defined for this 32-bit number. Information associated with specific values of these fields is included in Appendix F.

a. Radar System Category - bits 23 - 31.
This field specifies the type of radar system represented by this record.

b. Radar System Subcategory - bits 16 - 23.
This field allows further classification of radar systems.

c. Radar System ID - bits 0 - 15.
This field identifies the specific radar represented by this record.

The radar system data type is represented on Figure 5-14.

Radar System Category	Bits 23 - 31
Radar System Subcategory	Bits 16 - 23
Radar System ID	Bits 0 - 15

FIGURE 5-14. Radar system data type.

5.2.16 Repair type. Repair types shall be specified by a 16-bit enumeration (see Appendix C).

5.2.17 Service type. Service types shall be specified by an 8-bit enumeration (see Appendix F).

5.2.18 Supply Quantity Record. Supply quantity shall be represented by the Supply Quantity Record. This record shall contain fields specifying the type of supply and the quantity of that supply. These fields are described in 5.2.18.1 and 5.2.18.2.

5.2.18.1 Supply type. The supply type field shall be specified by an Entity Type Record (see 5.2.10 and Appendix B).

5.2.18.2 Quantity. The quantity field shall be specified by a 32-bit floating point number representing the number of units of a supply type. The unit measure depends on the supply type and shall use the standard international units of measure used for such supplies. The Supply Quantity Record is represented on Figure 5-15.

Supply Type	Entity Type: 64-bit record
Quantity	32-bit floating point

FIGURE 5-15. Supply Quantity Record.

5.2.19 Terrain Database Identifier Record. Identification of the terrain database shall be accomplished using the Terrain Database Identifier record. This identifier shall be specified by an 11 element character string representing the terrain database name and an 8-bit unsigned integer representing the

version number. The terrain database name shall be specified using the ASCII character set. This string shall begin with the most significant byte located at the lowest address. Bytes not used for characters shall contain zeros. The Terrain Database Identifier record is represented on Figure 5-16.

Terrain Database Name	11 element character string
Version Number	8-bit unsigned integer

FIGURE 5-16. Terrain Database Identifier Record.

5.2.20 Time stamping. Time stamping shall be used to indicate the time at which the data contained in the PDU is valid. This timestamp shall be specified using a 32-bit unsigned integer representing units of time passed since the beginning of the current hour. The least significant bit shall indicate whether the timestamp is absolute or relative.

5.2.20.1 Absolute timestamp. An absolute timestamp shall be used when simulator clocks are synchronized to Universal Coordinated Time (UTC). The use of the absolute timestamp shall be signified by the least significant bit set to one.

5.2.20.2 Relative timestamp. A relative timestamp shall be used when simulator clocks are not synchronized. Each simulator shall keep time beginning with an arbitrary starting point. The time indicated by the timestamp shall be relative to the simulator issuing the PDU. The use of the relative timestamp shall be signified by the least significant bit set to zero.

5.2.20.3 Scale. The scale of the timestamp shall be determined by setting one hour equal to 2^{31}, thereby resulting in each unit representing $3600sec/2^{31} = 1.676$ microseconds.

5.2.21 Vector Record. Vector values for entity coordinates, linear velocity, and linear acceleration shall be represented using the Vector Record. This record shall consist of three fields, each 32-bit floating point numbers. The unit of measure represented by these fields shall depend on the information represented. The values utilizing the Vector Record are described in detail below. The Vector Record is represented on Figure 5-17.

5.2.21.1 Entity coordinates. Location with respect to a particular entity shall be specified using a set of three coordinates for three orthogonal axes whose origin shall be the center of the bounding volume of the entity. The x-axis extends

in the positive direction out the front of the entity. The y-axis extends in the positive direction out the right side of the entity. The z-axis extends in the positive direction downward. Each vector component shall represent meters from the origin.

5.2.21.2 Linear acceleration vector. Linear acceleration shall be represented as a vector with x, y, and z components where x, y, and z shall represent directions of the entity's coordinate axis. Each vector component shall represent acceleration in meters per second squared.

5.2.21.3 Linear velocity vector. Linear velocity shall be represented as a vector with x, y, and z components where x, y, and z shall represent directions of the world coordinate axes. Each vector component shall represent velocity in meters per second.

First Vector Component	32-bit floating point
Second Vector Component	32-bit floating point
Third Vector Component	32-bit floating point

FIGURE 5-17. Vector Record.

5.2.22 World Coordinates Record. Location shall be specified by a set of three coordinates: x, y, and z. The origin of this coordinate system shall be the centroid of the earth, with the x-axis passing through the Prime Meridian at the equator, the y-axis passing through 90 degrees East longitude at the Equator, and the z-axis passing through the North Pole. These coordinates shall represent meters from the centroid of the earth. A 64-bit double precision floating point number shall represent the location for each coordinate. The World Coordinates Record is represented on Figure 5-18.

X - Coordinate	64-bit floating point
Y - Coordinate	64-bit floating point
Z - Coordinate	64-bit floating point

FIGURE 5-18. World Coordinates Record.

5.3 Protocol Data Units for Distributed Interactive Simulation

5.3.1 Introduction. Section 5.3 lists and describes the Protocol Data Units required by this standard. Note that on the Figures in section 5 all PDU fields begin on 32-bit boundaries to simplify the processing in the host computers. Padding bits used to achieve these 32-bit boundaries are shown only on the figures.

5.3.2 List of DIS Protocol Data Units.

I. Entity Information

- A. Entity State
  1. Entity State PDU

II. Entity Interaction

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

5.3.3 Protocol Data Unit (PDU) header. A protocol data unit header shall be the first part of each protocol data unit. This PDU header is represented on Figure 5-19. The fields of the PDU Header are described in 5.3.3.1, 5.3.3.2, and 5.3.3.3 (see also 4.7.1).

a. Protocol version - This field shall specify the version of protocol used in this PDU. This field shall be specified by an 8-bit unsigned integer.

b. Exercise identification - This field shall specify the exercise to which the PDU pertains. This field shall be represented by an exercise identifier (see 5.2.13).

c. Protocol Data Unit type - This field indicates the type of PDU that follows. This field shall be represented by an 8-bit enumeration.

FIELD SIZE (bits)	PROTOCOL DATA UNIT HEADER FIELDS	
8	PROTOCOL VERSION	8 - bit unsigned integer
8	EXERCISE INDENTIFIER	8 - bit unsigned integer
8	PDU TYPE	8 - bit enumeration
8	PADDING	8 bits unused

Figure 5-19. PDU Header.

5.3.4 Entity Information. (see also 4.7.2).

5.3.4.1 Entity State PDU. Information about a particular entity shall be communicated by issuing an Entity State PDU. The Entity State PDU shall contain the following fields:

a. Entity Identification - This field shall identify the entity issuing the PDU. This field shall be represented by an Entity Identifier Record (see 5.2.8).

b. Entity Type - This field shall identify the entity type responsible for the issue of this PDU. This field shall be represented by an Entity Type Record (see 5.2.10).

c. Timestamp - This field shall specify the time at which the data in the PDU is valid. This field shall be represented by a timestamp (see 5.2.20).

d. Entity Location - This field shall specify an entity's physical location in the simulated world. This field shall be represented by a World Coordinates Record (see 5.2.22).

e. Entity Velocity - This field shall specify an entity's linear velocity. This field shall be represented by a Vector Record (see 5.2.21.3).

f. Entity Acceleration - This field shall specify an entity's acceleration. This field shall be represented by a Vector Record (see 5.2.21.2).

g. Entity Orientation - This field shall specify an entity's orientation. This field shall be represented by an Euler Angles Record (see 5.2.11).

h. Entity Angular Velocity - This field shall specify an entity's angular velocity. This field shall be represented by an Angular Velocity Vector Record (see 5.2.2).

i. Dead Reckoning Parameters - This field will be used to provide parameters for dead reckoning the position and orientation of the entity. These parameters are currently undefined. This standard reserves 64 bits for these parameters.

j. Entity Appearance - This field shall specify the dynamic changes to the entity's attributes. This field shall be represented by a 32-bit unsigned integer. The

description of this 32-bit field is defined in Appendix D.

k. Marking - This field shall identify any unique markings on an entity (for example, a bumper number or country symbol). This field shall be represented by an Entity Marking Record (see 5.2.9).

l. Capabilities - This field shall specify the entity's capabilities. This field shall be represented by an Entity Capabilities Record (see 5.2.7).

m. Number of Articulated Parts - This field shall specify the number of articulated parts whose presence and position are communicated through the Articulated Parts Record. This field shall be represented by an 8-bit unsigned integer.

n. Articulated Parts - This field shall specify the orientation of each articulated part. This field shall be represented by an Articulated Parts Record (see 5.2.3).

The Entity State PDU is represented on Figure 5-20.

FIELD SIZE (bits)	ENTITY STATE PDU FIELDS	
48	ENTITY ID	SITE - 16 - bit unsigned integer
		HOST - 16 - bit unsigned integer
		ENTITY - 16 - bit unsigned integer
16	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
32	TIME STAMP	32 - bit unsigned integer
192	ENTITY LOCATION	X - Component - 64 - bit floating point
		Y - Component - 64 - bit floating point
		Z - Component - 64 - bit floating point
96	ENTITY LINEAR VELOCITY	X - Component - 32 - bit floating point
		Y - Component - 32 - bit floating point
		Z - Component- 32 - bit floating point
96	ENTITY LINEAR ACCELERATION	X - Component - 32 - bit floating point
		Y - Component - 32 - bit floating point
		Z - Component - 32 - bit floating point

FIGURE 5-20. Entity State PDU.

FIELD SIZE (bits)	ENTITY STATE PDU FIELDS (CONT'D)	
96	ENTITY ORIENTATION	Psi 32 - bit BAM
		Theta 32 - bit BAM
		Phi 32 - bit BAM
96	ENTITY ANGULAR VELOCITY	X - Component - 32 - bit signed integer
		Y - Component - 32 - bit signed integer
		Z - Component - 32 - bit signed integer
64	DEAD RECKONING PARAMETERS	64 bits - undefined
32	ENTITY APPEARANCE	32 - bit unsigned integer
96	ENTITY MARKING	CHARACTER SET - 8 - bit enumeration
		11 element character string
32	CAPABILITIES	32 bits of Boolean fields
24	PADDING	24 bits unused
8	# of articulated parts	8 - bit unsigned integer
n x (96m + 32)	ARTICULATED PARTS	See Figure G-1, Appendix G

n = # of articulated parts
m = # of articulation parameters for each part
(i = 1 to n)

FIGURE 5-20. Entity State PDU.

5.3.5 Entity Interaction.

5.3.5.1 Weapons fire. (see also 4.7.3)

5.3.5.1.1 Fire PDU. The firing of a weapon shall be communicated by issuing a Fire PDU. The Fire PDU shall contain the following fields:

a. Firing Entity Identification - This field shall identify the firing entity. This field shall be represented by an Entity Identifier Record (see 5.2.8).

b. Target Identification - This field shall identify the intended target. If the intended target is unknown, this field shall contain zeros. This field shall be represented by an Entity Identifier Record (see 5.2.8).

c. Munition Identification - This field shall identify the guided munition (launched by this firing action) as a unique entity. This number is assigned by the host computer modeling the entity that is issuing the Fire PDU. The number assigned to the munition shall be unique to the modeling host for that particular exercise. If all possible values have been assigned, the assignment shall begin again at one and proceed sequentially. A munition ID shall not be reused if it is currently active. A munition ID shall have a value of zero if the munition is not guided. This field shall be represented by an Entity Identifier Record (see 5.2.8).

d. Event Identification- This field shall contain a number generated by the firing entity to associate related events. This field shall be represented by an Event Identifier Record (see 5.2.12).

e. Timestamp - This field shall specify the time at which the data in the PDU is valid. This field shall be represented by a timestamp (see 5.2.20).

f. Location - This field shall specify the location from which the munition was launched. This field shall be represented by a World Coordinates Record (see 5.2.22).

g. Burst Descriptor - This field shall describe the type of munition fired, the warhead, the fuze, the quantity, and rate. This field shall be represented by a Burst Descriptor Record (see 5.2.5).

h. Velocity Vector - This field shall specify speed and direction of the fired munition. This field shall be represented by a Vector Record (see 5.2.21.3).

i. Range - This field shall specify the range (in meters) that an entity's fire control system has assumed in computing the fire control solution. This field shall be represented as a 32-bit floating point number. For systems where range is unknown or unavailable, this field shall contain a value of zero.

The Fire PDU is represented on Figure 5-21.

FIELD SIZE (bits)	FIRE PDU FIELDS	
48	FIRING ENTITY ID	SITE - 16 - bit unsigned integer
		HOST - 16 - bit unsigned integer
		ENTITY - 16 - bit unsigned integer
48	TARGET ENTITY ID	SITE - 16 - bit unsigned integer
		HOST - 16 - bit unsigned integer
		ENTITY - 16 - bit unsigned integer
48	MUNITION ID	SITE - 16 - bit unsigned integer
		HOST - 16 - bit unsigned integer
		ENTITY - 16 - bit unsigned integer
48	EVENT-ID	SITE - 16 - bit unsigned integer
		HOST - 16 - bit unsigned integer
		EVENT - 16 - bit unsigned integer
32	TIME STAMP	32 - bit unsigned integer
192	LOCATION IN WORLD	X-coordinate- 64 - bit floating pt
		Y-coordinate- 64 - bit floating pt
		Z-coordinate- 64 - bit floating pt
128	BURST DESCRIPTOR	MUNITION - See Entity Type Record
		WARHEAD - 16 - bit enumeration
		FUZE - 16 - bit enumeration
		QUANTITY - 16 - bit unsigned integer
		RATE -16 - bit unsigned integer
96	VELOCITY	X-component 32 - bit floating pt
		Y-component 32 - bit floating pt
		Z-component 32 - bit floating pt
32	RANGE	32 - bit floating pt

Figure 5-21. Fire PDU.

5.3.5.1.2 Detonation PDU. The detonation or impact of munitions shall be communicated by issuing a Detonation PDU. The Detonation PDU shall contain the following fields:

a. Firing Entity Identification - This field shall identify the firing entity. This field shall be represented by an Entity Identifier Record (see 5.2.8).

b. Target Entity Identification - This field shall identify the target entity. This field shall be represented by an Entity Identifier Record (see 5.2.8). If the target ID is unknown, this field shall contain the value zero.

c. Munition Identification - This field shall specify the munition's entity ID for the munition as specified in the Fire PDU that communicated the launch of the munition. Munitions that are not guided shall have an identification of zero. This field shall be represented by an Entity Identifier Record (see 5.2.8).

d. Event Identification. This field shall contain the same number as the event identifier of the Fire PDU that communicated the launch of the munition. This field shall be represented by an Event Identifier Record (see 5.2.12).

e. Timestamp - This field shall specify the time at which the data in the PDU is valid. This field shall be represented by a timestamp (see 5.2.20).

f. Location - This field shall specify the location of the detonation. If the target is unknown, this information shall be used for damage assessment. This field shall be represented by a World Coordinates Record (see 5.2.22).

g. Burst Descriptor - This field shall describe the type of munition fired, the warhead, the fuze, the quantity and the rate. This field shall be represented by a Burst Descriptor Record (see 5.2.5).

h. Velocity - This field shall specify the velocity of the munition immediately before detonation. This field shall be represented by a Vector Record (see 5.2.21.3).

i. Location with Respect to Target Entity - This field shall specify the location of the detonation in terms of the target entity's coordinates. This field shall

be represented by a Vector Record (see 5.2.21.1). If the ID of the target is unknown, this field shall contain zeros.

j. Detonation Result - This field shall specify the result of the detonation. This field shall be represented by an 8-bit enumeration (see Appendix F).

k. Energy - This field shall specify the energy produced by the detonation. This field shall be represented by a 32-bit floating point number specifying energy in joules.

l. Directionality - This field shall specify the directionality of the detonation. This field shall be represented by a 32-bit floating point number specifying the directionality in units of steradians. For a spherically symmetrical explosion, directionality shall be 4π .

m. Momentum - This field shall specify the impact momentum of the munition. This field shall be represented by a 32-bit floating point number specifying momentum in newton-seconds.

The Detonation PDU is represented on Figure 5-22.

FIELD SIZE (bits)	DETONATION PDU FIELDS	
48	FIRING ENTITY ID	SITE - 16 - bit unsigned integer
		HOST - 16 - bit unsigned integer
		ENTITY - 16 - bit unsigned integer
48	TARGET ENTITY ID	SITE - 16 - bit unsigned integer
		HOST - 16 - bit unsigned integer
		ENTITY - 16 - bit unsigned integer
48	MUNITION ID	SITE - 16 - bit unsigned integer
		HOST - 16 - bit unsigned integer
		ENTITY - 16 - bit unsigned integer
48	EVENT ID	SITE - 16 - bit unsigned integer
		HOST - 16 - bit unsigned integer
		EVENT - 16 - bit unsigned integer
32	TIME STAMP	32 - bit unsigned integer
192	LOCATION IN WORLD	X-coordinate - 64 - bit floating pt
		Y-coordinate - 64 - bit floating pt
		Z-coordinate - 64 - bit floating pt
128	BURST DESCRIPTOR	MUNITION - See Entity Type Record
		WARHEAD - 16 - bit enumeration
		FUZE - 16 - bit enumeration
		QUANTITY - 16 - bit unsigned integer
		RATE - 16 - bit unsigned integer

Figure 5-22. Detonation PDU.

