

1-1-1992

## Investigation Of OSI Protocols For Distributed Interactive Simulation: Final Report, A Transition Plan

University of Central Florida Institute for Simulation and Training

Find similar works at: <https://stars.library.ucf.edu/istlibrary>  
University of Central Florida Libraries <http://library.ucf.edu>

This Research Report is brought to you for free and open access by the Digital Collections at STARS. It has been accepted for inclusion in Institute for Simulation and Training by an authorized administrator of STARS. For more information, please contact [STARS@ucf.edu](mailto:STARS@ucf.edu).

---

### Recommended Citation

University of Central Florida Institute for Simulation and Training, "Investigation Of OSI Protocols For Distributed Interactive Simulation: Final Report, A Transition Plan" (1992). *Institute for Simulation and Training*. 129.

<https://stars.library.ucf.edu/istlibrary/129>

Contract Number N61339-91-C-0103  
STRICOM

August 20, 1992

# Investigation of OSI Protocols for Distributed Interactive Simulation

## Final Report: A Transition Plan

  
**IST**

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

University of Central Florida  
Division of Sponsored Research

B 160

IST-TR-92-29

Contract Number N61339-91-C-0103  
STRICOM

August 20, 1992

# **Investigation of OSI Protocols for Distributed Interactive Simulation**

## **Final Report: A Transition Plan**

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

**University of Central Florida  
Division of Sponsored Research**

**INVESTIGATION OF OSI PROTOCOLS  
FOR DISTRIBUTED INTERACTIVE SIMULATION  
FINAL REPORT:**

**A TRANSITION PLAN**

PREPARED FOR:

U.S. ARMY SIMULATION TRAINING AND INSTRUMENTATION COMMAND  
12350 Research Parkway  
Orlando, Florida 32826-3276

INVESTIGATION OF OSI PROTOCOLS FOR  
DISTRIBUTED INTERACTIVE SIMULATION

CONTRACT N61339-91-C-0103  
CDRL A003

August 20, 1992

Institute for Simulation and Training  
University of Central Florida  
12424 Research Parkway  
Orlando, Florida 32826



**INVESTIGATION OF OSI PROTOCOLS  
FOR DISTRIBUTED INTERACTIVE SIMULATION  
FINAL REPORT:**

**A TRANSITION PLAN**

IST Technical Report  
IST-TR-92-29

Written By: Margaret L. Loper  
Margaret L. Loper, *Principal Investigator*  
M. Bassiouni, David Shen, Huat Ng, and Gamini Bulumulle

Reviewed By: Scott H. Smith  
Scott H. Smith, *Program Manager*

Institute for Simulation and Training  
University of Central Florida  
12424 Research Parkway  
Orlando, Florida 32826

## TABLE OF CONTENTS

INTRODUCTION . . . . .	1
DISTRIBUTED INTERACTIVE SIMULATION . . . . .	3
2.1 The Applications . . . . .	3
2.1.1 Wargames . . . . .	3
2.1.1.1 Protocol Data Units . . . . .	3
2.1.2 Field Instrumentation . . . . .	3
2.1.2.1 Protocol Data Units . . . . .	4
2.1.3 Simulations and Simulators . . . . .	4
2.1.3.1 Protocol Data Units . . . . .	4
2.1.3.2 Processing Constraints . . . . .	6
2.2 The Seams . . . . .	9
SERVICE REQUIREMENTS . . . . .	12
3.1 Application Service Characterization . . . . .	12
3.1.1 Definitions . . . . .	13
3.1.2 PDU Characterization . . . . .	14
3.2 Interim and Long-Range Requirements . . . . .	17
STANDARDS AND PROTOCOLS . . . . .	18
4.1 Standards . . . . .	18
4.1.1 Internet Protocol Suite . . . . .	18
4.1.2 Open Systems Interconnection . . . . .	19
4.2 Protocols . . . . .	20
4.2.1 The Interim Choices for Multicast . . . . .	21
RESEARCH . . . . .	23
5.1 Abstract Syntax Notation One . . . . .	23
5.2 Estelle Formal Description . . . . .	25
5.3 Multipeer/Multicast . . . . .	27
TRANSITION PLAN . . . . .	30
6.1 Areas of Development . . . . .	30
6.1.1 Upper Layers . . . . .	30
6.1.2 Lower Layers . . . . .	32
6.2 Three Phases . . . . .	35
6.2.1 Phase 0: Internet Architecture . . . . .	36
6.2.2 Phase 1: OSI Architecture . . . . .	37
6.2.3 Phase 2: GOSIP Architecture . . . . .	39
6.3 Optimizing the Transition . . . . .	40
6.4 A Timeline . . . . .	41

7. CONCLUSIONS .....	44
ACKNOWLEDGEMENT .....	45
REFERENCES .....	46

## LIST OF TABLES AND FIGURES

Table 1	DIS Functional Requirements: Required (R) and Non-Required (NR) PDUs . .	5
Table 2	Sample DIS Traffic Analysis . . . . .	8
Table 3	DIS Communication Service Requirements . . . . .	12
Table 4	Characterization of Required DIS PDUs . . . . .	14
Table 5	Characterization of Non-Required DIS PDUs . . . . .	15
Table 6	Characterization of DIS Functional Requirements . . . . .	15
Table 7	DIS Application Service Model . . . . .	16
Table 8	Summary of DIS Service Requirements and Protocols . . . . .	21
Table 9	ASN.1 Overhead (in milliseconds) . . . . .	24
Table 10	Phase 0 Service Characterization . . . . .	36
Table 11	Phase 1 Service Characterization . . . . .	38
Table 12	Phase 2 Service Characterization . . . . .	40
Figure 1.	Processing Delays in Simulators . . . . .	6
Figure 2.	Integrating DIS Applications . . . . .	11
Figure 3.	Decomposing the Network Layer into Layer Functional Modules . . . . .	33
Figure 4.	Three Phase Communication Architecture . . . . .	35
Figure 5.	Timeline for DIS Applications and Standards Evolution . . . . .	43



## 1. INTRODUCTION

This report documents the research performed on contract N61339-91-C-0103, **Investigation of Open System Interconnection Protocols for Distributed Interactive Simulation**, for the period June 1991 - June 1992. The goal of the contract was to assess the impact of using Open System Interconnection (OSI) Protocols in the Distributed Interactive Simulation (DIS) environment. This was accomplished through three tasks: (1) Examine the relationship between DIS and OSI, (2) Establish an OSI research and development testbed, and (3) Participate in DIS and American National Standards Institute (ANSI)/International Organization for Standardization (ISO) standards processes.