FIELD SIZE (bits)	DETONATION PDU FIELDS (CONT'D)	
96	VELOCITY	X - component - 32 - bit floating pt.
		Y - component - 32 - bit floating pt.
		Z - component - 32 - bit floating pt.
96	LOCATION IN ENTITY COORDINATES	X - coordinate - 32 - bit floating pt.
		Y - coordinate - 32 - bit floating pt.
		Z - coordinate - 32 - bit floating pt.
8	DETONATION RESULT	8 - bit enumeration
24	PADDING	24 bits unused
32	ENERGY	32 - bit floating pt
32	DIRECTION-ALITY	32 - bit floating pt
32	MOMENTUM	32 - bit floating pt

Figure 5-22. Detonation PDU.

5.3.5.2 Update Threshold Control. (see also 4.7.5).

5.3.5.2.1 Update Threshold Request PDU. A request for a change in the threshold controlling the rate at which a simulator issues ES PDUs shall be communicated by an Update Threshold Request PDU. The Update Threshold Request PDU shall contain the following fields:

a. Issuing Entity Identification - This field shall identify the entity that is requesting the update threshold change. This field shall be represented by an Entity Identifier Record (see 5.2.8).

b. Responding Entity Identification - This field shall identify the entity requested to change its update threshold. This field shall be represented by an Entity Identifier Record (see 5.2.8).

c. Threshold - This field shall specify the size of the update threshold. This field shall contain two threshold types: Linear and Rotational. These types are described below:

    (1) Linear Threshold - This field shall specify the linear threshold values. Three values shall represent linear distance in meters in the direction of each of the entity's three coordinate axes. This field shall be represented by a Vector Record (see 5.2.21.1).

    (2) Rotational Threshold - This field shall specify the rotational threshold values as three angles in units of BAM. This field shall be represented by an Euler angles Record (see 5.2.11).

d. Duration of Change - This field shall specify the duration of the new update rate. The duration of the change shall not exceed three minutes and shall not equal zero. This field shall be represented as a 32-bit unsigned integer specifying time in seconds.

The Update Threshold Request PDU is represented on Figure 5-23.

FIELD SIZE (bits)	UPDATE THRESHOLD REQUEST PDU FIELDS	
48	ISSUING ENTITY ID	SITE - 16 - bit unsigned integer
		HOST - 16 - bit unsigned integer
		ENTITY - 16 - bit unsigned integer
48	RESPONDING ENTITY ID	SITE - 16 - bit unsigned integer
		HOST - 16 - bit unsigned integer
		ENTITY - 16 - bit unsigned integer
96	LINEAR THRESHOLD	x - 32 - bit floating pt
		y - 32 - bit floating pt
		z - 32 - bit floating pt
96	ROTATIONAL THRESHOLD	Psi - 32 - bit BAM
		Theta - 32 - bit BAM
		Phi - 32 - bit BAM
32	DURATION OF CHANGE	32 - bit unsigned integer

Figure 5-23. Update Threshold Request PDU.

5.3.5.2.2 Update Threshold Response PDU. Response to an Update Threshold Request PDU shall be communicated by issuing an Update Threshold Response PDU. The Update Threshold Response PDU shall contain the following fields:

a. Responding Entity Identification - This field shall identify the entity that is responding to the update threshold change request. This field shall be represented by an Entity Identifier Record (see 5.2.8).

b. Requesting Entity Identification - This field shall identify the entity that is requesting the update threshold change request. This field shall be represented by an Entity Identifier Record (see 5.2.8).

c. Update Change Result - This field shall specify the result of the update request. This field shall be represented as an 8-bit enumeration (see Appendix F).

d. Remaining Time - This field shall specify the time in seconds remaining if the entity already has tighter thresholds implemented. If the entity does not already have tighter thresholds implemented and the thresholds are accepted, then this field shall contain the number of seconds requested in the accepted Update Threshold Request PDU. This field shall be represented as an 8-bit unsigned integer.

The Update Threshold Response PDU is represented on Figure 5-24.

FIELD SIZE (bits)	UPDATE THRESHOLD RESPONSE PDU FIELDS	
48	RESPONDING ENTITY ID	SITE - 16 - bit unsigned integer
		HOST - 16 - bit unsigned integer
		ENTITY - 16 - bit unsigned integer
48	REQUESTING ENTITY ID	SITE - 16 - bit unsigned integer
		HOST - 16 - bit unsigned integer
		ENTITY - 16 - bit unsigned integer
8	RESULT	8 - bit enumeration
8	REMAINING TIME	8 - bit unsigned integer
16	PADDING	16 bits unused

FIGURE 5-24. Update Threshold Response PDU.

5.3.5.3 Logistic Support. (see also 4.7.6 and 6.5).

5.3.5.3.1 Service Request PDU. A request for logistics support shall be communicated by issuing a Service Request PDU. The Service Request PDU shall consist of the following fields:

a. Requesting Entity Identification - This field shall identify the entity that is requesting the service. This field shall be represented by an Entity Identifier Record (see 5.2.8).

b. Servicing Entity Identification - This field shall identify the entity that is able to provide the service requested. This field shall be represented by an Entity Identifier Record (see 5.2.8).

c. Service Type Requested - This field shall describe the type of service being requested. This field shall be represented by a Service Type (see 5.2.17).

d. Number of Supplies - For a service of resupply, this field shall specify the number of different supplies being requested. If the service requested is not resupply, this field shall contain the value zero. This field shall be represented by an 8-bit unsigned integer.

e. Supplies - For a service of resupply, this field shall specify the type of supply and the amount of that supply for the number of supplies specified above. If the service requested is not resupply, this field shall contain the value zero. This field shall be represented by a Supply Quantity Record (see 5.2.18).

The Service Request PDU is represented on Figure 5-25.

FIELD SIZE (bits)	SERVICE REQUEST PDU FIELDS	
48	REQUESTING ENTITY ID	SITE - 16 - bit unsigned integer
		HOST - 16 - bit unsigned integer
		ENTITY - 16 - bit unsigned integer
48	SERVICING ENTITY ID	SITE - 16 - bit unsigned integer
		HOST - 16 - bit unsigned integer
		ENTITY - 16 - bit unsigned integer
8	SERVICE TYPE	8 - bit enumeration
8	Number of (n) Supply types	8 - bit unsigned integer
16	PADDING	16 bits unused
n x 96	SUPPLY QUANTITY	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
		Quantity - 32 - bit floating pt

FIGURE 5-25. Service Request PDU.

5.3.5.3.2 Resupply Offer PDU. The offering of supplies shall be communicated by issuing a Resupply Offer PDU. The Resupply Offer PDU shall contain the following fields:

a. Receiving Entity Identification - This field shall identify the receiving entity. This field shall be represented by an Entity Identifier Record (see 5.2.8).

b. Supplying Entity Identification - This field shall identify the supplying entity. This field shall be represented by an Entity Identifier Record (see 5.2.8).

c. Number of Supplies - This field shall specify the number of different supply types being offered. This field shall be represented by an 8-bit unsigned integer.

d. Supplies - This field shall specify the type of supply and the amount of that supply for each of the supply types specified above. This field shall be represented by a Supply Quantity Record (see 5.2.18).

The Resupply Offer PDU is represented on Figure 5-27.

FIELD SIZE (bits)	RESUPPLY OFFER PDU FIELDS	
48	REQUESTING ENTITY ID	SITE - 16 - bit unsigned integer
		HOST - 16 - bit unsigned integer
		ENTITY - 16 - bit unsigned integer
48	SUPPLYING ENTITY ID	SITE - 16 - bit unsigned integer
		HOST - 16 - bit unsigned integer
		ENTITY - 16 - bit unsigned integer
8	Number of (n) Supply types	8 - bit unsigned integer
24	PADDING	24 bits unused
n x 96	SUPPLY QUANTITY	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
		Quantity - 32 - bit floating pt

FIGURE 5-26. Resupply Offer PDU.

5.3.5.3.3 Resupply Received PDU. The receipt of supplies shall be communicated by issuing a Resupply Received PDU. The Resupply Received PDU shall contain the following fields:

a. Receiving Entity Identification - This field shall identify the receiving entity. This field shall be represented by an Entity Identifier Record (see 5.2.8).

b. Supplying Entity Identification - This field shall identify the supplying entity. This field shall be represented by an Entity Identifier Record (see 5.2.8).

c. Number of Supplies - This field shall specify the number of different supplies taken by the receiver. This field shall be represented by an 8-bit unsigned integer.

d. Supplies - This field shall specify the type of supply and the amount of that supply for each of the supply types specified above. This field shall be represented by a Supply Quantity Record (see 5.2.18).

The Resupply Received PDU is represented on Figure 5-27.

FIELD SIZE (bits)	RESUPPLY RECEIVED PDU FIELDS	
48	RECEIVING ENTITY ID	SITE - 16 - bit unsigned integer
		HOST - 16 - bit unsigned integer
		ENTITY - 16 - bit unsigned integer
48	SUPPLYING ENTITY ID	SITE - 16 - bit unsigned integer
		HOST - 16 - bit unsigned integer
		ENTITY - 16 - bit unsigned integer
8	Number of (n) Supply types	8 - bit unsigned integer
24	PADDING	24 bits unused
n x 96	SUPPLY QUANTITY	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
		Quantity - 32 - bit floating pt

FIGURE 5-27. Resupply Received PDU.

5.3.5.3.4 Resupply Cancel PDU. The canceling of a service function by either the receiving or the supplying entity shall be communicated by issuing a Resupply Cancel PDU. The Resupply Cancel PDU shall contain the following fields:

a. Receiving Entity Identification - This field shall identify the entity that has requested the resupply service. This field shall be represented by an Entity Identifier Record (see 5.2.8).

b. Supplying Entity Identification - This field shall identify the supplying entity. This field shall be represented by an Entity Identifier Record (see 5.2.8).

The Resupply Cancel PDU is represented on Figure 5-28.

FIELD SIZE (bits)	RESUPPLY CANCEL PDU FIELDS	
48	RECEIVING ENTITY ID	SITE - 16 - bit unsigned integer
		HOST - 16 - bit unsigned integer
		ENTITY - 16 - bit unsigned integer
48	SUPPLYING ENTITY ID	SITE - 16 - bit unsigned integer
		HOST - 16 - bit unsigned integer
		ENTITY - 16 - bit unsigned integer

Figure 5-28. Resupply Cancel PDU.

5.3.5.3.5 Repair Complete PDU. When a Service Request PDU has been received and the repairing entity has completed a requested repair, the repairing host computer shall notify the receiver of the repair by issuing a Repair Complete PDU. The Repair Complete PDU shall contain the following fields:

a. Requesting Entity Identification - This field shall identify the entity that is requesting repairs. This field shall be represented by an Entity Identifier Record (see 5.2.8).

b. Repairing Entity Identification - This field shall identify the repairing entity. This field shall be represented by an Entity Identifier Record (see 5.2.8).

c. Repair - This field shall describe the repair performed. If the repairing entity is unable to provide the needed repair, this field shall have a value of zero. If the repairing entity makes all of the needed repairs, this field shall have a value of one. This field shall be represented by a repair type (see 5.2.16 and Appendix C).

The Repair Complete PDU is represented on Figure 5-29.

FIELD SIZE (bits)	REPAIR COMPLETE PDU FIELDS	
48	REQUESTING ENTITY ID	SITE - 16 - bit unsigned integer
		HOST - 16 - bit unsigned integer
		ENTITY - 16 - bit unsigned integer
48	REPAIRING ENTITY ID	SITE - 16 - bit unsigned integer
		HOST - 16 - bit unsigned integer
		ENTITY - 16 - bit unsigned integer
16	REPAIR	REPAIR TYPE-16 - bit enumeration
16	PADDING	16 bits unused

FIGURE 5-29. Repair Complete PDU.

5.3.5.3.6 Repair Response PDU. When a receiving entity receives a Repair Complete PDU from its repairing entity, the receiver shall acknowledge the receipt of the repair by issuing a Repair Response PDU. The Repair Response PDU shall contain the following fields:

a. Requesting Entity Identification. This field shall identify the entity that requested repairs. This field shall be represented by an Entity Identifier Record (see 5.2.8).

b. Repairing Entity Identification - This field shall identify the repairing entity. This field shall be represented by an Entity Identifier Record (see 5.2.8).

c. Repair Result - This field shall specify the result of the repair specified in the Repair Complete PDU. This field shall be represented by an 8-bit enumeration (see Appendix F).

The Repair Response PDU is represented on Figure 5-30.

FIELD SIZE (bits)	REPAIR RESPONSE PDU	
48	REQUESTING ENTITY ID	SITE - 16 - bit unsigned integer
		HOST - 16 - bit unsigned integer
		ENTITY - 16 - bit unsigned integer
48	REPAIRING ENTITY ID	SITE - 16 - bit unsigned integer
		HOST - 16 - bit unsigned integer
		ENTITY - 16 - bit unsigned integer
8	REPAIR RESULT	8 - bit enumeration
24	PADDING	24 bits unused

Figure 5-30. Repair Response PDU.

5.3.5.4 Collisions. (see also 4.7.7).

5.3.5.4.1 Collision PDU. Collisions between entities shall be communicated by issuing a Collision PDU. The Collision PDU shall contain the following fields:

a. Issuing Entity Identification - This field shall identify the entity that is issuing the PDU. This field shall be represented by an Entity Identifier Record (see 5.2.8).

b. Identification of Colliding Entity - This field shall identify the entity which has collided with the issuing entity. If the entity ID is unknown, this field shall contain zeros. This field shall be represented by an Entity Identifier Record (see 5.2.8).

c. Timestamp - This field shall specify the time at which the data in the PDU is valid. This field shall be represented by a timestamp (see 5.2.20).

d. 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 5.2.12).

e. Velocity Vector - This field shall contain the velocity vector (at the time the collision is detected) of the issuing entity. This field shall be represented by the Vector Record (see 5.2.21.3).

f. Mass - This field shall contain the mass of the issuing entity. This field shall be represented by a 64-bit floating point number representing kilograms.

g. Location - This field shall specify the location of the collision with respect to the entity with which the issuing entity collided. This field shall be represented by a Vector Record (see 5.2.21.1).

The Collision PDU is represented on Figure 5-31.

FIELD SIZE (bits)	COLLISION PDU FIELDS	
48	ISSUING ENTITY ID	SITE - 16 - bit unsigned integer
		HOST - 16 - bit unsigned integer
		ENTITY - 16 - bit unsigned integer
48	COLLIDING ENTITY ID	SITE - 16 - bit unsigned integer
		HOST - 16 - bit unsigned integer
		ENTITY - 16 - bit unsigned integer
32	TIME STAMP	32 - bit unsigned integer
48	EVENT ID	SITE - 16 - bit unsigned integer
		HOST - 16 - bit unsigned integer
		EVENT - 16 - bit unsigned integer
16	PADDING	16 bits unused
96	VELOCITY	x - 32 - bit floating pt
		y - 32 - bit floating pt
		z - 32 - bit floating pt
64	MASS	64 - bit floating pt
96	LOCATION (with respect to Entity)	x - 32 - bit floating pt
		y - 32 - bit floating pt
		z - 32 - bit floating pt

Figure 5-31. Collision PDU.

5.3.5.5 Electronic interaction. (see also 4.7.8).

5.3.5.5.1 Radar PDU. Radar emissions shall be communicated by issuing a Radar PDU. The Radar PDU shall contain the following fields:

a. Emitting Entity Identification - This field shall identify the entity that is operating the radar system represented in this PDU. This field shall be represented by an Entity Identifier Record (see 5.2.8).

b. Time of Emission - This field shall specify the time at which the data in the PDU is valid. This field shall be represented by a timestamp (see 5.2.20).

c. Event Identification - This field shall contain a number generated by the issuing simulator to identify fire control radar events with the firing of a weapon system. If this field is not used it shall contain the value zero. This field shall be represented by an Event Identifier Record (see 5.2.12).

d. Number of Radar Systems - This field shall specify the number of radar systems that the issuing entity possesses. This field shall be represented by an 8-bit unsigned integer. For each radar, the following fields shall be specified:

- Location - This field shall specify the location of the radar in the entity coordinates of the entity possessing the radar system. This field shall be represented by a Vector Record (see 5.2.21.1).

- Radar System - This field shall specify the type of radar system represented. This field shall be represented by a Radar System Data Type (see 5.2.15).

- Power - This field specifies the average power being transmitted in units of dBm (decibel-milliwatts). This field shall be represented by a 16-bit integer.

- Radar Mode - This field shall specify the mode that a particular radar is in. This

field shall be represented by an 8-bit enumeration (see Appendix F).

- Specific Data - This field is reserved for more complete information inclusion. Currently this field is not used. This field shall be represented by a 64-bit unsigned integer.