While specific sub-tasks were developed for the OSI testbed, the remainder of the project was a requirements analysis. These analyses were aimed at general communication services, i.e. basic data transfer, and specific service requirements, i.e. multicast data transfer. This contract originally investigated only OSI protocols to meet service requirements; however, the scope was eventually expanded to include other protocol standards. All work performed under this contract has been presented to either the Communication Architecture and Security Subgroup (CASS) or the Interface/Time Mission Critical subgroup of the DIS workshops. Much of the work has been incorporated into relevant DIS standards documents, namely the Communication Architecture for DIS (CADIS) standard. The analysis to determine DIS multicast requirements has been introduced to appropriate ANSI/ISO working groups for consideration.

The purpose of this report is reflected in the following quote on the DIS Concept of Operation:

*The concept of Distributed Interactive Simulation encompasses the capability to create synthetic, virtual representations of battlefield environments by systematically connecting separate elements or sub-components of simulation which reside at distributed, multiple locations. The property of connecting separate sub-components or elements affords the capability to configure a wide range of simulated battlefield representations patterned after the task force organization of units, both friendly and opposing, including joint and combined force operations to represent a wide range of warfighting missions facing U.S. forces today and in the future. Equally important is the property of interoperability which allows different simulation environments to efficiently and consistently interchange data elements essential to representing warfighting interactions and outcomes. In effect, interoperable simulations will exchange data in a manner such that the differences in the representation of the simulated battlefield will be transparent or "seamless" as experienced by the participants interacting with their particular representation of the battlefield. This property affords the opportunity for linking heterogeneous representations, each providing a locally consistent simulated environment, through use of buffers or translators to create a seamless interconnection. With these properties, it is possible to have simulation components which meet special purpose local uses and when required can link together to form larger scale battlefield representations. [12]*

This statement identifies the basic properties of distributed simulation. In the context of this paper, these properties have a special meaning. They will be defined from a communications perspective and from these properties the characteristics of the communication subsystem can be derived. The first property, *connecting separate sub-components or elements*, refers to the Applications. Applications are the DIS components, i.e. simulators, real hardware, and wargames. These components also represent the three types of simulation, namely continuous, real-time, and discrete. The second property, *interoperability...to create a seamless interconnection*, refers to the communication protocols which allow the Applications to communicate transparently. The communication protocols range from the physical layer (e.g., Ethernet) to the application layer and are used to communicate entity state information among the Applications.

Each DIS Application requires different services to make its communication "seamless." To achieve a seamless interconnection, some customization for unique requirements will also be required. Customization results from a lack of commercially available standards and products. A goal for DIS is to minimize customization by finding common solutions and thereby broaden the market appeal for such services. For DIS, customization is considered research.

The focus of this report is how the DIS community should transition from interim solutions to a long-lived architecture which includes the required customization. Before discussing the Transition Plan, we must first lay the groundwork. This consists of the following steps:

- Define the Applications.
- For each Application<sup>1</sup>, determine the service requirements.
- For each of the service requirements, identify a communication protocol which satisfies the requirement, thus defining a suite of protocols.
- Once the protocol suite has been established, examine commercial-off-the-shelf (COTS) standards for use by the Applications.
- If COTS products do not satisfy a requirement, classify it as long-range (i.e., research).

In the sections that follow, each step above is covered in sufficient detail to provide a stepping stone for the next issue. The last section coalesces this information into a Transition Plan which defines a strategy to evolve the communication architecture from interim to long-range services.

---

<sup>1</sup> This report uses the September 1991 version of the DIS Standard [1] as the basis for the communication analyses. This version of the standard does not contain information on instrumented or wargame Applications; and so this report addresses only the service requirements for Simulators/Simulations. Any special service requirements for Wargames or Instrumentation Applications are not considered in this plan.



## **2. DISTRIBUTED INTERACTIVE SIMULATION**

### **2.1 The Applications**

There are three categories of DIS Applications: Wargames, which incorporate aggregate level entities; Instrumentation, which brings real hardware into the loop; and Simulations, which may include both manned Simulators and Computer Generated Forces (CGF). For each category, there are existing applications, which will require retro-fitting to use the DIS standard and new procurements, which refer to the DIS standard. Each Application has different requirements (e.g., bandwidth and Protocol Data Units (PDUs)). The following sections give a brief introduction to each application area, further information can be found in [6].

#### **2.1.1 Wargames**

Wargames incorporate aggregate level entities (e.g., platoon, flight, and surface action groups) into the DIS environment. Wargames are discrete or event oriented simulations and are generally used for operations analysis and warfare assessment. Wargames or force level simulations deal with groupings of entities, i.e. unit level representation, in lieu of individual entities, i.e item level representation. Wargames will be integrated into the DIS simulation environment through aggregation of item level to unit level and deaggregation of unit level to item level representations.

##### **2.1.1.1 Protocol Data Units**

The current research on Aggregate and Deaggregate PDUs [11] is an example of integrating wargames into the simulation environment.

#### **2.1.2 Field Instrumentation**

Field instrumentation programs bring real platforms into the simulation environment. Exercise participants, e.g. aircraft, ships, and land vehicles, have complete autonomy within the constraints imposed by the Rules of Engagement. Each platform periodically collects its own kinematic information from real systems and detection/tracking information from tactical sensors. This information is encapsulated in messages that are transmitted over RF datalinks to a "core system", and is then communicated to other participants. Instrumentation PDUs are communicated via low bandwidth RF datalinks, on the order of 1200bps - 121kbps. Field instrumentation programs will also rely on satellite communication.