- Azimuth Center, Azimuth Half-angle, Elevation Center, Elevation Half-angle. These fields represent the angles required to describe the volume covered by the radar scan. All angles are measured in relation to entity coordinates with the origin described by the location field. The volume shall correlate with the radar mode field. These fields shall be represented as 32-bit unsigned integers representing BAMs.

- Number Illumined - This field shall specify the number of simulated entities that are illumined by a particular radar emission. This field shall be represented by an 8-bit unsigned integer (Note: This field was placed at a different location in Figure 5-32 to reduce the amount of padding required to maintain 32-bit boundaries).

  Based on the number illumined, the following fields are also included:

  (1) Target ID - This field shall specify the entity identifiers of the entities that are illumined by a particular radar. This field shall be made up of a list of Entity Identifier Records (see 5.2.8).

  (2) Radar Data - This field shall specify radar data specific to the applicable target ID. This field shall be specified by a 32-bit unsigned integer. Interpretation of these 32 bits has not been determined.

The Radar PDU is represented on Figure 5-32.

FIELD SIZE (bits)	RADAR PDU FIELDS		
48	EMITTING ENTITY ID	SITE - 16 - bit unsigned integer	
		HOST - 16 - bit unsigned integer	
		ENTITY - 16 - bit unsigned integer	
16	PADDING	16 bits unused	
32	TIME STAMP	32 - bit unsigned integer	
48	EVENT ID	SITE - 16 - bit unsigned integer	
		HOST - 16 - bit unsigned integer	
		EVENT - 16 - bit unsigned integer	
8	PADDING	8 bits unused	
8	Number of Radar Systems (n)	8 - bit unsigned integer	
n x ($96m_i$ + 352)	LOCATION (w/ respect to entity)	x - 32 - bit floating pt	352 bits
		y - 32 - bit floating pt	
		z - 32 - bit floating pt	
	RADAR SYSTEM	32 - bit unsigned integer	
	POWER	16 - bit integer	
	RADAR MODE	8 - bit enumeration	
	# ILLUMINED (m_i)	8 - bit unsigned integer	
	SPECIFIC DATA	64 - bit integer	
	SWEEP	Azimuth center - 32 - bit BAM	
		Azimuth sweep - 32 - bit BAM	
		Elevation center - 32 - bit BAM	
		Elevation sweep - 32 - bit BAM	
	TARGET ID	SITE - 16 - bit unsigned integer	96 x m_i bits (i = 1 to n)
		HOST - 16 - bit unsigned integer	
		ENTITY - 16 - bit unsigned integer	
	PADDING	16 bits unused	
	RADAR DATA	32 - bit unsigned integer	

Figure 5-32. Radar PDU.

6. NOTES

This section contains information of a general or explanatory nature that may be helpful, but is not mandatory.

6.1 Intended use. Protocol conforming to the requirements of this standard is intended for use in a Distributed Interactive Simulation. The purpose of this standard is to standardize the messages communicated between simulated entities in order to contribute to the interoperability of dissimilar simulators.

6.2 Description of Distributed Interactive Simulation.

6.2.1 Introduction. The term Distributed Interactive Simulation refers to an architectural approach in which the simulation is distributed across a number of independent and self-sufficient computers instead of one central computer. The term interactive refers to the fact that these computers constantly interact by sending messages about the current state of the simulation entities under their control, thereby allowing the other computers to incorporate these state changes into their simulations.

6.2.2 Definition. Distributed Interactive Simulation can be defined in the following manner:

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

6.2.3 DIS application. One application of DIS is the simulation of battle scenarios involving various vehicles simulators such as M1 tanks, F-16 fighter jets, or Navy ships. It is to this application that this standard is addressed.

6.2.4 Requirements for DIS. DIS has certain functional requirements. These requirements are to provide:

- Entity Information
- Entity Interaction
- DIS Management
- Environment Information

These requirements are described in detail in the Rationale Document that pertains to this standard. A brief description of each requirement is given in the paragraphs that follow.

6.2.4.1 Entity Information. Because of the great variety of simulated entities that can be involved in a single exercise, it is important to be able to communicate detailed information concerning each entity. This information includes the identity of the entity, its orientation, and how an entity might appear to others.

6.2.4.1.1 Types. The simulated entity could be a vehicle, a building, a munition (such as a missile) or a cloud of smoke. A method for identifying entities that would allow for a variety of different objects is needed.

6.2.4.1.2 Location and orientation. The issue of location and orientation of an entity is important for representation of the simulated entity by a computer. To help maintain network traffic within acceptable limits, the location and orientation information should contain velocity and sometimes acceleration. This information allows the receiving computer to model (dead reckon) the position of the entity over time without requiring constant updates over the network.

6.2.4.1.3 Appearances. The appearance of a simulated entity can be expressed in a number of ways. For example, in addition to visual appearance, the entity also has a certain infrared signature. If the exercise is taking place in the ocean, the sound an entity makes can identify it. Consequently, each entity has two "appearances": one is from the reflection of visible light and the other is from the emission of sound or electromagnetic energy such as heat, radar, radio and so forth.

6.2.4.2 Entity Interaction. Throughout a simulation exercise, entities interact with each other. This interaction may be represented by weapons fire, update threshold control, logistics support, collisions, or electronic interactions (for example, radar or sonar).

6.2.4.2.1 Weapons fire. When a simulated entity fires its weapon, its simulator needs to communicate the location of the firing weapon and the type of munition fired. Depending on the munition type, a determination of the detonation location is made. Given the munition type and the location of detonation, all simulators assess their own entity damage.

6.2.4.2.2 Update Threshold Control. The frequency with which one simulated platform must transmit an update of its

location and orientation to another platform often depends on what task the operator of the simulator is attempting to execute. If the operator of one platform is simply observing the other platform's motion for identification, the exact location of the platform is less critical and frequent updates are not required. However, if the operator is tracking the other platform in preparation for firing, or is maneuvering in formation, the exact location is critical and a higher update rate is required. DIS requires a means of controlling platform location and orientation update thresholds in order to meet the requirements for some critical operator tasks without overloading the network while the operator is executing less critical tasks.

6.2.4.2.3 Logistic Support. Certain services may be represented in a simulated exercise such as resupply or repair of vehicles. These functions and similar ones are provided by logistics support.

6.2.4.2.4 Collisions. It is necessary to represent the collision of entities in a simulation. When a collision occurs, both entities must be aware of the collision. Each determines any resulting damage.

6.2.4.2.5 Electronic Interaction. The development of technology in the area of sensory data has produced a variety of sensors and emitters ranging from the sonar of ships to the tracking radar of aircraft. Representation of these devices is essential in a simulation exercise.

6.2.4.3 DIS Management. Centralized control of a Distributed Interactive Simulation is necessary to manage the operation of the network hardware and certain aspects of the simulation exercise, and to allow for the gathering of performance measures. DIS management functions can be divided into three categories: Network Management, Simulation Management and Performance Measures.

6.2.4.3.1 Network Management. Network management functions handle the basic network functions such as load management, monitoring of nodes and gateways, and error recovery. The network manager would also have knowledge of entities on the network, such as entity location and network address. Analysis of network performance is also performed here.

6.2.4.3.2 Simulation Management. There is also a need for centralized control of the simulation exercise. Functions of simulation management include: Start, Restart, Maintenance, and Shutdown of the exercise. Other functions required include the introduction of late players and the collection and distribution of certain types of data.

6.2.4.3.3 Performance Measures. The purpose of distributed simulations is to either train individuals to work as a team, or to evaluate the performance of developmental hardware in a realistic operational scenario. In either case, some performance measures will be required beyond those that can be observed from responses transmitted over the communications lines.

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

6.2.4.4.1 Changes in the terrain. Changes in the terrain can be caused by a number of factors. These include engineering effects, such as the construction of a bridge or a building; weapons effects, which could destroy objects such as those created via engineering effects, as well as change the shape of the terrain through the impact of shells or explosion of mines; or natural effects such as flooding.

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

6.2.4.4.3 Degrees of ambient illumination. Night battles as well as day battles should be simulated.

6.2.4.4.4 Other environmental effects. Other environmental effects could also be considered. The effect of a nuclear blast located some distance away from a simulated entity might be represented. The effects of water temperature and salinity on the propagation of sound must be considered.

6.3 Interim Protocol

6.3.1 Activation and Deactivation of Simulated Entities. For DIS it is necessary to provide a means to introduce simulated entities into a simulation and to remove them. This capability is a DIS management function. Since DIS management protocol is not within the scope of this standard, an activate/deactivate function is recommended as an interim recommendation.

Information associated with activation and deactivation of a simulated entity in a DIS exercise should be communicated through the use of four PDUs: Activate Request PDU, Activate Response PDU, Deactivate Request PDU, and Deactivate Response PDU.

6.3.1.1 Activate Request PDU. The Activate Request PDU should be used by a host computer to introduce a platform entity into a DIS exercise.

6.3.1.1.1 Information contained in the Activate Request PDU. The Activate Request PDU contains the following information:

a. The identification of the host computer that is initiating the activation of the new entity.

b. The identification assignment to the new entity.

c. The time in the simulated world.

d. The reason for activation.

e. The identification of the terrain database to be used in this exercise.

f. The type of entity being activated.

g. The organizational unit to which the newly activated entity will belong.

h. The unique markings on the entity being activated.

i. The capabilities of the new entity.

j. The radar systems the new entity has.

k. The state of the vehicle subsystems. Currently the use of this field is optional. If the field is not used, it should contain zeros.

l. The initial location in the simulated world.

m. The initial orientation of the new entity.

n. Default update threshold values.

o. The stores on board the entity being activated.

p. The description and initial orientation of articulated and attached parts.

6.3.1.1.2 Issuance of the Activate Request PDU. The Activate Request PDU should be issued by a host computer to activate or initialize another host computer or simulation entity that intends to participate in the DIS exercise. This PDU should be issued when one or more of the following conditions exist:

a. The host computer is to activate an entity intending to join an exercise that is about to commence.

b. The host computer is to activate an entity intending to join an exercise that is already in progress.

c. The host computer is reactivating an entity that has been destroyed.

d. The host computer is re-initializing an entity to assign the entity different characteristics (for example, different type or different weapon systems).

6.3.1.1.3 Receipt of the Activate Request PDU. Upon receipt of the Activate Request PDU, the entity being requested to activate should respond by issuing an Activate Response PDU.

6.3.1.2 Activate Response PDU. The Activate Response PDU should be used to respond to an Activate Request PDU.

6.3.1.2.1 Information contained in the Activate Response PDU. The Activate Response PDU contains the following information:

a. The identification of the entity issuing the PDU.

b. The time limit indicating the maximum amount of time expected before the activated entity begins to issue Entity State PDUs.

c. An indication of whether or not the request has been accepted, and if not, the reason why.

6.3.1.2.2 Issuance of the Activate Response PDU. The Activate Response PDU should be issued by an entity upon receipt of an Activate Request PDU to inform the requesting entity whether or not the activate request has been accepted.

6.3.1.2.3 Receipt of the Activate Response PDU. Upon receipt of the Activate Response PDU, the activating entity should not take any action. The activating entity should note whether or not the activate request has been accepted. It should

be noted that the receipt of the Activate Response PDU does not indicate the entity has successfully completed the activation process.

6.3.1.3 Deactivate Request PDU. The Deactivate Request PDU should be used to withdraw entities from a DIS exercise. Entities may withdraw themselves or may be requested to withdraw by a simulation manager computer.

6.3.1.3.1 Information contained in the Deactivate Request PDU. The Deactivate Request PDU contains the following information:

a. The identification of the entity to be deactivated.

b. The reason for deactivation.

6.3.1.3.2 Issuance of the Deactivate Request PDU. The Deactivate Request PDU should be issued in either of the following instances:

a. A host computer intending to withdraw its own entity from the simulation exercise wishes to inform other host computers of its intention.

b. The simulation manager computer intending to request that a host computer cease simulating its entity wishes to inform the entity that it is being deactivated and seeks to inform other host computers of the deactivation. In this case, a response is required in the form of a Deactivate Response PDU.

6.3.1.3.3 Receipt of the Deactivate Request PDU. Upon receipt of the Deactivate Request PDU, if the PDU was issued for reason (a.) above, those receiving the PDU should cease to dead reckon and display the withdrawn entity. No response to this PDU is required.

If the PDU was issued for reason (b.) above, the host computer being requested to withdraw should respond immediately with a Deactivate Response PDU.

6.3.1.4 Deactivate Response PDU. The Deactivate Response PDU should be used when an entity is required to respond to a Deactivate Request PDU.

6.3.1.4.1 Information contained in the Deactivate Response PDU. The Deactivate Response PDU contains the following information:

a. The identification of the entity being requested for deactivation.

b. An indication of whether or not the request has been accepted, and if not, the reason why.

6.3.1.4.2 Issuance of the Deactivate Response PDU. The Deactivate Response PDU should be issued in response to a Deactivate Request PDU when an entity has been requested by another entity to withdraw its entity from the DIS exercise.

6.3.1.4.3 Receipt of the Deactivate Response PDU. Upon receipt of the Deactivate Response PDU, the deactivating entity should not take any action. It should cease to dead reckon the deactivating entity and note whether or not the deactivate request has been accepted.

6.3.2 PDUs associated with activation and deactivation

6.3.2.1 Activate Request PDU. Introduction of an entity to a simulation exercise should be performed by issuing an Activate request PDU. The Activate Request PDU contains the following fields:

a. Activating Host ID - This field identifies the host that is activating the new entity. This field is represented by a Simulation Address Record (see 5.2.8.1).

b. New Entity ID - This field identifies the entity. This field is represented by an Entity Identifier Record (see 5.2.8).

c. Date/Time - This field specifies the time and date that the simulation exercise is to begin in the simulated world. This field is represented by a Date/Time Record (see 5.2.6).

d. Reason - This field gives the reason for activating an entity. This field is represented as an 8-bit enumeration. Reasons for activation that have been defined are as follows:

(1) Other - any reason other than those listed below.

(2) Exercise start - the entity is being introduced into a new exercise.

(3) Exercise restart - the entity is being re-introduced to an exercise that has been interrupted.

(4) Reconstitution - the entity is being activated to provide it with a new location, new operational status.

e. Terrain Database ID - This field should identify the terrain database being used for this exercise. This field is represented by a Terrain Database Identifier (see 5.2.19).

f. Entity Type - This field describes the type of entity (platform, life form, and so forth) being activated, and is represented by an Entity Type Record (see 5.2.10).

g. Unit - This field identifies the unit with which an entity is associated. This field is represented by an Organizational Unit Record (see 5.2.14 and Appendix E).

h. Marking - This field identifies any unique markings on an entity (for example, a bumper number or country symbols). This field is represented by an Entity Marking Record (see 5.2.9).

i. Capabilities - This field identifies the entity capabilities. This field is represented by an Entity Capabilities Record (see 5.2.7).

j. Entity Subsystems - This field has not been specifically defined but has been provided for including specific information concerning an entity's subsystems. The standard sets aside 64 bits for this field.

k. Location - This field specifies the location for the new entity in world coordinates. This field is represented by a World Coordinates Record (see 5.2.22).

l. Orientation - This field describes the entity's orientation in terms of three angles. This field is represented by an Euler Angles Record (see 5.2.11).

m. Default Update Thresholds - This field specifies the default values for update thresholds (see 4.7.5.1).

Default values for linear thresholds should be represented by a Vector Record (see 5.2.21.1). Default values for rotational thresholds should be represented by an Euler Angles Record (see 5.2.11).

n. Number of Radar Systems - This field specifies the number of radar systems that the activated entity will have. This field is represented by an 8-bit unsigned integer.

o. Radar Systems - This field indicates the number and types of emitters this entity has. This field is represented by a Radar Systems Data Type (see 5.2.15).

p. Stores - This field describes the amount and types of stores of the given entity. The first field specifies the number of different types of stores. Based on this number of stores, the next field (or set of fields) is represented by one or more Supply Quantity Records.

(1) Number of Different Types of Stores - This field specifies the number of different types of stores an entity is to possess. This field is represented by an 8-bit unsigned integer.

(2) Stores - For each type of store (or supply) the type and amount of each is represented by a Supply Quantity Record (see 5.2.18).

q. Number of Articulated Parts - This field specifies the number of articulated parts. This field is represented by an 8-bit unsigned integer.

r. Articulated Parts. This field describes the orientation of each articulated part. This field is represented by an Articulated Part Record (see 5.2.3).

The Activate Request PDU is represented on Figure 6-1.

FIELD SIZE (bits)	ACTIVATE REQUEST PDU FIELDS	
32	ACTIVATE HOST ID	SITE - 16 - bit unsigned integer
		HOST - 16 - bit unsigned integer
16	PADDING	16 bit unused
48	NEW ENTITY ID	SITE - 16 - bits unsigned integer
		HOST - 16 - bit unsigned integer
		ENTITY - 16 - bit unsigned integer
32	DATE/TIME	32 - bit unsigned integer
8	REASON	8 - bit enumeration
24	PADDING	24 bits unused
96	TERRAIN D-BASED ID	11 element character string
		Version # - 8 - bit unsigned integer
64	ENTITY TYPE	Entity Kind - 8 - bit enumeration
		Domain - 8 - bit enumeration
		Country - 16 - bit enumeration
		Category - 8 - bit enumeration
		Subcategory - 8 - bit enumeration
		Specific - 8 - bit enumeration
		Extra - 8 - bit enumeration
96	UNIT	FORCE - 8 - bit enumeration
		COUNTRY - 16 - bit enumeration
		SERVICE - 8 - bit enumeration
		HIERARCHY - 8 x 8 bit enumeration
96	MARKING	Character set - 8 - bit enumeration
		11 element character string
32	ENTITY CAPABILITIES	32 bits of boolean fields

FIGURE 6-1. Activate Request PDU.