The majority of new procurements recommending DIS are instrumentation programs. The following are examples of new or upgrade programs requiring DIS: Battle Force Tactical trainer (BFTT), Tactical Combat Training System (TCTS), Mobile Automated Instrumentation Suite (MAIS), Tactical Aircrew Combat Training System (TACTS), Joint Aircrew Combat Training System (JACTS), and National Test Center (NTC).



## 2.1 Protocol Data Units

Instrumentation systems require only a subset of the information contained in the current DIS PDU's. Consequently, research efforts [3] are recommending reduced-sized PDUs for these applications (i.e., Field Instrumentation (FI) PDUs). The FI PDUs will be transformed to "normal" DIS PDUs before entering the simulation environment.

## 3 Simulations and Simulators

Simulation is a computer replication of the behavior of entities or collections of entities (units). Simulated entities may be fully or partially automated. Simulations/simulators will use the DIS PDUs [1] to communicate data from one simulation entity to another. Simulations/simulators will be interconnected via Local Area Networks (LANs) and Wide Area Networks (WANs) which can provide high data rates on the order of Gigabits/s. It is anticipated that DIS LANs will support 1,000 entities and WANs must support 100,000 entities by the mid 1990's.

The only new procurement recommending DIS is the Close Combat Tactical Trainer (CCTT); however, there are numerous existing simulations and simulators which will use DIS for interoperability. The Interservice/Industry Training System Conference (I/ITSC) DIS Interoperability Demonstration will be an example of the connection of existing simulators. Approximately 25 simulators from 16 participants will be interconnected for the first DIS exercise.

### 3.1 Protocol Data Units

According to [1], DIS functional requirements are to provide: Entity Information, Entity Action, DIS Management, and Environment Information. Within each category, PDUs have been defined or recommended to satisfy specific requirements. The September 1991 version of the DIS standard defines ten required PDUs and six recommended interim PDUs.<sup>2</sup> A summary of the DIS functional requirements is presented in Table 1. A more detailed explanation of the PDUs can be found in [1].

---

<sup>2</sup>The September version of the DIS standard specifies three recommended PDUs for Update Threshold Control. At this writing, those PDUs have been removed from the standard and therefore, will not be included in this document.

- I. ENTITY INFORMATION
  - A. Entity State (R)
    - 1. Entity State PDU
  
- II. ENTITY INTERACTION
  - A. Weapons Fire (R)
    - 1. Fire PDU
    - 2. Detonation PDU
  - B. Logistics Support (R)
    - 1. Service Request PDU
    - 2. Resupply Offer PDU
    - 3. Resupply Received PDU
    - 4. Resupply Cancel PDU
    - 5. Repair Complete PDU
    - 6. Repair Response PDU
  - C. Collisions (R)
    - 1. Collision PDU
  - D. Electronic Interaction (NR)
    - 1. Emitter PDU
    - 2. Radar PDU
  
- III. DIS MANAGEMENT
  - A. Network Management (NR)
  - B. Simulation Management (NR)
    - 1. Activate Request PDU
    - 2. Activate Response PDU
    - 3. Deactivate Request PDU
    - 4. Deactivate Response PDU
  - C. Performance Measures (NR)
  
- IV. ENVIRONMENT INFORMATION
  - A. Changes in the Terrain (NR)
  - B. Weather Conditions (NR)
  - C. Degrees of Ambient Illumination (NR)
  - D. Other Environmental Effects (NR)

Table 1  
 DIS Functional Requirements:  
 Required (R) and Non-Required (NR) PDUs

### 2.1.3.2 Processing Constraints

There are two types of simulation LANs: homogeneous and heterogeneous. A homogeneous LAN is one in which all equipment (i.e., computing platforms, image generators(IGs), and simulation models) is provided by a single vendor. For example, SIMNET constitutes a homogeneous LAN. Within this environment, processing delays are usually constant and predictable across all simulators. Conversely, a heterogeneous LAN is composed of dissimilar computing platforms (PCs, workstations, etc.), IGs (fixed versus dynamic priority) and simulation models. A heterogeneous environment introduces a range of operating speeds and performance to the network. One of the results of this heterogeneity is a reduction in the number of entities that can be simultaneously represented on the network. An example of a heterogeneous LAN is the I/ITSC demonstration.

An analysis to determine the maximum number of entities that can be simultaneously represented on the I/ITSC network has identified five simulator processing constraints:

- (i) the bandwidth of the physical medium,
- (ii) the rate at which the physical interface hardware can read/write information (in PDUs/sec),
- (iii) the rate at which data can traverse the communication protocol stack (in PDUs/sec),
- (iv) the number of entities each simulator can track, and
- (v) the number of dynamic coordinate systems each simulator's IG can manage.

From a survey of I/ITSC demonstration participants, values for constraints (ii) through (v) have a broad range as shown in Figure 1.

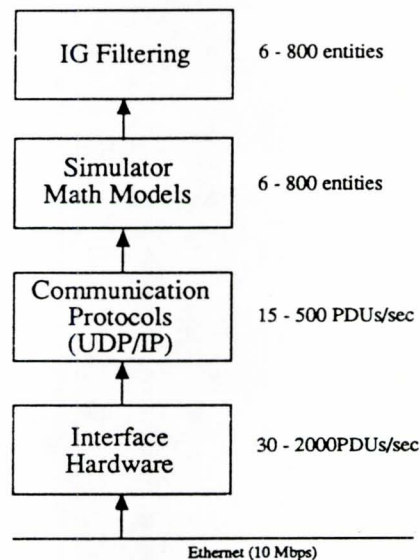


Figure 1. Processing Delays in Simulators



Using a bandwidth analysis<sup>3</sup> similar to that presented in [2], let's examine a sample network traffic analysis and identify which simulator delays will present problems. The exercise will consist of 100 tanks, 11 aircraft, and 1 ship which broadcast Entity State (ESPDU), Fire (FPDU), and Detonation (DPDU) PDUs (no Emitter PDUs (EPDU) or tactical voice links). In a low conflict environment (i.e. only ESPDUs are generated and they occur at .2 Hz per entity), traffic on the DIS network will be on the order of 55,018 bits/sec or 22 PDUs/sec; a low rate that can probably be handled by most simulators participating in an exercise. But what if the environment is high conflict, where PDUs are broadcast at the following rates: 2 Hz for a tank ESPDU, 8 Hz for an aircraft ESPDU, 1 Hz for a ship ESPDU, .1 Hz for all FPDUs, and .1 Hz for all DPDU's. The PDU traffic increases to 843,930 bits/sec or 311 PDUs/sec. The results of this analysis are shown in Table 2. The following formulas were used to determine the size of each PDU (in bits):