FIELD SIZE (bits)	ACTIVATE REQUEST PDU FIELDS (CONT'D)	
64	SUBSYSTEM PARAMETERS	64 bits undefined
96	ENTITY LOCATION	X COORDINATE - 64 - bit floating point
		Y COORDINATE - 64 - bit floating point
		Z COORDINATE - 64 - bit floating point
96	ORIENTATION	32 - bit BAM
		32 - bit BAM
		32 - bit BAM
192	DEFAULT UPDATE THRESHOLD VALUES	X COORDINATE - 32 - bit floating point
		Y COORDINATE - 32 - bit floating point
		Z COORDINATE - 32 - bit floating point
		32 - bit BAM
		32 - bit BAM
		32 - bit BAM
24	PADDING	24 bits unused
8	Number of Radar Systems (r)	8 - bit unsigned integer
r x 32	RADAR SYSTEMS	32 - bit unsigned integer
24	PADDING	24 bits unused
8	Number of Stores (s)	# of stores - 8 - bit enumeration
s x 96	STORES	STORE TYPE - ENTITY TYPE 64 bits
		QUANTITY - 32 - bit floating pt
24	PADDING	24 bits unused
8	Number of (n) Articulated Parts	8 - bit unsigned integer
n x ($96m_i$ + 64)	ARTICULATED PARTS	See Figure G-1, Appendix G.

n = # articulated parts
i = 1 to n
m_i = # of parameters for articulated part #

FIGURE 6-1. Activate Request PDU.

6.3.2.2 Activate Response PDU. Response to an Activate Request PDU should be communicated by the issue of an Activate Response PDU. The Activate Response PDU contains the following fields:

a. Entity Identification - This field identifies the entity receiving the Activate Request PDU. This field is represented by an Entity Identifier Record (see 5.2.8).

b. Time Limit - This field specifies an upper limit for the period of time a simulator responding to an Activate Request PDU has to begin issuing Entity State PDUs (see 5.3.4.1). If the request is not accepted, the time limit field should contain the value zero. This field is represented by a 16-bit unsigned integer specifying time in seconds.

c. Result - This field describes the result of the activate request. This field is represented by an 8-bit enumeration. Results of activation are defined as follows:

    (1) Other - Result other than those listed below.

    (2) Activate Request Accepted - Request for activation has been accepted.

    (3) Invalid Activate Parameter - Parameter in Activate Request PDU is not valid. Activate request not accepted.

    (4) Unexpected Activate Parameter - Parameter in Activate Request PDU is unexpected. Activate request not accepted.

The Activate Response PDU is represented on Figure 6-2.

FIELD SIZE (bits)	ACTIVATE RESPONSE PDU FIELDS	
48	ENTITY ID	SITE -16 - bit unsigned integer
		HOST-16 - bit unsigned integer
		ENTITY-16 - bit unsigned integer
16	TIME LIMIT	16 - bit unsigned integer
8	ACTIVATE RESULT	8 - bit enumeration
24	PADDING	24 bits unused

Figure 6-2. Activate Response PDU.

6.3.2.3 Deactivate Request PDU. Deactivation of a simulated entity should be performed by issuing a Deactivate Request PDU. The Deactivate Request PDU contains the following fields:

a. Entity Identification - This field identifies the entity to be deactivated. It is represented by an Entity Identifier Record (see 5.2.8).

b. Reason - This field provides the reason for deactivation of the entity. This data type is represented by an 8-bit enumeration. The reasons for deactivation are defined as follows:

(1) Other - Reason other than those listed below.

(2) Exercise end - End of the exercise.

(3) Entity Withdrawn - Entity withdrawn from the exercise.

(4) Entity Destroyed - Entity has been destroyed and no longer exists.

The Deactivate Request PDU is represented on Figure 6-3.

FIELD SIZE (bits)	DEACTIVATE REQUEST PDU FIELDS	
48	ENTITY ID	SITE -16 - bit unsigned integer
		HOST-16 - bit unsigned integer
		ENTITY-16 - bit unsigned integer
8	DEACTIVATE REASON	8 - bit enumeration
8	PADDING	8 bits unused

Figure 6-3. Deactivate Request PDU.

6.3.2.4 Deactivate Response PDU. Response to a Deactivate Request PDU should be communicated by issuing a Deactivate Response PDU. The Deactivate Response PDU contains the following fields:

a. Entity Identification - This field identifies the entity to be deactivated. It is represented by an Entity identifier Record (see 5.2.8).

b. Result - This field indicates whether the deactivation request has been accepted, and, if not, why. This field is an 8-bit enumeration. The results defined for deactivation are as follows:

(1) Other - Reason other than those listed below.

(2) Deactivate request accepted - Request for deactivation has been accepted.

(3) Invalid deactivate parameter - Parameter in the Deactivate Request PDU is invalid. Deactivate request not accepted.

(4) Unexpected deactivate reason - Reason for deactivation is unexpected. Deactivate request not accepted.

(5) Entity not active - Entity is presently not active. Deactivation not required.

The Deactivate Response PDU is represented on Figure 6-4.

FIELD SIZE (bits)	DEACTIVATE RESPONSE PDU FIELDS	
48	ENTITY ID	SITE -16 - bit unsigned integer
		HOST-16 - bit unsigned integer
		ENTITY-16 - bit unsigned integer
8	DEACTIVATE REASON	8 - bit enumeration
8	PADDING	8 bits unused

Figure 6-4. Deactivate Response PDU.

6.4 Proposed protocol. The following PDU is proposed to handle electromagnetic emissions in a DIS exercise. The protocol required by this standard can only support a limited number of radars. The proposed PDU is not a requirement but is recommended for adoption as part of a future draft of this standard.

6.4.1 Emitter Type Record. The types of emitters an entity has should be specified by an Emitter Type Record. This record should specify the emitter class, the mode number, and the database entry number. Each simulator should contain detailed emitter information in a database. The fields of this record are described in the subparagraphs that follow.

6.4.1.1 Emitter class. The emitter class should be specified by an 8-bit enumeration (see Appendix F).

6.4.1.2 Database entry number. The database entry number should be specified by a 16-bit unsigned integer. It should represent the number associated with the particular type of emitter.

6.4.1.3 Mode number. The mode number should be specified by an 8-bit unsigned integer. It should represent the mode of a particular emitter as listed in the referenced emitter data base.

The Emitter Type Record is represented on Figure 6-5.

Emitter Class	8-bit enumeration
Database Entry Number	16-bit unsigned integer
Mode Number	8-bit unsigned integer

FIGURE 6-5. Emitter Type Record.

6.4.2 Conditions for use of the Emitter PDU. The Emitter PDU should be issued by a simulator when the following conditions exist:

a. The discrepancy between an entity's actual location and its dead reckoned location exceeds a predetermined threshold. (See 4.7.5 and 5.3.5.2 on Update Threshold Control for more information on control of the threshold value.

b. A predetermined amount of time has elapsed since the issuing of the last Emitter PDU.

c. An emitter changes modes.

6.4.3 Emitter PDU. All simulators that require emitter information should have a database of information concerning the capabilities of certain types of emitters. Each Emitter PDU should provide information about the state of all of an entity's emitters for a particular database. If an emitter from another database is activated or deactivated, a separate Emitter PDU should be issued. An Emitter PDU should contain the following fields:

a. Emitting Entity Identification - This field should identify the entity that is issuing the Emitter PDU. This should be represented by an Entity Identifier Record (see 5.2.8).

b. Entity Type - This field should identify the entity type. This should be represented by an Entity Type Record (see 5.2.10).

c. Timestamp - This field should specify the time at which the data in the PDU is valid. This field should be represented by a timestamp (see 5.2.20).

d. Entity Location - This field should specify an entity's physical location in the simulated world. This field should be represented by a World Coordinates Record (see 5.2.22).

e. Dead Reckoning Parameters - This field should be used to provide parameters for dead reckoning the position and orientation of the entity. These parameters are currently undefined. The standard reserves 64 bits for these parameters.

f. Entity Velocity - This field should specify an entity's linear velocity. This field should be represented by a Vector Record (see 5.2.21.3).

g. Entity Acceleration - This field should specify an entity's acceleration. This field should be represented by a Vector Record (see 5.2.21.2).

h. Entity Orientation - This field should specify an entity's orientation. This field should be represented by an Euler Angles Record (see 5.2.11).

i. Number of emitters - The number of active emitters should be specified by an 8-bit unsigned integer.

j. Database number - The database number should be specified by an 8-bit unsigned integer, and should represent the particular database pertaining to the emitter(s) of interest (see Appendix F).

k. Emitter Type - This field should specify the class, the database entry number, and the mode number for each emitter. This should be represented by an Emitter Type Record (see 6.4.1).

l. Emitter Parameters - Emitter parameters should be used to further specify emitter operation beyond mode.

(1) Power - This field specifies the average power being transmitted in units of dBm (decibel-milliwatts). This field should be represented by a 16-bit unsigned integer.

(2) Three other parameters are to be determined. These should be represented as three fields of 16-bit floating point numbers.

m. Sweep - These fields (Azimuth Center, Azimuth Half-Angle, Elevation Center, Elevation Half-Angle) represent the angles required to describe the volume covered by directional emitters. For omnidirectional emitters, these fields should contain zeros. All angles are measured in relation to entity coordinates with the origin described by the Location field. Azimuth is measured in the entity's x-y plane. Elevation is measured in the entity's x-z plane. These fields should be represented as 32-bit unsigned integers representing BAMs.

The Emitter PDU is represented on Figure 6-6.

FIELD SIZE (bits)	EMITTER PDU FIELDS	
48	EMITTING ENTITY ID	SITE-16 - bit unsigned integer
		HOST-16 - bit unsigned integer
		ENTITY-16 - bit unsigned integer
16	PADDING	16 bits unused
64	ENTITY TYPE	64 bits of enumerated fields
32	TIME OF EMISSION	TIME STAMP- 32 - bit unsigned integer
192	ENTITY LOCATION	X COORDINATE- 64 - bit floating point
		Y COORDINATE- 64 - bit floating point
		Z COORDINATE- 64 - bit floating point
64	DEAD RECKONING PARAMETERS	64 bits undefined
96	ENTITY LINEAR VELOCITY	X - Comp.-32 - bit floating point
		Y - Comp.-32 - bit floating point
		Z - Comp.-32 - bit floating point
96	ENTITY LINEAR ACCELERATION	X - Comp.-32 - bit floating point
		Y - Comp.-32 - bit floating point
		Z - Comp.-32 - bit floating point
96	ENTITY ORIENTATION	Psi 32 - bit BAM
		Theta 32 - bit BAM
		Phi 32 - bit BAM
8	# OF EMITTERS (n)	8 - bit unsigned integer
8	DATABASE #	8 - bit unsigned integer
16	PADDING	16 bits unused
n x 224 (N=# OF EMITTERS)	EMITTER TYPE	CLASS-8 - bit enumeration
		DBASE ENTRY#- 16 - bit unsigned integer
		MODE#: 8 - bit unsigned integer
	EMITTER PARAMETERS	POWER: 16 - bit unsigned integer
		PARAMETER #1 TBD 16 - bit floating pt
		PARAMETER #2 TBD 16 - bit floating pt
		PARAMETER #3 TBD 16 - bit floating pt
	SWEEP (DIRECTIONAL EMITTERS ONLY)	Azimuth Center: 32 - bit BAM
		Azimuth Half-angle: 32 - bit BAM
		Elevation Center: 32 - bit BAM
		Elevation Half-angle: 32 - bit BAM

Figure 6-6. Emitter PDU.

6.5 Examples of logistic support

6.5.1 Resupply Service. If the service requested by the receiver is resupply, a scenario such as the following may take place:

A host computer that receives the Service Request PDU notices that one of its own entities is identified as the specified supplier. It responds by offering some portion of whatever supplies are currently loaded on that entity. This condition is shown on Figure 4-4 (see 4.7.6.2.2) as a transition from the Ready State to the Offering State. Meanwhile, the receiver's host computer re-issues its Service Request PDU every five seconds until such an offer is forthcoming. The offer takes the form of a Resupply Offer PDU issued by the supplier's host computer. The supplies offered should be a subset of those possessed by the supplier, and a subset of those requested by the receiver.

Upon receiving the offer of supplies, the receiver changes from the Requesting State to the Receiving State. The receiver then has up to one minute to acknowledge the receipt of those supplies by returning to the supplier a Resupply Received PDU listing the exact supplies taken. The receiver need not accept all of the supplies offered, but instead can indicate in its receipt acknowledgement just how much it did accept. After delaying up to one minute, the receiver issues its Resupply Received PDU and returns to the Ready State. When the supplier receives the Resupply Received PDU, it also returns to the Ready State, and the procedure is complete.

The time required to return the Service Received PDU, and the quantity of supplies reported by that PDU as taken, determine the rate at which the supplier and receiver are able to transfer munitions. For example, an M1 tank obtaining 105 mm shells from an ammunition supply truck might acknowledge receipt of a single round after 40 seconds; this results in a simulated rate of resupply for the M1 tank of one round every 40 seconds.

Throughout the transfer process, both the receiver's and the supplier's host computers continue to monitor the conditions necessary for the transfer. If any of these conditions cease to hold, either host computer can abort the transfer by issuing a Service Cancel PDU, with the result that no supplies are transferred (for the transfer that was in process). Alternatively, the receiver can terminate the transfer early but accept some of the supplies offered by issuing a Resupply Received PDU for the partial load. Finally, if the supplier waits in the Offering State for a full minute but receives no

Service Received PDU (perhaps the receiver has withdrawn from the exercise), it should return to the Ready State and assume that no supplies were taken.

6.5.2 Repair Service. If the service requested by the receiver is repair, a scenario such as the following may take place:

A host computer that receives the Service Request PDU and notices that one of its own entities is identified as the specified repairing entity can allow the crew of its entity to perform a repair on the receiver. This action is shown on Figure 4-6 (see 4.7.6.3.2) as a transition from the Ready State to the Offering state. As long as Service Request PDUs continue to be received, the repair process may be allowed to proceed and the repairing entity remains in the Offering State. However, if Service Request PDUs cease to be received and are not seen for a period of 12 seconds, the repairing entity's simulator must assume that the conditions for repair service are no longer all true. It must, therefore, abort the repair process. The entity then returns to the Ready State.

If the repair process successfully runs to completion, the repairing entity may then accomplish the repair by issuing a Repair Complete PDU to notify the receiver of the repair and returning to the Ready State. The receiver's simulator acknowledges the receipt of the Repair Complete PDU by returning a Repair Response PDU. (This acknowledgement simply indicates that the repair was performed, not that the repair was appropriate or that the disabled vehicle has been made operational because of the repair.)

6.6 Example of Update Threshold Control. If Update Threshold Control is used in DIS, a scenario such as the following may take place:

A host computer determines that it requires more frequent entity state information from another host computer. The first host computer (the requestor) issues an Update Threshold Request PDU to the second host computer (the responder). This condition is shown on Figure 4-1 (see 4.7.5.3) as a transition from the Ready State to the Requesting State.

The responder receives the Update Threshold Request PDU and determines that the requested threshold is less than the current threshold value that governs the rate at which Entity State PDUs are issued. The responder adopts the new thresholds and issues an Update Threshold Response PDU. This condition is shown on Figure 4-2 (see 4.7.5.3) as a transition from the Default Threshold State to the Altered Threshold State. A timer is set

to the amount of time indicated in the Update Threshold Request PDU.

If the responder receives another Update Threshold Request PDU requesting even smaller threshold values, the responder implements the newest request and resets the timer for the time requested. The responder issues another Update Threshold Response PDU. If the second request is for threshold values that are not smaller than those currently implemented, the responder issues an Update Threshold Response PDU indicating that the change is rejected and specifying the time remaining in the current implementation.

After the timer expires, the responder goes into the Hold Altered Threshold State and sets another timer for five seconds to allow the requestor or another host computer to request a threshold value that is less than the default. When this timer expires, the entity returns to the Default Threshold State.

6.7 Distributed Interactive Simulation Protocol Data Units expressed in Ada code. The following is an Ada-like[4] pseudo code representation of the basic data types and records and the Protocol Data Units discussed in section 5. The code is presented for clarification only and is not a requirement for use in the standard. Padding that is shown on the Figures in section 5 are not represented in the following code, although the padding is a requirement for this standard.

6.7.1 Basic data types and records.

6.7.1.1 Angles.

```
subtype bam_16 is unsigned_16;

subtype bam_32 is unsigned_32;
```

6.7.1.2 Angular Velocity Vector Record.

```
type angular_velocity_vector is record
   rate_x_axis  : signed_32;
   rate_y_axis  : signed_32;
   rate_z_axis  : signed_32;
end record;
```

6.7.1.3 Articulated Part Record.

```
type parts_param is record
   param_type  : unsigned_32;
   param_value : float_32;
end record;

subtype parts_quantity is unsigned_16;

type articulated_parts is record
   change         : unsigned_16;
   part_attached  : unsigned_16;
   parameters     :  array(integer range 1..parts_quantity)
                     of parts_param;
end record;
```

[4]Some of the Ada code constructs in this section do not follow Ada syntax rules.

6.7.1.4 Boolean. A boolean data type is a single bit representing a true-false value. It is an enumeration type of one bit, where the value 0 is interpreted as false and 1 as true.

6.7.1.5 Burst Descriptor Record.

WARHEAD AND FUZE LISTS ARE FOUND IN APPENDIX B

```
type burst_descriptor is record
   munition    : entity_type;
   warhead     : warhead_list;
   fuze        : fuze_list;
   quantity    : unsigned_16;
   rate        : unsigned_16;
end record;
```

6.7.1.6 Date/Time data type.

```
subtype date_time is unsigned_32;
```

6.7.1.7 Entity Capabilities Record.

```
type entity_capabilities is record
   ammunition_supply : boolean;
   fuel_supply       : boolean;
   misc_supply       : boolean;
   repair            : boolean;
   reserved          : binary_string(1..28);
end record;
```

6.7.1.8 Entity Identifier Record.

```
type simulation_address is record
   site : unsigned_16;
   host : unsigned_16;
end record;

type entity_id_type is record
   simulator : simulation_address;
   entity    : unsigned_16;
end record;
```

6.7.1.9 Entity Marking.

```
max_entity_marking_length : constant := 11;

type character_set_list is (ascii_character_set,
                            other);

type text_type is array(integer range
                        1..max_entity_marking_length)
                        of character;

type entity_marking is record
   character_set : character_set_list;
   text          : text_type;
end record;
```

6.7.1.10 Entity Type.

```
type entity_kind_list is (other,
                          platform,
                          munition,
                          life_form,
                          environmental,
                          cultural_feature);

type domain_list is (other,
                     land,
                     air,
                     surface,
                     subsurface,
                     space);

type country_list is (SEE APPENDIX F FOR COUNTRY LIST)
```

CATEGORY, SUBCATEGORY, SPECIFIC, AND EXTRA LISTS ARE DEPENDENT UPON THE KIND FIELD AND ARE FOUND IN APPENDIX H2

```
type entity_type is record
   kind        : entity_kind_list;
   domain      : domain_list;
   country     : country_list;
   category    : category_list;
   subcategory : subcategory_list;
   specific    : specific_list;
   extra       : extra_list;
end record;
```

6.7.1.11 Euler angles: Representation of entity orientation.

```
type euler_angles is record
   psi   : bam_32;
   theta : bam_32;
   phi   : bam_32;
end record;
```

6.7.1.12 Event Identifier Record.

```
type event_id is record
   address         : simulation_address;
   event_id_number : unsigned_16;
end record;
```

6.7.1.13 Exercise identifier.

```
subtype exercise_id is unsigned_8;
```

6.7.1.14 Organizational Unit Record.

```
   type force_type_list is (other,
                            blue_forces,
                            red_forces,
                            white_forces);

   SEE APPENDIX F FOR country_id_list

   type service_type_list is (service_other,
                              service_resupply,
                              service_repair,
                              service_tow,
                              service_rescue);
```

```
type unit_type_list is (unit_type_irrelevant,
                        unit_type_1,
                        unit_type_2,
                        unit_type_3,
                        unit_type_4,
                        unit_type_5,
                        unit_type_6,
                        unit_type_7,
                        unit_type_8);

type organizational_unit is record
   force_type        : force_type_list;
   country           : country_id_list;
   service           : service_type_list;
   unused            : binary_string(1..8);
   hierarchy         : unit_type_list;
end record;
```

6.7.1.15 Radar system.

```
subtype radar_system is unsigned_32;
```

6.7.1.16 Repair type.

REPAIR TYPE LIST IS FOUND IN APPENDIX C

6.7.1.17 Service type.

```
type service_type_list is (service_other,
                           service_resupply,
                           service_repair,
                           service_tow,
                           service_rescue);
```

6.7.1.18 Supply Quantity Record.

```
type supply_quantity is record
   supply   : entity_type;
   quantity : float_32;
end record;
```

6.7.1.19 Terrain database identifier.

```
max_terrain_name_length : constant := 11;

type terrain_name_type is array(integer range
                                1..max_terrain_name_length)
                                of character;

type terrain_database_id is record
   terrain_name     : terrain_name_type;
   version          : unsigned_8;
end record;
```

6.7.1.20 Time stamp: Representation of time.

```
subtype time_stamp is unsigned_32;
```

6.7.1.21 Vector Record.

```
type vector is record
   first_comp  : float_32;
   second_comp : float_32;
   third_comp  : float_32;
end record;
```

6.7.1.22 World Coordinates Record.

```
type world_coordinates is record
   x : float_64;
   y : float_64;
   z : float_64;
end record;
```

6.7.2 Distributed Interactive Simulation Protocol Data Units.

6.7.2.1 PDU header.

```
type pdu_type_list is   (entity_state,
                         fire,
                         detonation,
                         service_request,
                         resupply_offer,
                         resupply_received,
                         resupply_cancel,
                         repair_complete,
                         repair_response,
                         update_threshold_request,
                         update_threshold_response,
                         collision,
                         radar,
                         activate_request,
                         activate_response,
                         deactivate_request,
                         deactivate_response,
                         emitter);

type pdu_header is record
   version  : unsigned_8;
   exercise : exercise_id;
   pdu_type : pdu_type_list;
end record;
```

6.7.2.2 Entity State PDU.

```
type entity_state_pdu is record
   entity_id           : entity_id_type;
   entity              : entity_type;
   time                : time_stamp;
   location            : world_coordinates;
   velocity            : velocity_vector;
   acceleration        : acceleration_vector;
   orientation         : euler_angles;
   angular_rates       : angular_velocity_vector;
   deadrec_param       : unsigned_64;
   dynamic_appearance  : unsigned_32;
   marking             : entity_marking;
   capabilities        : entity_capabilities;
   parts               : articulated_parts;
end record;
```

6.7.2.3 Fire PDU.

```
type fire_pdu is record
   firing_entity     : entity_id_type;
   target            : entity_id_type;
   munition_id       : entity_id_type;
   event_id          : event_id_type;
   time              : time_stamp;
   location          : world_coordinates;
   burst             : burst_descriptor;
   velocity          : velocity_vector;
   munition_range    : unsigned_32;
end record;
```

6.7.2.4 Detonation PDU.

```
type detonation_result_list is     (other,
                                    impact,
                                    no_detonation_impact);

type detonation_pdu is record
   attacker_id       : entity_id_type;
   target_id         : entity_id_type;
   munition_id       : entity_id_type;
   event_id          : event_id_type;
   time              : time_stamp;
   location_world    : world_coordinates;
   burst             : burst_descriptor;
   velocity          : velocity_vector;
   location_target   : vector;
   result            : detonation_result;
   energy            : float_32;
   directionality    : float_32;
   momentum          : float_32;
end record;
```

6.7.2.5 Update Threshold Request PDU.

```
type update_threshold_request_pdu is record
   requestor_id      : entity_id_type;
   responder_id      : entity_id_type;
   linear_threshold  : vector;
   angular_threshold : euler_angles;
   duration          : unsigned_32;
end record;
```

6.7.2.6 Update Threshold Response PDU.

```
type update_result is (other,
                       change_accepted,
                       change_inappropriate,
                       change_not_implemented);

type update_threshold_response_pdu is record
   responder_id         : entity_id_type;
   requestor_id         : entity_id_type;
   update_change_result : update_result;
   time_remaining       : unsigned_8;
end record;
```

6.7.2.7 Service Request PDU.

```
type service_type_list is (service_other,
                           service_resupply,
                           service_repair,
                           service_tow,
                           service_rescue);

type service_request_pdu is record
   requestor_id    : entity_id_type;
   supplier_id     : entity_id_type;
   service_type    : service_type_list;
   number_supplies : unsigned_8;
   supplies        : array(integer range 1..number_supplies)
                     of supply_quantity;
end record;
```

6.7.2.8 Resupply Offer PDU.

```
type resupply_offer_pdu is record
   receiver_id        : entity_id_type;
   supplier_id        : entity_id_type;
   number_supplies    : unsigned_8;
   available_supplies : array(integer range 1..number_supplies)
                              of supply_quantity;
end record;
```

6.7.2.9 Resupply Received PDU.

```
type service_complete_pdu is record
   receiver_id          : entity_id_type;
   supplier_id           : entity_id_type;
   number_supplies       : unsigned_8;
   supplies_transferred : array(integer range 1..number_supplies)
                           of supply_quantity;
end record;
```

6.7.2.10 Resupply Cancel PDU.

```
type resupply_cancel_pdu is record
   requestor_id : entity_id_type;
   supplier_id  : entity_id_type;
end record;
```

6.7.2.11 Repair Complete PDU.

```
subtype repair_type_list is unsigned_16;

type repair_offer_pdu is record
   receiver_id   : entity_id_type;
   repairer_id   : entity_id_type;
   repair_type   : repair_type_list;
end record;
```

6.7.2.12 Repair Response PDU.

```
type repair_result_list is (other,
                            repair_completed,
                            invalid_repair,
                            repair_interrupted);

type repair_response_pdu is record
   receiver_id   : entity_id_type;
   repairer_id   : entity_id_type;
   repair_result : repair_result_list;
end record;
```

6.7.2.13 Collision PDU.

```
type collision_pdu is record
   entity_id      : entity_id_type;
   colliding_id   : entity_id_type;
   time           : time_stamp;
   event          : event_id;
   velocity       : vector;
   mass           : float_64;
   location       : vector;
end record;
```

6.7.2.14 Radar PDU.

```
type mode_list is (unused,
                   search,
                   doppler_hprf,
                   doppler_mprf,
                   doppler_lprf,
                   monopulse,
                   acquisition,
                   tracking,
                   track_while_scan,
                   terrain_follow,
                   data_link);

subtype radar_data is unsigned_32;

type radar_info is record
   location              : vector;
   system                : radar_system;
   power                 : unsigned_16;
   mode                  : mode_list;
   specific              : unsigned_64;
   azimuth_center        : bam_32;
   azimuth_half_angle    : bam_32;
   elevation_center      : bam_32;
   elevation_half_angle  : bam_32;
   number_illumined      : unsigned_8;
   targets               : array(integer range 1..number_illumined)
                           of radar_data;
end record;
```

```
type radar_pdu is record
   emitting_entity   : entity_id_type;
   time_emission     : time_stamp;
   event             : event_id;
   number_systems    : unsigned_8;
   system_specific   : array(integer range 1..number_systems)
                       of radar_info;
end record;
```

6.7.2.15 Interim PDUs.

6.7.2.15.1 Activate Request PDU.

```
type reason_list is (other,
                     exercise_start,
                     exercise_restart,
                     reconstitution);

type stores is record
   number_of_stores  : unsigned_8;
   stores            : array(integer range 1..number_of_stores)
                       of supply_quantity;
end record;

type threshold_values is record
   linear_thresholds     : vector;
   rotational_thresholds : euler_angles;
end record;
```

```
type activate_request_pdu is record
   activating_host     : simulation_address;
   activated_entity    : entity_id_type;
   simulated_time      : date_time;
   reason              : reason_list;
   terrain_database    : terrain_database_id;
   entity              : entity_type;
   unit                : organizational_unit;
   marking             : entity_marking;
   capabilities        : entity_capabilities;
   subsystems          : unsigned_64;
   location            : world_coordinates;
   orientation         : euler_angles;
   default_thresholds  : threshold_values;
   number_radars       : unsigned_8;
   radars              : array(integer range 1..number_radars)
                         of radar_system;
   stores_on_board     : stores;
   number_parts        : unsigned_8;
   parts_position      : articulated_parts;
end record;
```

6.7.2.15.2 Activate Response PDU.

```
type result_list is (other,
                     activate_request_accepted,
                     invalid_activate_parameter,
                     unexpected_activate_parameter);

type activate_response_pdu is record
   activating_entity    : entity_id_type;
   time_limit           : unsigned_16;
   result               : result_list;
end record;
```

6.7.2.15.3 Deactivate Request PDU.

```
type deactivate_reason is (other,
                           exercise_end,
                           entity_withdrawn,
                           entity_destroyed);
```

```
type deactivate_request_pdu is record
   deactivating_entity  : entity_id_type;
   reason               : deactivate_reason;
end record;
```

6.7.2.15.4 Deactivate Response PDU.

```
type deactivate_result is (other,
                           deactivate_request_accepted,
                           invalid_deactivate_reason,
                           unexpected_deactivate_reason,
                           entity_not_active);

type deactivate_response_pdu is record
   deactivating_entity  : entity_id_type;
   result               : deactivate_result;
end record;
```

6.7.2.16 Recommended PDU: The Emitter PDU.

```
type emitter_class_list is (other,
                            sound,
                            infrasonic,
                            very_low_frequencies,
                            low_frequencies,
                            medium_frequencies,
                            high_frequencies,
                            very_high_frequencies,
                            ultra_high_frequencies,
                            super_high_frequencies,
                            extremely_high_frequencies,
                            infrared,
                            visible,
                            ultraviolet,
                            x_rays,
                            gamma_rays,
                            cosmic_rays);


type emitter_type is record
   emitter_class      : emitter_class_list;
   mode_number        : unsigned_8;
   entry_number       : unsigned_16;
end record;
```

```
type emitter_parameter_type is record
   power         : unsigned_16;
   parameter_2   : unsigned_16;
   parameter_3   : unsigned_16;
   parameter_4   : unsigned_16;
end record;

type sweep_angles is record
   azimuth_center       : bam_32;
   azimuth_half_angle   : bam_32;
   elevation_center     : bam_32;
   elevation_half_angle : bam_32;
end record;


type emitter_pdu is record
   emitting_entity    : entity_id_type;
   entity             : entity_type;
   time_of_emission   : time_stamp;
   location           : world_coordinates;
   dr_parameters      : unsigned_64;
   velocity           : vector;
   acceleration       : vector;
   orientation        : euler_angles;
   number_emitters    : unsigned_8;
   database_number    : unsigned_8;
   emitter            : emitter_type;
   emitter_parameters : emitter_parameter_type;
   sweep              : sweep_angles;
end record;
```

APPENDIX A

The Open Systems Interconnection Reference Model

10. SCOPE.

10.1 Scope. This Appendix gives an overview of the Open Systems Interconnection (OSI) Reference Model. This Appendix is not a mandatory part of the standard. The information contained herein is intended for guidance only.

20. APPLICABLE DOCUMENTS. This section is not applicable to this appendix.

30. DETAILS.

30.1 OSI Reference Model. The design of a computer network consists of different layers or levels. Each layer is built upon its predecessor and is responsible to provide services to the higher layers in a manner transparent to the higher layers. Different networks may have a different number of layers or different functions within the layers. In 1984, the OSI Reference Model was developed by the International Organization for Standardization (ISO) as a model of a computer communications architecture. The model is 'Open' because it refers to systems that are open for communication with other systems. The OSI Reference Model is shown on Figure A-1.

APPENDIX A

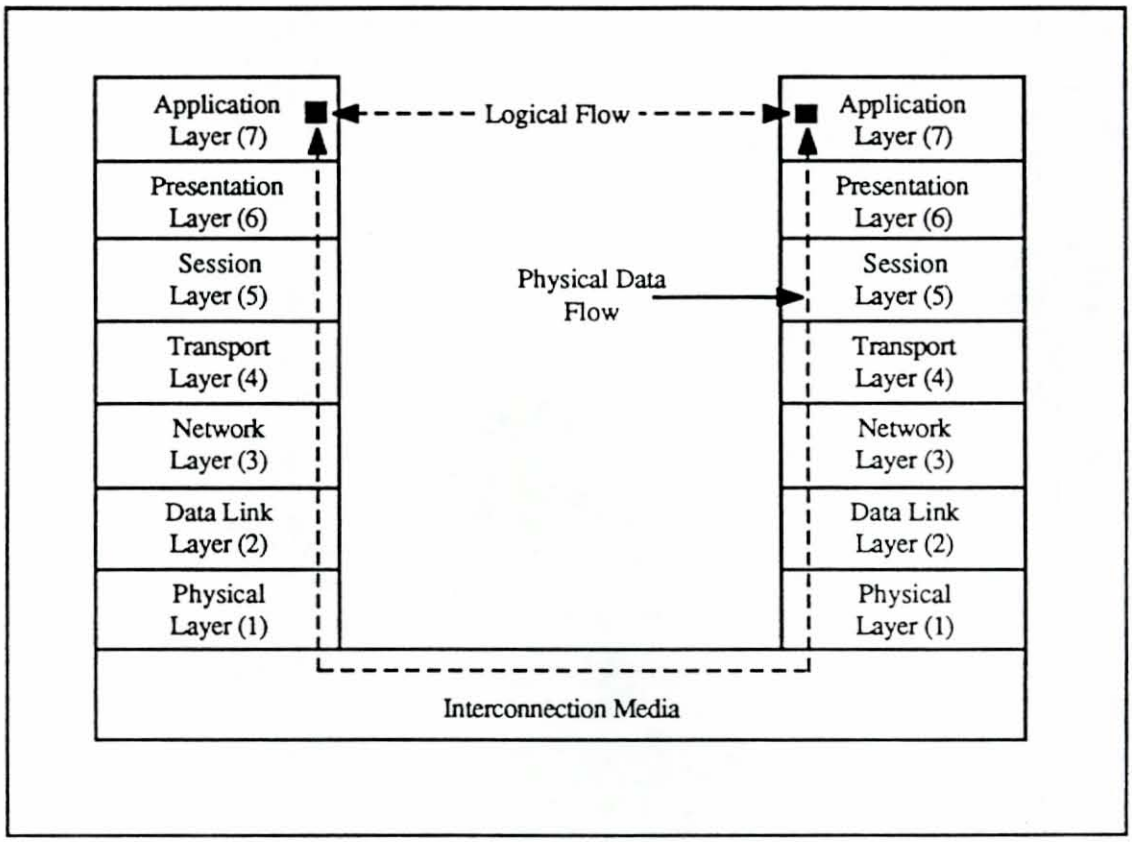


FIGURE A-1. Open Systems Interconnection Reference Model.