<u>PDU</u>	<u>FORMULA</u>	<u>REMARKS</u>
ESPDU	$1152 + 128A$	A = # of articulated part records
FPDU	704	
DPDU	$800 + 128H$	H = # of articulated parts hit
EPDU	$192 + E(160 + B(304 + 96T))$	E = # of emitter beams B = # beams per emitter T = # of targets per beam

While 843k bits/sec will not exceed the bandwidth of Ethernet or IEEE 802.3, the rate of 311 PDUs/sec begins to push the upper bound of the processing capability of some simulators. For example, IST's PC-based simulators can process only 75 PDUs/sec at the Ethernet interface. Also, a 16 MIP single-processor machine using the UDP/IP communication protocols and running only one process, i.e. receiving DIS PDUs but not dead reckoning position, can process only 200-250 PDUs/sec. When a second process is added, that rate drops to 80-100 PDUs/sec. A rate of 311 PDUs/sec would quickly overwhelm both the Ethernet interface and the communication protocols of these simulators.

---

<sup>3</sup> This analysis was conducted by Ken Doris of Grumman Space and Electronics.

**SAMPLE PDU SIZING**

	<i>A</i>	<i>H</i>	<i>E</i>	<i>B</i>	<i>T</i>	<i>ESPDU</i>	<i>FPDU</i>	<i>DPDU</i>	<i>EPDU</i>
TANK	5	1	1	1	1	2220	1132	1356	1180
AIRCRAFT	20	2	3	1	2	4140	1132	1484	2588
SURFACE SHIP	50	5	10	1	5	7552	1132	1868	10060
<i>OVERHEAD BITS/PDU=</i>									428

**SAMPLE RATES PER ENTITY TYPE PER PDU TYPE**

	<i>LOW RATE</i>				<i>HIGH RATE</i>			
	<i>ESPDU</i>	<i>FPDU</i>	<i>DPDU</i>	<i>EPDU</i>	<i>ESPDU</i>	<i>FPDU</i>	<i>DPDU</i>	<i>EPDU</i>
TANK	0.2	0	0	0	2	0.1	0.1	0
AIRCRAFT	0.2	0	0	0	8	0.1	0.1	0
SURFACE SHIP	0.2	0	0	0	1	0.1	0.1	0

**SAMPLE EXERCISE TRAFFIC ESTIMATES**

% ENTITIES AT HIGH RATE		0%	20%	40%	60%	80%	100%
% ENTITIES AT LOW RATE		100%	80%	60%	40%	20%	0%
100 TANKS		44,400	129,296	214,192	299,088	383,984	468,880
11 AIRCRAFT		9,108	80,726	152,344	223,962	295,580	367,198
1 SHIPS		1,510	2,779	4,047	5,315	6,584	7,852
0 TACTICAL VOICE LINKS		0	0	0	0	0	0
0 TACTICAL DATA LINKS		0	0	0	0	0	0
TOTAL TRAFFIC	BITS/SEC	55,018	212,801	370,583	528,365	686,147	843,930
	PDU <sub>s</sub> /SEC	22	80	138	196	254	311

Table 2  
Sample Traffic Analysis for DIS Exercise



From these initial calculations we can make the following assertions about simulator processing constraints:

- 1) The interface hardware and communication protocols (i.e., UDP/IP) processing constraints will present the biggest problem in determining the number of entities participating in an exercise. Scenario development will not solve this problem but lower layer filtering (i.e, multicast) can. In this example where all entities are in high conflict, major problems could occur with the low-performance simulators. Major problems which may be encountered include simulator failure, network latency, or network failure. A rate of 311 PDUs/sec will quickly overwhelm the lower bounds of 30 and 15 PDUs/sec for hardware interface and communication protocols, respectively.
- 2) Simulator math models and IG constraints will be secondary problems and may be alleviated through upper layer filtering and/or prioritization. In fact, most simulators already filter or prioritize the entities in their field of view, but some simulators do not. Filtering is dependent on the application. Currently, simulators tend to use proximity between entities as a criteria for filtering. However, more elaborate filtering schemes will be necessary as DIS evolves. Scenario development can help this constraint by placing entities out of densely populated areas (i.e., on different pieces of terrain) so as to "filter" for the math models and IGs.

## 2.2 The Seams

The mechanism which connects the DIS Applications and allows them to communicate is the communication protocols. The communication protocols will span the entire network and must be seamless across the boundaries of simulations, wargames, and instrumentation. To make its communication useful to support its operation, each Application will require certain services. If the services differ from one Application to the next, the communication protocols must be able to compensate for the differences while maintaining a seamless service across the boundaries. The communication protocols are of two types: the lower layer protocols provide the necessary communication services (e.g., multicast) and the upper layer protocols provide the means through which the Applications communicate (i.e., the PDUs).

Fundamentally, DIS is integrating three types of simulation and with each comes different requirements. The Application's requirements will be satisfied by selecting the appropriate communication services. For example, one DIS requirement is multicast. However, the taxonomy of multicast is diverse and it is likely that more than one form of multicast will be required to satisfy the requirements of the Applications. Field Instrumentation, which uses core systems as a central base for connecting player units, would favor a server or centralized type of multicast. In contrast, simulations/simulators will require a distributed form of service. The communication protocols must accommodate both types of multicast and allow them to interwork in a WAN environment.

For the Application protocols, there are also diverse requirements. The simulation/simulator PDUs are well known [1] while the wargame and field instrumentation PDUs are still being developed. However, we do know that FI PDUs and wargame PDUs will be different from what is currently contained in [1]. As noted in 2.1.2.1, current research is recommending reduced-size FI PDUs. The FI PDUs will be transformed to "normal" DIS PDUs before entering the simulator environment. To integrate wargames into the simulator environment, force level simulations must aggregate item level to unit level and deaggregate unit to item level representations. This transformation of Application layer information must take place at the boundaries of each domain.