APPENDIX A

30.2 Layer functions in the OSI Reference Model. It is important to understand that OSI is not an architecture in and of itself. The intent of the OSI model is that protocols be developed to perform the functions of each layer. The functions provided by each layer, as presented by Tannenbaum[1], are summarized in Table I.

TABLE I. Functions of OSI Layers

LAYER	FUNCTION
1-Physical	Concerns the transmission of unstructured bit stream over physical medium; deals with the mechanical, electrical, functional, and procedural characteristics to access the physical medium
2-Data Link	Provides for the reliable transfer of information across the physical link; sends blocks of data (frames) with the necessary synchronization, error control, and flow control.
3-Network	Provides upper layers with independence from the data transmission and switching technologies used to connect systems; responsible for establishing, maintaining, and terminating connections.
4-Transportation	Provides reliable, transparent transfer of data endpoints; provides end-to-end error recovery and flow control.
5-Session	Provides the control structure for communication between applications; establishes, manages, and terminates connections (sessions) between cooperating applications.
6-Presentation	Provides independence to the application processes from differences in data representation (syntax).
7-Application	Provides access to the OSI environment for users and also provides distributed information services.

[1]Tannenbaum, Computer Networks. Prentice Hall: 1988.

APPENDIX B

Fuze, Warhead and Supply Types

10. SCOPE.

10.1 Scope. This Appendix specifies the interpretation of the two 16-bit enumerations that represent the Fuze (previously called Detonator) and Warhead types specified by this standard. This Appendix also specifies the interpretation of enumeration fields for supply types that are not munitions. This Appendix is a mandatory part of the standard. The information contained herein is intended for compliance.

20. APPLICABLE DOCUMENTS. This section is not applicable to this appendix.

30. DETAILS.

30.1 Fuze and Warhead. Munitions are defined beyond munition type in terms of Fuze and Warhead type. The values presently defined for this field are as follows:

Fuze (Detonator)

Field Value	Type
0000	Other
1000	Contact
1100	Contact, instant
1200	Contact, Delayed
2000	Timed
3000	Proximity
4000	Command
5000	Altitude
6000	Depth
7000	Acoustic

APPENDIX B

Warheads

Field Value	Warhead
0000	Other
1000	High Explosive
1100	High Explosive, Plastic
1200	High Explosive, Incendiary
1300	High Explosive, Fragmentation
1400	High Explosive, Anti-Tank
1500	High Explosive, Bomblets
1600	High Explosive, Shaped Charge
2000	Smoke
3000	Illumination
4000	Practice
5000	Kinetic
6000	Unused
7000	Nuclear
8000	Chemical, General
8100	Chemical, Blister agent
8200	Chemical, Blood agent
8300	Chemical, Nerve agent
9000	Biological, General

30.2 Supply types. In the Logistic Support resupply service PDU's (see 5.3.5.3), munitions are defined using the munition types defined in Appendix H1 and H2. For consistency, supplies other than munitions using these PDU's are defined using the same classifications of Kind, Domain, Country, Category, and Designation. The Specific and Extra fields are unused and shall contain the value zero. The values presently defined for these fields are as follows:

Kind	Domain	Country	Category	Designation
7 Supplies				
	Unused			
		Unused		
			0 Other	
			1 Fuels	
				0 Other
				1 Gasoline
				2 Diesel Fuel
				3 JP4
				4 Fuel Oil
			2 Oils	
				0 Other
				1 TBD
			3 Lubricants	
				0 Other
				1 TBD
			4 Food	
				0 Other
				1 TBD
			5 Spare Parts	
				0 Other
				1 TBD

APPENDIX C

Defined Repair Types

10. SCOPE.

10.1 Scope. This Appendix defines the different Repair Types specified by this standard. This Appendix is a mandatory part of the standard. The information contained herein is intended for compliance.

20. APPLICABLE DOCUMENTS. This section is not applicable to this appendix.

30. DETAILS.

The following repair types are defined for all platform types.

30.1 General repair codes. These codes applies to all entities that may require repair:

Field Value	Meaning
0	no repairs performed
1	repair everything

30.2 Subsystems. Platforms consist of a number of subsystems. Within these major subsystems, repairs can be defined.

30.2.1 Propulsion systems.

Field Value	System to Repair
10	motor
20	starter
30	alternator
40	generator
50	battery
60	engine coolant leak
70	fuel filter
80	transmission oil leak
90	engine oil leak
100	pumps
110	filters
120-999	Other propulsion systems to be defined

30.2.2 Hull/Airframe/Body.

Field Value	System to Repair
1000	hull
1010	airframe
1020-1499	other hull/airframe/body types to be defined

30.2.3 Interface to environment systems.

Field Value	System to Repair
1500	propeller
1510	transmission
1520	filters
1530	brakes
1540-1999	other interface to environment systems to be defined

30.2.4 Weapon systems.

Field Value	System to Repair
2000	gun elevation drive
2010	gun stabilization system
2020	gunner's primary sight (GPS)
2030	commander's extension to the GPS
2040	loading mechanism
2050-3999	other weapons systems to be defined

30.2.5 Fuel systems.

Field Value	System to Repair
4000	fuel transfer pump
4010	fuel lines
4020	gauges
4030-4499	other fuel systems to be defined

30.2.6 Electronic and communications systems.

Field Value	System to Repair
4500	electronic counter measure systems
4600	electronic warfare systems
4700	laser range finder
4800	radios
4900	intercoms
5000	coders
5100	decoders
5200	lasers
5300	computers
5400	emitters
5600	detection systems
5601-7999	other electronic and communications systems to be defined

30.2.7 Life support systems.

Field Value	System to Repair
8000	air supply
8010	filters
8020	water supply
8030	refrigeration system
8040	chemical, biological, and radiologic protection
8050	water wash down systems
8060	decontamination systems
8070-8999	other life support systems to be defined

30.2.8 Hydraulics systems and actuators.

Field Value	System to Repair
9000	water supply
9010	cooling system
9020	winches
9030	catapults
9040	cranes
9050	launchers
9060-9999	other hydraulics systems and actuators to be defined

30.2.9 Auxiliary craft.

Field Value	System to Repair
10000	life boats
10010	landing craft
10020	ejection seats
10030-10499	other auxiliary craft to be defined

APPENDIX D

Appearance Field Bit Definition

10. SCOPE.

10.1 Scope. This Appendix defines the different Appearances specified by the appearance field in the Entity State PDU. This Appendix is a mandatory part of the standard. The information contained herein is intended for compliance.

20. APPLICABLE DOCUMENTS. This section is not applicable to this appendix.

30. DETAILS.

This appendix defines the 32-bit Appearance field used in the Entity State PDU to describe changes to entities' appearance attributes.

30.1 Platforms. The 32 bits are defined for a platform depending on the domain it operates in. The following sections describe the bit definition of this field for all domains. The Domain field is defined in the Entity Type Record in 5.2.10.

30.1.1 Platforms of the land domain. The 32-bit appearance field for land platforms is defined as follows:

Name	Bits	Purpose
Destroyed	0	is 1 if platform is destroyed, 0 otherwise
Smoke_plume	1	is 1 if a smoke plume is rising from the platform, 0 otherwise
Flaming	2	is 1 if flames are rising from the platform, 0 otherwise
Dust_cloud	3-4	describes any dust cloud being raised by the platform: 0: no dust cloud 1: small dust cloud 2: medium dust cloud 3: large dust cloud

Paint_scheme	5-6	describes the paint scheme of the platform: 0: not applicable 1: winter colors 2: desert colors
Launcher	7	is 1 if the platform's missile launcher is raised, 0 otherwise
Engine_smoke	8	is 1 if the platform is emitting engine smoke, 0 otherwise
Hatch	9-11	describes whether the hatch is open: 0: not applicable 1: hatch is closed 2: hatch is open 3: hatch is open and commander is looking out
	12-29	other appearance attributes specific to land platforms can be defired

30.1.2 Platforms of the Air Domain. The 32-bit appearance field for air platforms is defined as follows:

Name	Bits	Purpose
Destroyed	0	is 1 if platform is destroyed, 0 otherwise
Flaming	1	is 1 if flames are rising from the platform, 0 otherwise
After_burner	2	is 1 if after burner is on, 0 otherwise
Running_lights	3	is 1 if running lights are on, 0 otherwise
Speed_brake	4	is 1 if brake is deployed, 0 otherwise
	5-31	other appearances attributes specific to air platforms can be defined

30.1.3 Platforms of the Surface Domain. The 32-bit appearance field for surface platforms is defined as follows:

Name	Bits	Purpose
Destroyed	0	is 1 if platform is destroyed, 0 otherwise
Flaming	1	is 1 if flames are rising from the platform, 0 otherwise
wake	2-4	describes the presence and size of bow wake around the platform: 0: none 1: small 2: medium 3: large
Running_lights	5	is 1 if running lights are on, 0 otherwise
	6-31	appearances attributes specific to surface platforms

30.1.4 Platforms of the Sub-surface Domain. The 32-bit appearance field for sub-surface platforms is defined as follows:

Name	Bits	Purpose
Destroyed	0	is 1 if platform is destroyed, 0 otherwise
wake	1-3	describes the presence and size of bow wake around the platform: 0: none 1: small 2: medium 3: large
Running_lights	4	is 1 if running lights are on, 0 otherwise

Name	Bits	Purpose
Hatch	5-7	describes whether the hatch is open: 0: not applicable 1: hatch is closed 2: hatch is open 3: hatch is open and commander is looking out
	8-31	appearances attributes specific to sub-surface platforms

30.1.5 Platforms of the Space Domain. The 32-bit appearance field for space platforms is defined as follows:

Name	Bits	Purpose
Destroyed	0	is 1 if platform is destroyed, 0 otherwise
	1-31	appearances attributes specific to space platforms

30.2 Munitions.

30.2.1 Guided munitions. The 32-bit appearance field for guided munitions is defined as follows:

Name	Bits	Purpose
Launch_flash	0	is 1 if launch is producing a flash, 0 otherwise
Rocket_flame	1	is 1 if flame is present, 0 otherwise
	2-31	appearances attributes specific to munitions

30.3 Life forms.

30.3.1 Life forms. The 32-bit appearance field for Life Forms is defined as follows:

Name	Bits	Purpose
Lifeform_state	0-3	describes the state of the life form: 0: destroyed 1: upright 2: kneeling 3: prone
Weapon_1[2]	4	is 1 if weapon_1 is deployed, 0 if weapon is stowed
Weapon_2	5	is 1 if weapon_2 is deployed, 0 if weapon is stowed
	6-31	appearances attributes specific to life forms

30.4 Environmentals.

30.4.1 Environmentals. The 32-bit appearance field for Environmentals is defined as follows:

Name	Bits	Purpose
Density	0-3	describes the density of the environmental: 0: clear 1: hazy 2: dense 3: very dense 4: opaque
	4-31	appearances attributes specific to environmentals

[2]Weapon_1 is the life form's primary weapon. If the life form has a secondary weapon, this is specified by Weapon_2 (bit 5).

30.5 Cultural features.

30.5.1 Cultural features. The 32-bit appearance field for cultural features is defined as follows:

Name	Bits	Purpose
Damage	0-2	describes the damaged appearance of a cultural feature: 0: none 1: slight damage 2: moderate damage 3: destroyed
Smoke_plume	1	is 1 if a smoke plume is rising from the cultural feature, 0 otherwise
Flaming	2	is 1 if flames are rising from the cultural feature, 0 otherwise
	3-31	appearances attributes specific to cultural features

APPENDIX E

ORGANIZATIONAL UNIT

10. SCOPE.

10.1 Scope. This Appendix defines the types of Organizational Units to which an entity can belong. This Appendix is a mandatory part of the standard. The information contained herein is intended for compliance.

20. APPLICABLE DOCUMENTS. This section is not applicable to this appendix.

30. DETAILS.

30.1 Hierarchy of units. Each unit in the simulation belongs to a hierarchy of organizational units. The hierarchy shall consist of the following components:

Force
Country
Service
Not Used
Service Hierarchy

30.1.1 Force ID. Force shall refer to the highest level organizational component. This field shall be an 8-bit enumeration and shall have the following values:

Field Value	Force
0	Other
1	Blue Forces (U.S., NATO, SEATO, Etc.)
2	Red Forces (USSR, Warsaw Pact, Medellin Cartel, Etc.)
3	White Forces (Neutral)

30.1.2 Country ID. Country shall refer to the country to which this unit belongs. This field shall be an 8-bit enumeration (see Appendix F).

30.1.3 Service ID. Service shall refer to the military service to which the unit belongs. This field shall be an 8-bit enumeration and shall have the following values:

Field Value	Service ID
0	Other
1	Army
2	Air Force
3	Coast Guard
4	Marines
5	Navy

30.1.4 Hierarchy. The service hierarchy shall consist of eight 8-bit unsigned integers that, starting at the top of the hierarchy, give the unit organizational number at each level of the hierarchy. The representation of the eight fields will depend on the service represented.

30.1.4.1 Army. For Army units, the unit organizational numbers shall be listed in the following order:

Army Number
Corps Number
Division Number
Regiment Number
Battalion Number
Company Number
Platoon Number
Squad Number

FIGURE E-1. Army Organizational Order.

30.1.4.2 Air Force. For Air Force units, the unit organizational numbers shall be listed in the following order:

Air Force Number
Wing Number
Squadron Number
Flight Number
Division Number
Section Number
Spare
Spare

FIGURE E-2. Air Force Organizational Order.

30.1.4.3 Coast Guard. For Coast Guard Units, the unit organizational numbers have not been determined.

30.1.4.4 Marines. For Marine units, the unit organizational numbers shall be listed in the following order:

Fleet Number
Marine Expeditionary Force Number
Marine Expeditionary Brigade Number
Marine Expeditionary Unit Number
Spare
Spare
Spare
Spare

FIGURE E-3. Marine Organizational Order.

30.1.4.5 Navy. For Navy units, the unit organizational numbers shall be listed in the following order:

Fleet Number
Force Number
Group Number
Unit Number
Element Number
Spare
Spare
Spare

FIGURE E-4. Navy Organizational Order.

APPENDIX F

Enumeration and Bit-encoded Values

10. SCOPE.

10.1 Scope. This Appendix defines the enumeration values and the bit-encoded values for DIS Version 1.0. This Appendix is a mandatory part of the standard. The information contained herein is intended for compliance.

20. APPLICABLE DOCUMENTS. This section is not applicable to this appendix.

30. DETAILS.

30.1 Enumeration values and bit-encoded values for DIS required PDUs.

30.1.1 Appearance field bit definitions. See Appendix D for these values.

30.1.2 Country ID. Countries shall be identified as follows:

Field Value	Country
0	Other
1	Afghanistan
2	Albania
3	Algeria
4	Andorra
5	Angola
6	Antarctica
7	Antigua and Barbuda
8	Argentina
9	Australia
10	Austria
11	Bahamas
12	Bahrain
13	Bangladesh
14	Barbados
15	Belgium
16	Belize
17	Benin
18	Bhutan
19	Bolivia
20	Botswana
21	Brazil
22	Brunei
23	Bulgaria

30.1.2 Country ID (cont).

Field Value	Country
24	Burkina Faso
25	Burma
26	Burundi
27	Cambodia/Kampuchea
28	Cameroon
29	Canada
30	Cape Verde Islands
31	Central African Republic
32	Chad
33	Chile
34	People's Republic of China
35	Taiwan
36	Colombia
37	Comoro Islands
38	Congo
39	Costa Rica
40	Cote d'Ivoire
41	Cuba
42	Cyprus
43	Czechoslovakia
44	Denmark
45	Djibouti
46	Dominica
47	Dominican Republic
48	Ecuador
49	Egypt
50	El Salvador
51	Equatorial Guinea
52	Ethiopia
53	Fiji
54	Finland
55	France
56	Gabon
57	Germany
58	Ghana
59	Greece
60	Grenada
61	Guatemala
62	Guinea
63	Guinea-Bissua
64	Guyana
65	Haiti

30.1.2 Country ID (cont).

Field Value	Country
66	Honduras
67	Hungary
68	Iceland
69	India
70	Indonesia
71	Iran
72	Iraq
73	Ireland
74	Israel
75	Italy
76	Jamaica
77	Japan
78	Jordan
79	Kenya
80	Kiribati
81	Korea
82	Democratic People's Republic of Korea
83	Republic of Korea
84	Kuwait
85	Laos
86	Lebanon
87	Lesotho
88	Liberia
89	Libya
90	Liechtenstein
91	Luxembourg
92	Madagascar
93	Malawi
94	Malaysia
95	Maldives
96	Mali
97	Malta
98	Mauritania
99	Mauritius
100	Mexico
101	Monaco
102	Mongolia
103	Morocco
104	Mosambique
105	Namibia (South West Africa)
106	Nauru
107	Nepal

APPENDIX F

30.1.2 Country ID (cont).

Field Value	Country
108	Netherlands
109	Aruba
110	Netherlands Antilles
111	New Zealand
112	Nicaragua
113	Niger
114	Nigeria
115	Norway
116	Oman
117	Pakistan
118	Panama
119	Papua New Guinea
120	Paraguay
121	Peru
122	Philippines
123	Poland
124	Portugal
125	Qatar
126	Romania
127	Rwanda
128	St. Christopher and Nevis
129	St. Lucia
130	St. Vincent and the Grenadines
131	San Marino
132	Sao Tome and Principe
133	Saudi Arabia
134	Senegambia
135	Senegal
136	Gambia
137	Seychelles
138	Sierra Leone
139	Singapore
140	Solomon Islands
141	Somalia
142	South Africa
143	Bophuthatswana
144	Ciskei
145	Transkei
146	Venda
147	Spain
148	Sri Lanka
149	Sudan

30.1.2 Country ID (cont).

Field Value	Country
150	Suriname
151	Swaziland
152	Sweden
153	Switzerland
154	Syria
155	Tanzania
156	Thailand
157	Togo
158	Tonga
159	Trinidad and Tobago
160	Tunisia
161	Turkey
162	Tuvalu
163	Uganda
164	Union of Soviet Socialist Republics
165	United Arab Emirates
166	United Kingdom
167	Northern Ireland
168	United States
169	Uruguay
170	Vanuatu
171	Vatican City State
172	Venezuela
173	Vietnam
174	Western Samoa
175	Yemen Arab Republic
176	Yemen, People's Democratic Republic of
177	Yugoslavia
178	Zaire
179	Zambia
180	Zimbabwe
181	Palestine Liberation Organization
200	Neutral

30.1.3 Detonation result. The result of a detonation shall be defined by one of the following:

Field Value	Detonation Result
0	Detonation
1	Impact
2	No impact or detonation

30.1.4 Entity marking. Entity markings may have the following character sets:

Field Value	Character set
0	Unused
1	ASCII

30.1.5 Organizational unit values. See Appendix E.

30.1.6 PDU kinds. The kind of PDU is identified as follows:

30.1.6.1 Reserved values. Values less than 128 are reserved for official use.

30.1.6.2 Defined values. Defined values are as follows:

Field Value	PDU Kind
0	Other
1	Entity State PDU
2	Fire PDU
3	Detonation PDU
4	Collision PDU
5	Radar PDU
6	Service Request PDU
7	Resupply Offer PDU
8	Resupply Received PDU
9	Resupply Cancel PDU
10	Repair Complete PDU
11	Repair Response PDU
12	Update Threshold Request PDU
13	Update Threshold Response PDU
14	Activate Request PDU
15	Activate Response PDU
16	Deactivate Request PDU
17	Deactivate Response PDU
18	Emitter PDU

30.1.6.3 Temporary. Values greater than 128 may be used for experimental purposes.

30.1.7 Radar mode. Modes defined for radar are as follows:

Field Value	Radar Modes
0	Off
1	Search
2	Doppler HPRF
3	Doppler MPRF
4	Doppler LPRF
5	Monopulse
6	Acquisition
7	Tracking
8	Track While Scan
9	Terrain Follow
10	Data Link

30.1.8 Radar system category. Radar system category is specified by bits 28-31 of a 32-bit unsigned integer defined as the Radar System Record. Values for this four bit field are assigned as follows:

Field Value	Category
0	Reserved (unused)
1	Air-Based Fire Control
2	Air-Based Search
3	Ground-Based Fire Control
4	Ground-Based Search
5	Sea-Based Fire Control
6	Sea-Based Search

30.1.9 Radar system subcategory. Radar system subcategory is specified by bits 16-23 of a 32-bit unsigned integer defined as the Radar System Record. Values for this field are to be determined.

30.1.10 Radar system ID. Radar system ID is specified by bits 0-15 of a 32-bit unsigned integer defined as the Radar System Record. Values for this 16-bit field are assigned as follows:

Field Value	Category
0	Reserved (unused)
1	AN/APG-66 (F-16A)
2	AN/APG-68 (F-16C)
3	AN/APG-63 (F-15)
4	AN/APG-65 (F/A-18)
5	AN/APG-70 (F-15E)
6	JayBird (Mig-21, Su-24, Su-20/22)
7	(Mig-31)
8	(Mig-29)
9	(Mig-27)
10	(Su-27)
11	AN/APY-2 (E-3A)
12	SUAWACS (IL-76 Mainstay)
13	FoxFire (Mig-25)
14	HighLark (Mig-23S)
15	AN/APS-125 (E2C Hawkeye)
16	LN-66 HP (SH-2F Seasprite)
17	AN/APS-116 (S-3A/B Viking)
18	AN/APS-115 (P3B Orion)
19	AN/SPQ-9
20	AN/SPQ-9A
21	AN/SPG-60
22	AN/SPS-49
23	AN/SPS-55
24	AN/SPS-67
25	AN/SPS-10
26	AN/SPY-1

30.1.11 Repair types. See Appendix C.

30.1.12 Result of Repair. Results for Repair Response PDU are defined as follows:

Field Value	Result of Repair
0	other
1	repair ended
2	invalid repair
3	repair interrupted
4	service canceled by the supplier

30.1.13 Service type. Service types are as follows:

Field Value	Service Type
0	Other
1	Resupply
2	Repair

30.1.14 Update change result. Results for an Update Threshold Response PDU are defined as follows:

Field Value	Update Change Result
0	Unused
1	Change has been accepted and will be implemented
2	Tighter threshold already implemented
3	Change will not be implemented

30.2 Enumeration values and bit-encoding for interim and proposed PDUs.

30.2.1 Activate reason. Reasons for activation of an entity in the Activate Request PDU are defined as follows:

Field Value	Activate Reason
0	Other
1	Exercise Start
2	Exercise Restart
3	Exercise Entry
4	Entity Reconstitution

30.2.2 Activate result. Results of activation in the Activate Response PDU are defined as follows:

Field Value	Activate Result
0	Other
1	Activate Request Accepted
2	Invalid Activate Parameter
3	Unexpected Activate Parameter

30.2.3 Deactivate reason. Reasons for deactivation of an entity in the Deactivate Request PDU are defined as follows:

Field Value	Deactivate Reason
0	Other
1	Exercise End
2	Entity Withdrawn
3	Entity Destroyed

30.2.4 Deactivate reason. Results of deactivation of an entity in the Deactivate Response PDU are defined as follows:

Field Value	Deactivate Reason
0	Other
1	Deactivate Request Accepted
2	Invalid Deactivate Parameter
3	Unexpected Deactivate Reason
4	Vehicle Not Active

30.2.5 Emitter data base number. Values defined for Emitter Database Number are:

Field Value	Data Base
0	Other
1	UTSS

30.2.6 Emitter class. Table II defines the Emitter Classes in the Emitter PDU.

APPENDIX F

Table II. Electromagnetic Spectrum: Emitter Classes.

Class Number (Field Value)	Region	Wavelength
0	Other	----------
1	Sound	----------
2	Infrasonic	Above 100km
3	Very Low Frequencies	100km - 10km
4	Low Frequencies	10km - 1km
5	Medium Frequencies	1000m - 100m
6	High Frequencies	100m - 10m
7	Very High Frequencies	10m - 1m
8	Ultra High Frequencies	1m - 10cm
9	Super High Frequencies	10cm - 1cm
10	Extremely High Frequencies	1cm - 10^{-1}cm
11	Infrared	10^{-1}cm - 10^{-4}cm
12	Visible	10^{-4}cm - 10^{-5}cm
13	Ultraviolet	10^{-5}cm - 10^{-6}cm
14	X Rays	10^{-6}cm - 10^{-8}cm
15	Gamma Rays	10^{-8}cm - 10^{-11}cm
16	Cosmic Rays	Below - 10^{-11}cm

APPENDIX G

Articulated Parts

10. SCOPE.

10.1 Scope. This Appendix defines the different Articulation Parameter Types specified by this standard. This Appendix is a mandatory part of the standard. The information contained herein is intended for compliance.

20. APPLICABLE DOCUMENTS. This section is not applicable to this appendix.

30. DETAILS.

30.1 Articulation parameters: Articulation parameters values shall be represented in the following manner:

30.1.1 Elevation. Elevation shall be measured as an angle in units of BAMs. The value zero shall be used to represent an angle horizontal relative to an entity's x-y plane. Stationary entities shall be measured relative to the horizon. Positive direction shall be up.

30.1.2 Azimuth. Azimuth shall be measured as an angle in units of BAMs. The value zero shall be used to represent direction toward the front of an entity and positive angles are measured clockwise relative to the entity's x-axis. For anything that does not move or does not have an obvious front (for example, stationary surface-to-air missile launcher) then the value zero shall represent North.

30.1.3 Extension. Extension shall be measured as the length of the part in one dimension past the entity coordinates (as if the entity position was at the origin) measured in meters. The value zero shall represent fully retracted and 2^{32} shall represent fully extended. Intermediate positions are represented as a value between zero and 2^{32}.

30.1.4 Position. Position shall specify the location of articulated parts with a particular path to which its movement is limited. The value for this type of variable shall be measured in a manner similar to BAMs. The value zero shall represent fully retracted and 2^{32} shall represent fully extended. Intermediate positions are represented as a value between zero and 2^{32}.

30.1.5 Station. Station shall specify the station to which an attached part (fixed articulated part) is attached. Stations shall be numbered sequentially beginning with one and incrementing by one. The order of numbering shall be from top to bottom, then back to front, then left to right. The only exception shall be aircraft wing stations. The fuselage stations, left wing stations, and right wing stations shall be separated into different categories. Wing stations shall be numbered from inboard to outboard. The value field shall contain the Entity Type Record representing the type of store located at the specified station.

30.2 Guidelines for choosing parameters. Although it is possible for any articulated part to be represented by any parameter, Table G-1 provides a guideline for determining which type of parameter should be used for certain types of parameters.

Table G-1. Guidelines for choosing articulation parameters.

PART	RECOMMENDED PARAMETER
horizontal control surfaces	elevation
vertical control surfaces	azimuth
extendable items	extension
fixed position items	station
fixed path items	position
turrets	azimuth
guns	elevation
moveable missile launcher	azimuth and elevation

Rate parameters shall be used when dead reckoning of an articulated part is expected. Also, in using these rules the neutral position for aircraft rudders shall be equal to $2^{32} / 2$. It shall not be zero.

30.3 Articulation Parameters. This table consists of two sections. The first section represents the type of parameter sent. There are 20 possible values. Of these values, 16 are presently defined. The second section of the table represents the specific part parameter whose value shall be measured in terms of the type of parameter specified in the first section of the table. The first 1000 entries are reserved for identifying stations. The rest identify articulation parameters. Each entry in the second section of the table is separated by 20 values, corresponding to the 20 possible parameter types represented by the first section.

30.3.1 Parameter types. The following digits define the parameter sent.

XX01 =	position
XX02 =	position rate
XX03 =	extension
XX04 =	extension rate
XX05 =	X
XX06 =	X-rate
XX07 =	Y
XX08 =	Y-rate
XX09 =	Z
XX10 =	Z-rate
XX11 =	azimuth
XX12 =	azimuth-rate
XX13 =	elevation
XX14 =	elevation-rate
XX15 =	rotation
XX16 =	rotation rate

XX17 - XX19 Not defined - They may be used for acceleration if necessary

30.3.2 Specific attached and articulated parts.

30.3.2.1 Attached part stations.

INDEX	DEFINITION
	Air, Ground, Surface or Sub-surface Entities
0	NOTHING, EMPTY
0001 - 0019	primary launcher store stations[3]
0020 - 0039	secondary launcher store stations
0040 - 0059	weapon mounts[4]
0060 - 0079	sensor mounts[5]
0080	bulldozer blade
0081	mine clearing
0082	rollers
0083	swimming gear
0084 - 0099	other attached parts

30.3.2.2 Articulating parts.

INDEX	DEFINITION
	Air, Ground, Surface or Sub-surface Entities
	Air Entities:
0101 - 0119	left wing stations
0121 - 0139	right wing stations
	Ground Entities:
0141 - 0159	cargo stations
0161 - 0179	armor mounts
0181 - 0999	UNDEFINED

---

[3]For aircraft this refers to only the store stations on the fuselage. Wing stations are defined separately.

[4]Jeep machine gun mounts or ship guns are not considered as articulated parts.

[5]A FLIR (Forward Looking InfraRed) system on an F-16 for example. On the actual aircraft they are not attached to station hardpoints.

APPENDIX G

INDEX	DEFINITION	
1000 - 1019	rudder	
1020 - 1039	left flap	
1040 - 1059	right flap	
1060 - 1079	left aileron	
1080 - 1099	right aileron	
1100 - 1999	other Aircraft Control Surfaces defined as needed	
2000 - 2019	periscope	
2020 - 2039	generic antenna	
2040 - 2059	snorkel	
2060 - 2999	other extendable parts defined as needed	
3000 - 3019	landing gear	
3020 - 3039	tail hook	
3040 - 3059	speed brake	
3060 - 3079	left weapon bay door	
3080 - 3099	right weapon bay doors	
3100 - 3119	tank or APC hatch	
3120 - 3999	other fixed position parts defined as needed	
4000 - 4019	Primary turret number	1
4020 - 4039	Primary turret number	2
4040 - 4059	Primary turret number	3
4060 - 4079	Primary turret number	4
4080 - 4099	Primary turret number	5
4100 - 4119	Primary turret number	6
4120 - 4139	Primary turret number	7
4140 - 4159	Primary turret number	8
4160 - 4179	Primary turret number	9
4180 - 4199	Primary turret number	10
4200 - 4219	Primary gun number	1
4220 - 4239	Primary gun number	2
4240 - 4259	Primary gun number	3
4260 - 4279	Primary gun number	4
4280 - 4299	Primary gun number	5
4300 - 4319	Primary gun number	6
4320 - 4339	Primary gun number	7
4340 - 4359	Primary gun number	8
4360 - 4379	Primary gun number	9
4380 - 4399	Primary gun number	10

APPENDIX G

INDEX	DEFINITION	
4400 - 4419	Primary launcher	1
4420 - 4439	Primary launcher	2
4440 - 4459	Primary launcher	3
4460 - 4479	Primary launcher	4
4480 - 4499	Primary launcher	5
4500 - 4519	Primary launcher	6
4520 - 4539	Primary launcher	7
4540 - 4559	Primary launcher	8
4560 - 4579	Primary launcher	9
4580 - 4599	Primary launcher	10
4600 - 4619	Primary defense systems	1[6]
4620 - 4639	Primary defense systems	2
4640 - 4659	Primary defense systems	3
4660 - 4679	Primary defense systems	4
4680 - 4699	Primary defense systems	5
4700 - 4719	Primary defense systems	6
4720 - 4739	Primary defense systems	7
4740 - 4759	Primary defense systems	8
4760 - 4779	Primary defense systems	9
4780 - 4799	Primary defense systems	10
4800 - 4819	Primary radar	1[7]
4820 - 4839	Primary radar	2
4840 - 4859	Primary radar	3
4860 - 4879	Primary radar	4
4880 - 4899	Primary radar	5
4900 - 4919	Primary radar	6
4920 - 4939	Primary radar	7
4940 - 4959	Primary radar	8
4960 - 4979	Primary radar	9
4980 - 4999	Primary radar	10

[6]Point defense systems like phalanx guns, or visible chaff, flare, or smoke dispensers.

[7]Any radar dish or movable antenna or sensor.