The interworking of both communication and Application services must take place at the boundaries of each Application. Using the concepts presented in [4], the boundaries can be defined in the following manner. In a broad sense, each Application can be considered a Cell. A Cell is a collection of homogeneous simulation entities connected by a network [4]. For example, a CCTT LAN is a Cell as is a homogeneous distributed Wargame or a MAIS C<sup>3</sup> Center with associated player units. In a WAN, the Cells will be interconnected by Cell Interface Units (CIUs) or Cell Adapter Units (CAUs). CIUs will be used to connect Cells to the WAN which conform to the DIS Standard; CAUs will be used to connect non-standard Cells by translating information. CCTT will be a DIS Standard Cell and will therefore use a CIU to connect to the WAN. Examples of non-standard Cells would be SIMNET and MAIS. These Cells will require a CAU. Therefore, the communication protocols (both lower layer and upper layer) must interwork at the Cell "boundaries" which are the CIUs/CAUs.

As defined in [4], the CIUs/CAUs will provide intercell services such as message filtering (i.e., multicast group management), translation of messages, data compression, and aggregation/deaggregation of simulation entities. These functions are both communication and Application services. Thus, the CIUs/CAUs become an important part of the "seamless interconnection" of the Applications, as shown in Figure 2.

It is worth noting that while Cells are defined as standard and non-standard, each Cell will be composed of LANs which may be either homogeneous or heterogeneous (see section 2.1.3.2). As discussed earlier, heterogeneity introduces a wide range of performance and also limits the number of entities that can be simultaneously represented on the network. Applying this knowledge to the field instrumentation and wargame environments, it is possible that each Application will have different performance ranges. If some form of compensation is required (based on performance ranges) to integrate the Applications, this too will be a function of the CIUs/CAUs.



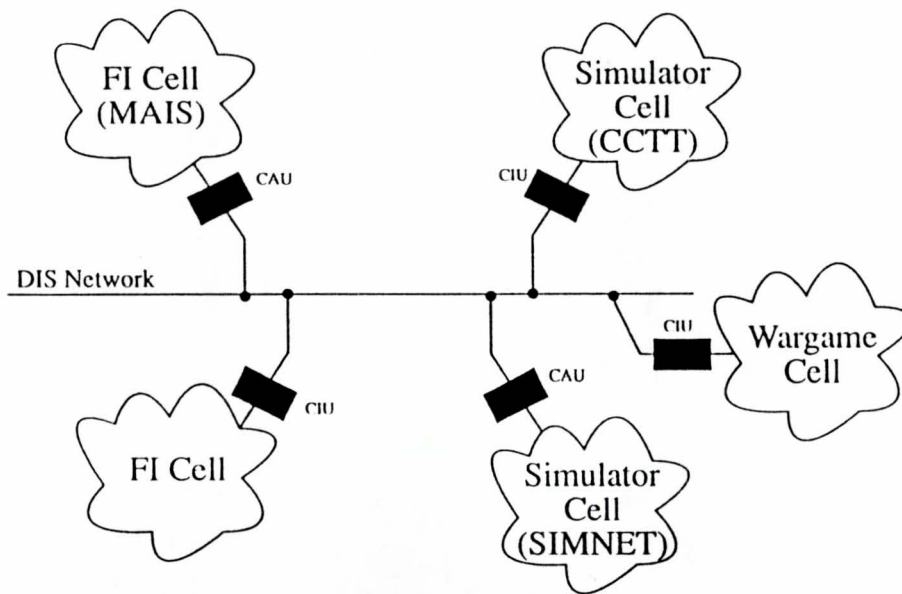


Figure 2. Integrating DIS Applications

It is apparent that a range of services must be accommodated and interworked within the DIS network. Since this will be accomplished by the communication protocols, it is necessary that each Application clearly define its requirements. As noted in the Introduction, only the service requirements for Simulations/Simulators have been identified. Very little information about Wargames and Instrumentation Applications has been presented to the CASS. While the service requirements for Wargames and Instrumentation may not differ greatly from Simulators, it is necessary to understand all requirements so that a "seamless interconnection" of the three Applications can be achieved.

From the discussion above, we know that the Cell boundaries (i.e. CIUs and CAUs) will be responsible for integrating the Applications, hence they are the "seams". Since the CIUs/CAUs will be responsible for converting both lower layer and upper layer protocols, the configuration and design are a critical factor in the success of DIS. Some conversions will be easier than others. For example, non-DIS to DIS within a single Application will require less translating than non-DIS to DIS conversion across Applications which require different communication services. Because of the range of potential conversions, there is considerable performance impact on the design of the CIUs/CAUs. These boundaries could also be a performance bottleneck depending the magnitude of conversions being performed. It is worth noting that the seams may "burst" if we put too much strain on them. Integrating the communication protocols at the Cell boundaries is an area for further research.

The remainder of this report will focus on the *known* communication service requirements and how these requirements are matched with communication protocols to provide the required service.

## SERVICE REQUIREMENTS

CASS has developed a set of communication service requirements for the communication system of the DIS standard. These requirements are based on experience with state-of-the-art distributed simulation activities, as well as projections based on anticipated use and evolution of technology base. The purpose of the communication subsystem for DIS is to provide an appropriate interconnected environment (i.e., hardware, software, and simulations) for effective integration of both locally and globally distributed simulation entities.

Distributed simulation environment support requires various types of communication. Data communications, including voice, may have real-time requirements and will likely be augmented to include such things as video and other forms of pictorial information. It is desirable from a performance and communications management perspective for each of these forms of traffic to share communications facilities, instead of having disjoint facilities for each.

A summary of the communication service requirements developed by the CASS [2] is shown in Table 3.

- Unicast/Multicast/Broadcast
- Real Time/Non-Real Time Operating Speeds
- Reliable/Unreliable Transmission
- Seamless Local/Global Communication
- Bulk Transfer/Bursty Interactive Traffic
- Multi-Level Security
- Network Management
- Synchronization

Table 3  
DIS Communication Service Requirements

### Application Service Characterization

The service requirements in Table 3 are of two types: services required by all PDUs and services required by some PDUs. The services that apply to all are network management, security, seamless local/global communication, and synchronization. The remaining services vary for each PDU depending on what makes its communication practical. This remaining list of requirements is referred to as Application Service Characteristics (ASC) and includes: unicast/multicast/broadcast; reliable/unreliable; real time/non-real time; and bulk transfer/bursty interactive traffic. The ASCs were used to characterize the services required by the PDUs. The results of this characterization is a service model which will be used to develop the interface to the communication and lower layers.



### 3.1.1 Definitions

<b>Broadcast Mode (BC)</b>	A transmission mode in which a single message is sent to all network destinations, i.e. one-to-all. Broadcast is a special case of multicast.
<b>Multicast Mode (MC)</b>	A transmission mode in which a single message is sent to multiple network destinations, i.e. one-to-many.
<b>Non-Real Time Service</b>	Any protocol function which does not require real time service. (see Real Time Service.)
<b>Real Time Service</b>	A service which satisfies timing constraints imposed by the service user. The timing constraints are user specific and should be such that the user will not be adversely affected by delays within the constraints. <sup>4</sup> (DIS requires 5% of all data be processed within 100ms and 95% be completed within 300ms <sup>5</sup> , therefore the DIS real time threshold is 100ms.) <sup>6</sup>
<b>Reliable Service</b>	A communication service in which the number and type of errors that the user finds in the data is acceptable for the application. Reliable communication may require specific mechanisms in order to achieve the user's requirements: error detection and notification, such as bit errors based on a too high bit error rate as defined by the user, or error detection and correction from PDU errors, such as bit errors, duplicated PDUs, missing PDUs, or out-of-sequence PDUs.
<b>Unicast Mode (UC)</b>	A transmission mode in which a single message is sent to a single network destination, i.e. one-to-one.
<b>Unreliable Service</b>	A communication service in which transmitted data is not acknowledged. Such data typically arrives in order, complete, and without errors. However, if an error occurs, nothing is done to correct it (e.g., there is no retransmission). This type of service is also known as "best effort".

---

<sup>4</sup> Some data communications, e.g. voice, may require compensation to meet the timing constraint.

<sup>5</sup> These numbers, taken from [2], are based on limited experience and are provided only as an experimental baseline.

<sup>6</sup> The amount of delay acceptable in a given application depends on the nature and intended use of the application. For some applications the acceptable delay may be less than 100ms or greater than 300ms.

### 3.1.2 PDU Characterization

The application services for the required and recommended DIS PDUs are shown in Tables 4 and 5, respectively. Rationale for the characterization of PDUs can be found in [2] and [5].

Inter-entity communication in a distributed interactive simulation environment consists largely of packets sent between two or more of the simulation participants. These packets are usually small, <250 bytes, and constitute the majority of PDU traffic. All PDUs listed in Table 4 and 5 fall into the "small packet" characterization. There are situations which mandate non-real time, point-to-point, reliable bulk transfer, however. Such situations arise when moving large items such as database files or video images. Although bulk transfer/bursty interactive traffic are included as application service characteristics, it is not presented in the summary tables for the following reason. The bulk transfers fall into the Network and/or Simulation Management functions. Consequently, bulk transfer is considered a special case.

	Reliable	Unreliable	BC	MC	UC	Real Time
Entity State		*		*		*
Fire	<u>desired</u>	*		*		*
Detonation	<u>desired</u>	*		*		*
Service Request		*			*	(few seconds)
Resupply Offer		*			*	(few seconds)
Resupply Received		*			*	(few seconds)
Resupply Cancel		*			*	(few seconds)
Repair Complete		*			*	(few seconds)
Repair Response		*			*	(few seconds)
Collision	*				*	*

Table 4  
Characterization of Required DIS PDUs

	Reliable	Unreliable	BC	MC	UC	Real Time
Emitter	<u>desired</u>	*		*		*
Radar	<u>desired</u>	*		*		*
Activate Request		*			*	
Activate Response		*			*	
Deactivate Request		*			*	
Deactivate Response		*			*	

Table 5  
Characterization of Non-Required DIS PDUs

It is anticipated that DIS Management will require additional capability beyond the Activate and Deactivate PDUs. Although these capabilities are not specified in the September 1991 version of the standard, Table 6 projects additional application requirements for these areas.

	Reliable	Unreliable	BC	MC	UC	Real Time
Network Management		*			*	
Simulation Management	*	*		<u>desired</u>	*	

Table 6  
Characterization of DIS Functional Requirements

### 3.1.3 Communication Classes

From the ASC Tables shown above, three service models emerge as characterizing the DIS application.

#### CLASS 1 Unreliable Multicast

A mode of operation where the multicast service provider uses no added mechanisms for reliability except those inherent in the underlying service.



**CLASS 2 Unreliable Unicast**

A mode of operation where the unicast service provider uses no added mechanisms for reliability except those inherent in the underlying service.

**CLASS 3 Reliable Unicast**

A mode of operation where the unicast service provider uses whatever mechanisms are available to ensure the data is delivered in sequence with no duplicates and no errors.

The service model is shown in Table 7. As can be seen from Tables 4-6, there exists a desire for a reliable multicast service. However, there are no such protocols in existence - it is still a research problem. With this in mind, a Class 4 service will emerge as the standards and protocols to support reliable multicast are developed.

CLASS 1 Unreliable Multicast	CLASS 2 Unreliable Unicast	CLASS 3 Reliable Unicast	CLASS 4 Reliable Multicast
Entity State	Service Request	Collision	<i>future</i>
Fire	Resupply Offer	Simulation Management	
Detonation	Resupply Received		
Emitter	Resupply Cancel		
Radar	Repair Complete		
Simulation Management	Repair Response		
	Network Management		
	Activate Request		
	Activate Response		
	Deactivate Request		
	Deactivate Response		

Table 7  
DIS Application Service Model

### 3.2 Interim and Long-Range Requirements

The majority of the communication services required by DIS can be satisfied with COTS protocols; however, there are requirements which cannot. Consequently, the DIS service requirements fall into two categories: interim services, which are required to support immediate DIS experiments, demonstrations, and tests, and the customization or long-range services, which require development. The services which apply to all the PDUs, namely network management, security, seamless local/global communication, and synchronization, are interim services because they can be satisfied with COTS products.<sup>7</sup> From the service model developed in Table 7 above, Classes 2 and 3 are also interim services. Actually, reliable and unreliable unicast are basic communication services used by many types of applications. To a certain extent, Class 1 is also an interim service. Remember from the definitions in Section 3.1.1 that broadcast is a subset of multicast. Unreliable broadcast is another basic communication service used by many applications.