APPENDIX G

INDEX	DEFINITION
5000 - 5019	Secondary turret number 1
5020 - 5039	Secondary turret number 2
5040 - 5059	Secondary turret number 3
5060 - 5079	Secondary turret number 4
5080 - 5099	Secondary turret number 5
5100 - 5119	Secondary turret number 6
5120 - 5139	Secondary turret number 7
5140 - 5159	Secondary turret number 8
5160 - 5179	Secondary turret number 9
5180 - 5199	Secondary turret number 10
5200 - 5219	Secondary gun number 1
5220 - 5239	Secondary gun number 2
5240 - 5259	Secondary gun number 3
5260 - 5279	Secondary gun number 4
5280 - 5299	Secondary gun number 5
5300 - 5319	Secondary gun number 6
5320 - 5339	Secondary gun number 7
5340 - 5359	Secondary gun number 8
5360 - 5379	Secondary gun number 9
5380 - 5399	Secondary gun number 10
5400 - 5419	Secondary launcher 1
5420 - 5439	Secondary launcher 2
5440 - 5459	Secondary launcher 3
5460 - 5479	Secondary launcher 4
5480 - 5499	Secondary launcher 5
5500 - 5519	Secondary launcher 6
5520 - 5539	Secondary launcher 7
5540 - 5559	Secondary launcher 8
5560 - 5579	Secondary launcher 9
5580 - 5599	Secondary launcher 10
5600 - 5619	Secondary defense systems 1
5620 - 5639	Secondary defense systems 2
5640 - 5659	Secondary defense systems 3
5660 - 5679	Secondary defense systems 4
5680 - 5699	Secondary defense systems 5
5700 - 5719	Secondary defense systems 6
5720 - 5739	Secondary defense systems 7
5740 - 5759	Secondary defense systems 8
5760 - 5779	Secondary defense systems 9
5780 - 5799	Secondary defense systems 10

APPENDIX G

INDEX	DEFINITION	
5800 - 5819	Secondary radar	1
5820 - 5839	Secondary radar	2
5840 - 5859	Secondary radar	3
5860 - 5879	Secondary radar	4
5880 - 5899	Secondary radar	5
5900 - 5919	Secondary radar	6
5920 - 5939	Secondary radar	7
5940 - 5959	Secondary radar	8
5960 - 5979	Secondary radar	9
5980 - 5999	Secondary radar	10

Figure G-1 illustrates the Articulated Parts Record.

APPENDIX G

FIELD SIZE (bits)	ARTICULATED PARTS RECORD	
FIRST ARTICULATED PART # bits: $96m_1 + 64$	PADDING - 16 bits unused	
	Change indicator - 16 - bit unsigned integer	
	Part ID - attached to - 16 - bit unisgned integer	
	# of parameters (m_1) - 16 - bit unisgned integer	
	$96m_1$ bits	Parameter #1 type - 32 - bit unsigned integer
		Parameter #1 value - 64 - bit floating pt/Entity type
		Parameter #2 type
		Parameter #2 value
		:
		:
SECOND ARTICULATED PART # bits: $96m_2 + 64$	PADDING - 16 bits unused	
	Change indicator - 16 - bit unsigned integer	
	Part ID attached to - 16 - bit unsigned integer	
	# of parameters (m_2) - 16 - bit unsigned integer	
	$96m_2$ bits	Parameter #1 type - 32 - bit unsigned integer
		Parameter #1 value - 64 - bit floating pt/entity type
		:
		:
:		
nth ARTICULATED PART # bits: $96m_n + 64$	PADDING - 16 bits unused	
	Change indicator - 16 - bit unsigned integer	
	Part ID attached to - 16 - bit unsigned integer	
	# of parameters (m_n) - 16 - bit unsigned integer	
	$96m_n$ bits	Parameter #1 type
		Parameter #1 value
		Parameter #2 type
		Parameter #2 value

Total bits for Articulated Parts Record:

$n \times (96m_i + 64)$

n = # articulated parts

i = 1 to n

m = number of parameters for articulated part i

FIGURE G-1. Articulated Part Record.

30.4 Example #1. To represent the articulation of a tank gun, azimuth of the turret is required as well as the elevation of the gun. This represents two articulated parts. The tank itself is considered to have Part ID = 0. The turret is attached to the tank. It has Part ID = 1 but is attached to Part ID = 0. We wish to dead reckon the position of the turret so we will require two articulation parameters: the azimuth and the azimuth rate. The gun is attached to the turret and has a Part ID = 2 but is attached to Part ID = 1. It has one articulation parameter, namely the elevation of the gun.

30.4.1 Parameter types. For the turret, parameter #1 is the azimuth. The value to represent this type is 4010 where 4000 represents primary turret and 10 represents azimuth. Parameter #2 is the azimuth rate in BAMs. The value to represent this type is 4011 where 4000 represents primary turret and 11 represents the azimuth rate in BAMs per second. For the gun, parameter #1 is elevation. The value to represent this type is 4212 where 4200 represents the primary gun and 12 represents elevation.

The Articulated Part Record would be represented in the following manner:

```
Number of Articulated Parts:                2
================================================================
   Change indicator:      all 1's - representing a change
   Part ID - attached to:                    0 (tank)
   # of parameters for part #1:              2
      Parameter #1 type:                   4010
      Parameter value:              (an angle in BAMs)
      Parameter #2 type:                   4011
      Parameter value:              (rate in BAMS/sec)
----------------------------------------------------------------
   Change indicator:      all 1's - representing a change
   Part ID - attached to:                    1 (turret)
   # of parameters for part #2:              1
      Parameter #1 type:                   4212
      Parameter value:              (an angle in BAMs)
```

30.5 Example #2. To represent the presence of an air-to-surface missile on an aircraft it is necessary to specify the station to which the missile is attached and the type of missile. Since the specific details (such as country of origin, type of warhead, etc) is not crucial for depiction of the missile, the type of missile shall be specified by the Munition Type values specified in Appendix B. For the presence of a missile on the first left wing station the parameter type would be 0101 for the first left wing station. The parameter value would indicate the type of munition attached to that station. The Articulated Part Record would be represented in the following manner:

```
Number of Attached Parts:                   1
================================================================
   Change indicator:     all 0's - representing no change
   Part ID - attached to:                    0 (aircraft)
   # of parameters for part #1:              1
      Parameter #1 type:                  0101
      Parameter value:          (type of missile from munition
                                type)
```

If the missile was fired, the next ES PDU would have an Articulated Part Record with the following information on the launched missile:

Number of Attached Parts:	1
Change indicator:	all 1's - representing change
Part ID - attached to:	0 (aircraft)
# or parameters for part #1:	1
Parameter #1 type:	0101
Parameter value:	(type of missile from munition type)

The change indicator specifies that the munition that was present is no longer present. The fact that the missile is no longer present is communicated with each ES PDU that is issued for 5 minutes after the missile is launched. After that time, its absence no longer needs to be communicated and all information concerning that particular attached part shall cease to be communicated.

30.6 Articulated Part movement and update threshold values. It should be noted that angular movement of an articulated part may result in a violation of linear thresholds. Absolute linear error at each link along a chain of articulated parts must be considered in calculating discrepancies.

APPENDIX H1

Entity Types

10. SCOPE.

10.1 Scope. This Appendix specifies the interpretation of the 64-bit enumeration that represents the Entity Type. Entity Types are arranged in a hierarchical order such that higher fidelity simulations may depict detailed representations of the entity (such as an F-16B) while lower fidelity simulations may depict the same entity in a more generic manner (such as an F-16 or a Fighter Aircraft). This Appendix is a mandatory part of the standard. The information contained herein is intended for compliance.

20. APPLICABLE DOCUMENTS. This section is not applicable to this appendix.

30. DETAILS.

30.1 Entity Kind. The first field specifies the entity's kind. This field is an 8-bit enumeration. The definitions of the remaining fields are based on this classification. The defined values for this field are as follows:

Field Value	Kind
0	Other
1	Platform
2	Munition
3	Life form
4	Environmental
5	Cultural Feature

The following is a detailed discussion concerning the remaining fields for each of the classifications described above.

30.1.1 Platform Kind. The Platform entity kind refers to vehicles such as ships, tanks, aircraft and submarines. Fields used to uniquely describe platforms are detailed in the paragraphs that follow.

30.1.1.1 Domain. The second field defines a platform's Domain. This field is an 8-bit enumeration. The defined values for this field are as follows:

Field Value	Domain
0	Other
1	Land
2	Air
3	Surface
4	Subsurface
5	Space

30.1.1.2 Country. The third field represents the country to which a particular platform's design is attributed. This field is a 16-bit enumeration. The defined values for this field are contained in Appendix F.

30.1.1.3 Specific Platform Definition. For an entity whose type is platform, the remaining fields uniquely define all platforms under the headings of Category, Designation, Model and Series. The defined values for these headings are contained in Appendix H2 Section 30.1.

30.1.2 Munition Kind. The Munition entity kind refers to types of munitions such as missiles, ballistic rounds, bullets and torpedoes. Fields used to uniquely describe munitions are detailed in the paragraphs that follow.

30.1.2.1 Domain. The second field defines the domain of the target entity for which the munition is intended. This field is an 8-bit enumeration. The defined values for this field are as follows:

Field Value	Domain
0	Other
1	Anti-Air
2	Anti-Armor
3	Anti-Guided Munition
4	Anti-Radar
5	Anti-Satellite
6	Anti-Ship
7	Anti-Submarine
8	Battlefield Support
9	Strategic
10	Miscellaneous

30.1.2.2 Country. The third field represents the country to which a particular munition's design is attributed. This field

is a 16-bit enumeration. The defined values for this field are contained in Appendix F.

30.1.2.3 Specific Munition Definition. For an entity whose kind is munition, the remaining fields uniquely define all munitions under the headings of Category, Designation, Model and Series. The defined values for these headings are contained in Appendix H2.

30.1.2.4 Fuze and Warhead. Munitions are further defined in terms of their Fuze and Warhead. The values presently defined for these fields are contained in Appendix B.

30.1.3 Life Form Kinds. The Life Form entity kind refers to Dismounted Infantry, Scouts, SEALS or other pedestrian soldiers. They are treated in a manner similar to platforms since they can move and can launch munitions. Fields used to uniquely define life forms are detailed in the paragraphs that follow.

30.1.3.1 Domain. The second field defines a life form's Domain. This field is an 8-bit enumeration. The defined values for this field are the same as for platforms (see 30.1.1.1).

30.1.3.2 Country. The third field defines a life form's country of origin. This field is a 16-bit enumeration. The defined values for this field are found in Appendix F.

30.1.3.3 Specific Life Form Definition. For an entity whose kind is life form, the remaining fields will uniquely define all life forms under the headings of Category, Designation, and Number of Individuals[8]. The last 8-bit field is unused at this time and shall contain the value zero. The defined values for these fields are as follows:

[8]The field specifying the Number of Individuals allows representation of one or more life forms using one PDU.

APPENDIX H1

Kind Dom Count Cat Desig Num Unused

4 Lifeform

0 Other
1 Land

See Appendix F

0 Other
1 Dismounted Infantry (with)

0 Other
See Note 1*

Number of indiv. Life Forms**

Unused

2 Air

See Appendix F

0 Other
1 Parachutist (with)

0 Other
See Note 1*

Number of indiv. Life Forms**

Unused

3 Surface

See Appendix F

0 Other
1 Swimmer (with)

0 Other
See Note 1*

Number of indiv. Life Forms**

Unused

APPENDIX H1

Kind Dom Count Cat Desig Num Unused

4 Lifeform

3 Surface (cont)

See Appendix F

2 Occupied life raft

0 Other
1 TBD

Number of indiv. Life Forms**

Unused

4 Subsurface

See Appendix F

0 Other
1 Diver (with)

0 Other
See Note 1*

Number of indiv. Life Forms**

Unused

5 Space

See Appendix F

0 Other
1 Astronaut (with)

0 Other
See Note 1*

Number of indiv. Life Forms**

Unused

* Note 1. Weapon designations will be specified using the Life Forms weapons designation table in Appendix H2.

** Note 2. Number of Life Forms:

1-100	Number of individual Life Forms
101	Fire Team
102	Squad
103	Platoon
104	Company
105	Battalion
106	Regiment
107	Division
108	Corps
109	Army

30.1.4 Environmental Kinds. The Environmental entity kind refers to clouds, smoke and biologics. Fields used to uniquely define environmentals are detailed in the paragraphs that follow.

30.1.4.1 Domain. The second field defines the domain in which the environmental exists. This field is an 8-bit enumeration. The defined values for this field are the same as for platforms (see 30.1.1.1).

30.1.4.2 Country. This field is unused for environmentals and shall contain the value zero.

30.1.4.3 Specific Environmental Definition. For an entity whose kind is environmental, the remaining fields will uniquely describe all environmentals under the headings of Category, and Size. The last two fields are unused at this time and shall contain the value zero. The defined values for these headings are as follows:

Kind Dom Count Cat Size Unused Unused

5 Environmental

0 Other
1 Land

Unused

0 Other
1 Smoke
2 Fog
3 Dust Cloud

See Note 1*

2 Air

Unused

0 Other
1 Smoke
2 Fog
3 Flock of Birds
4 Cloud
5 Cloud With Rain Falling
6 Cloud With Snow Falling

See Note 1*

3 Surface

Unused

0 Other
1 TBD

See Note 1*

4 Subsurface

Unused

1 Thermocline
2 Knot
3 School of Fish
4 Whale
5 School of Shrimp

Kind Dom Count Cat Size Unused Unused

5 Environmental (cont)

5 Space

Unused

0 Other
1 TBD

See Note 1*

---

* Note 1 Size

0	Other
20	Very Small
40	Small
60	Medium
80	Large
100	Very Large

30.1.5 Cultural Feature Kind. The Cultural Feature entity kind refers to engineering, weapons, and natural effects. These include craters, earth mounds, and vehicle tracks. Fields used to uniquely identify cultural features are detailed in the paragraphs that follow.

30.1.5.1 Domain. The second field defines the domain in which the cultural feature exists. This field is an 8-bit enumeration. The defined values for this field are the same as for platforms (see 30.1.1.1).

30.1.5.2 Country. This field is unused for cultural features and shall contain the value zero.

30.1.5.3 Specific Cultural Feature Definition. For an entity whose kind is cultural feature, the remaining fields will uniquely define all cultural features under the headings of Category and Size. The last two fields are unused at this time. The defined values for these headings are as follows:

APPENDIX H1

Kind Dom Count Cat Size Unused Unused

5 Cultural Feature

0 Other
1 Land

Unused

0 Other
1 Bridge-concrete-two lane

See Note 1*

2 Bridge-concrete-four lane
3 Bridge-truss-two lane
4 Bridge-truss-four lane
5 Bridge-suspension-two lane
6 Bridge-suspension-four lane
7 Building-one story
8 Building-two story
9 Building-three story
10 Building-four story
11 Tracks-tank
15 Crater
16 Ditch

2 Air

Unused

0 Other
1 TBD

3 Surface

Unused

0 Other
1 Oil Derrick
2 Buoy
3 Pier

Kind Dom Count Cat Size Unused Unused

5 Cultural Feature

4 Subsurface

Unused

0 Other
1 Oil Derrick
2 Buoy
3 Sunken Ship

5 Space

Unused

0 Other
1 TBD

---

*Note 1:
Entity Size

0	Other
20	Very small
40	Small
60	Medium
80	Large
100	Very large

DRAFT

APPENDIX H2

APPENDIX H2 IS IN PRESS

STANDARDIZATION DOCUMENT IMPROVEMENT PROPOSAL

(See Instructions - Reverse Side)

1. DOCUMENT NUMBER	2. DOCUMENT TITLE
MIL STD (DRAFT)	Protocol Data Units for Entity Information and Entity Interaction in a Distributed Interactive Simulation

3a. NAME OF SUBMITTING ORGANIZATION	4. TYPE OF ORGANIZATION *(Mark one)*
	☐ VENDOR
3b. ADDRESS *(Street, City, State, ZIP Code)*	☐ USER
	☐ MANUFACTURER
	☐ OTHER *(Specifiy):*

5. PROBLEM AREAS

a. Paragraph Number and Wording:

b. Recommended Wording:

c. Reason/Rationale for Recommendation:

6. REMARKS

7a. NAME OF SUBMITTER *(Last, First, MI) - Optional*	b. WORK TELEPHONE NUMBER *(Include Area Code) - Optional*
c. MAILING ADDRESS *(Street, City, State, ZIP Code) - Optional*	8. DATE OF SUBMISSION *(YYMMDD)*

0147-1463

INSTRUCTIONS: In a continuing effort to make our standardization documents better, IST has provided this form for use in submitting comments and suggestions for improvements. All users of military standardization documents are invited to provide suggestions. This form may be detatched, folded along the lines indicated, taped along the loose edge (DO NOT STAPLE), and mailed. Postage is required. In block 5, be as specific as possible about particular problem areas such as wording which requires interpretation, or was too rigid, restrictive, loose, ambiguous, or incompatible, and give proposed wording changes which would alleviate the problems. Enter in block 6 any remarks not related to a specific paragraph of the document. If additional space is required, additional pages may be added and the forms mailed in an appropriately sized envelope to the address specified below.

NOTE: This form may not be used to request copies of documents, nor to request waivers, deviations, or clarification of specification requirements on current contracts. Comments submitted on this form do not constitute or imply authorization to waive any portion of the referenced document(s) or to amend contractual requirements.

(Fold along this line)

(Fold along this line)

POST OFFICE WILL NOT DELIVER MAIL WITHOUT PROPER POSTAGE

INSTITUTE FOR SIMULATION & TRAINING
12424 RESEARCH PKWY. SUITE 300
ORLANDO, FLORIDA 32826
ATTN. Dr. Bruce McDonald