The real problem with any application is the development or customization needed to meet unique requirements (not the basic communication services used by many applications). For DIS, one such customization is multicast (and consequently Communication Classes 1 and 4). As shown in Section 2.1.3.2, large scale DIS exercises will require multicast to selectively transmit information among simulators. With multicast, extraneous messages are never generated. Thus, it is not just a filtering technique but an optimization of the communication resources. Multicast will reduce the amount of PDU traffic a simulator must process by not sending information which is of no interest to it. To allow hundreds of thousands of entities to simultaneously participate in simulation exercises [4], multicast will be required. DIS desires a full range of multicast capabilities including: reliability, static and dynamic groups, lower layer filtering, flexible naming and addressing, and dynamic management by upper layers. Unfortunately, these capabilities cannot be satisfied with any current COTS protocol.

---

<sup>7</sup> Security may be an exception, the CASS is currently investigating this requirement.



## 4. STANDARDS AND PROTOCOLS

DIS requires state of the art communication services and protocols. To facilitate the interoperability of dissimilar simulations and reduce cost, industry communication standards will be adopted to maximize the use of COTS products and to maximize the base of practical technical knowledge.

### 4.1 Standards

At the January 1992 meeting of the CASS, proposals for the DIS interim architecture were discussed. Three proposals were submitted: one based on Internet standards, one based on OSI standards, and one based on the Navy's Survivable Adaptable Fiber Optic Embedded Network (SAFENET) architecture. Because the SAFENET architecture is composed of predominantly OSI protocols, the discussion focused on Internet and OSI standards. Technically both the Internet and OSI standards met the service requirements; however, questions were raised about the maturity of OSI.

Maturity can be defined by the following characteristics: Protocol Maturity, Product Availability, Product Maturity, Cost, Implementations, and Risk. Protocol maturity is measured by the number of years since the protocol was published. As a protocol becomes stable, fewer revisions are necessary to the standard and the protocol and its associated products become mature. The number of products available from vendors is the measure of protocol availability. Product maturity must be based on protocol maturity and is consequently affected by revisions to the standards. The cost of products is linked to protocol availability. The fewer the products on the market, the more expensive they are. The number of implementations and risk are also related. As the number of implementations grows, risk is reduced. This is a result of gaining experience with the protocols and lessons learned on product interoperability.

#### 4.1.1 Internet Protocol Suite

The Internet Protocol Suite (IPS) is a family of protocols based on the Transmission Control Protocol/Internet Protocol (TCP/IP) standards. The IPS base stack was published in the mid 1970's. Products are available from most vendors for both workstation and PC platforms. The cost varies and in most vendors is included in the cost of the hardware. The IPS is the defacto standard for computer networking and boasts numerous implementations, most notably the global Internet. There is no risk associated with products and only minimal risk associated with in-house implementations. Due to the large installed base of IPS and the twenty years of development, these protocols and their corresponding products are very mature.

One interesting note on the maturity of this protocol suite:

*It is so mature it suffers from old age.*

The Network Layer protocol IP is running out of address space, which will be depleted within the next two years [10]. Several proposals are on the table to solve the problem. Most proposals are merely "patches" which will extend the life of IP; they are not long term solutions to the real problem. The most widely accepted proposal is to use the OSI Network layer addressing scheme and shift to the OSI Connectionless Network Protocol (CLNP). With the exception of the addressing structure, IP and CLNP are practically the same protocol. They provide the same functionality and are virtually indistinguishable. Another sign of the maturity of IPS is performance problems with the Transport layer protocol TCP. One such problem is a result of octet sequencing. The sequence number space has too fine a granularity. Consequently, while the sequence number field is "large" enough, it wraps too fast at high speeds. Again, there are temporary solutions, but they have other problems which impact performance. In contrast, the corresponding OSI Transport protocol uses message sequencing and thereby avoids this problem.

#### **4.1.2 Open Systems Interconnection**

The other option for protocol interoperability is to comply with the Government Open Systems Interconnection Profile (GOSIP) mandate which has been in effect since August 1990. GOSIP is the U.S. Government program for adoption of OSI standards across all Federal agencies. DIS will benefit from the OSI/GOSIP architecture through: reduced cost, increased interoperability (both nationally and internationally), and increased application-level functionality. The Institute for Simulation and Training (IST) has developed the DIS protocol standard with the goal of using the GOSIP protocols. Unfortunately, GOSIP has not reached the level of maturity of the Internet protocol suite and consequently, many view GOSIP compliance as a long-term goal.

The OSI base stack was published in the mid 1980's. Based purely on the chronological age, the IPS base stack is more mature; however, many of the OSI protocols are based on their Internet predecessors and therefore gain stability from lessons learned. Current documentation shows approximately 450 OSI products from 80 suppliers. From a recent survey of major computer vendors, users have a choice of an IPS or OSI stack for workstations; no OSI products for PC platforms have been identified. Several OSI products are still in development by vendors, for instance Network Management. Product maturity is hard to measure, but due to the limited installed base of OSI products, maturity is not near the level of the IPS. The cost of OSI products is higher than that of the IPS for several reasons. First, the development of the IPS was funded in large part by Federal agencies through research grants. Therefore, vendors did not have to spend their own money to mature the protocols and products. In contrast, OSI is being developed by industry. Consequently, the capital expended in the development of both the protocols and products is passed on to the customer. From a limited survey, OSI base stacks range from zero to several thousand dollars. However, one major computer vendor ships all computer systems with dual (Internet and OSI) stacks.

Although OSI cannot boast implementations as numerous as IPS, OSI is slowly growing and is even being integrated into the global Internet. The National Science Foundation Network



(NSFnet) backbone has offered national CLNP<sup>8</sup> service since August 1990. There are approximately 25 regional networks which are part of this "OSI over the Internet" testbed, including: Energy Sciences Network (ESnet), NASA Network (NASAnet), Southeastern Universities Research Association Network (SURAnet), and New England Academic and Research Network (NEARnet). These regional networks route both Internet and OSI traffic. There is also a world X.400 (OSI electronic mail) backbone connecting the U.S., Europe, and Pacific Rim. In addition, several new major Government procurements are specifying OSI/-GOSIP communication services. These procurements include the Department of the Treasury, the Department of Energy, and the Department of Agriculture. The Federal Aviation Administration (FAA) is also starting new OSI research projects.

There is risk associated with OSI products due to limited experience with the protocols. However, the integration of OSI into the Internet will help reduce the risk by exposing industry, academia and government to the protocols and products. In fact, the implementation of OSI routing protocols running in NSFnet is now available free to the public. As OSI implementations are tested and made available to the public, the risk associated with OSI products will diminish.

## 4.2 Protocols

The DIS communication architecture is composed of a suite of protocols which satisfy the established service requirements (presented in Section 3). For example, the bulk transfer service requirement will be satisfied by a file transfer protocol, for which DIS can choose either Internet or OSI COTS products. Both the Internet and OSI standards are composed of a large number of protocols, not all of which are required by DIS. For the remainder of this paper, the term *base stack* will be used to designate the subset of Internet or OSI protocols required by DIS for operation. The protocol suite which satisfies the service requirements is shown in Table 8.

The base stack shown in Table 8 encompasses all *known* DIS communication service requirements. However, as stated previously, not all requirements can be met with COTS products. One of these services is multicast. Consequently, multicast will have to be phased in over a period of years as the services and protocols are adopted by the standards bodies. As a result, the base stack for the interim architecture will differ from the base stack for the long-range architecture.

---

<sup>8</sup> CLNP is analogous to the Internet IP protocol.

Required Services	Base Stack
• Network Management	• Network Management & Terminal
• Bulk Transfer	• File Transfer
• Reliable Unicast	• Connection-Oriented Transport
• Unreliable Unicast	• Connectionless Transport
• Unreliable Multicast	• Network Layer Multicast
• Reliable Multicast	• Transport Layer Multicast
• Seamless Local/Global Communication	• Connectionless Network
• Security	• Multi-Level Security
• Synchronization	• Time

Table 8  
Summary of DIS Service Requirements and Protocols

#### 4.2.1 The Interim Choices for Multicast

Today there are three possibilities for multicast: IP Multicast (IPMC), Stream-II (ST-II), and the Xpress Transfer Protocol (XTP). IP multicast, part of the IPS family, is the only commercial multicast product but has only limited availability. Further, IPMC in its current form does not meet the requirements of DIS: *It is not a real-time protocol*. For this reason, a modified IPMC has been proposed. This approach relies, in part, on a protocol which has no known implementations. This proposal would require significant development cost and would introduce significant risk since there is no prototype implementation on which to base the protocol. The development of such a protocol would require a minimum of a year plus an additional two years to introduce it into the global Internet. Only after substantial use by the Internet community would there be commercially available products. The down side to this proposal is that IP, and consequently IPMC, will migrate to a successor (possibly OSI's CLNP) in the same three year time frame it would take to develop this protocol.

The Internet ST-II protocol is also part of the IPS family but is considered an experimental protocol. It is not commercially available but is the only protocol which has been proved to work for DIS applications (i.e., multicast and meets real time requirements). ST-II is the multicast protocol used by SIMNET and has been successfully applied to large scale exercises such as WAREX '90. The Stream protocol has a multi-destination simplex structure where a stream is a directed tree carrying traffic from a source to the destination. ST-II is a Network Layer protocol which means that routers must be capable of switching ST-II traffic for internetworking. The long-haul testbed for simulation applications, the Defense Simulation



Internet (DSI), currently supports IP and ST-II traffic. This compatibility with a permanent infrastructure is an added benefit for ST-II. On the other hand, ST-II is not currently available over public or "ad hoc" networks.

The last protocol, XTP, is commercially available but offered by only two vendors. Although this protocol offers both multicast and high speed, it has not been proven for DIS applications and has never been proven for wide area networks. While advertised as high-speed, it has yet to significantly out perform IPS' TCP or OSI's TP4. XTP is neither an IPS nor an OSI standard.

Although the ST-II and XTP protocols are candidates to meet interim multicast needs of DIS, they do not meet the long-term requirements. Consequently,

**Multicast must be developed for DIS.**

To do so, the DIS community has several options: develop a near-term version of multicast, such as the proposal to modify IPMC, or develop multicast for the long-range architecture (i.e., the GOSIP compliant stack). However, given the similarity of IP and CLNP, it would be possible to develop a multicast solution which is relatively "protocol independent" and can evolve to GOSIP as the architecture does. There is no obvious justification to develop unique solutions for each phase of the architecture and considerable reason not to.

## 5. RESEARCH

Under this contract, IST was responsible for several research tasks related to the DIS communication architecture. Three main tasks were undertaken: evaluate the overhead of Abstract Syntax Notation One (ASN.1) and assess its impact on the performance of DIS, develop a formal description of DIS using the Estelle Formal Description Technique (FDT) to test the protocol, and develop OSI multicast service definitions and protocol specifications which satisfy DIS long-range requirements. The ASN.1 and Estelle descriptions of DIS have been completed; the task associated with long-range multicast requirements is on-going. As mentioned previously, the problem with any application is the customization required to meet unique requirements - multicast is one of these unique requirements. Hence, this is where much of the future research is needed. A major aspect of the on-going research is to ensure a graceful evolution of the architecture. The following sections give a brief summary of the accomplishments of each task. More detail can be found in the final reports for each task.

### 5.1 Abstract Syntax Notation One

In large scale DIS systems, various heterogeneous computing nodes are used as vehicle simulators and as control and data logging elements. Such nodes may differ in the way they represent typed data. Integers, for example, are typically stored in two or four bytes and their binary representation is usually based on either the 1's complement or the 2's complement notation. Likewise, characters are usually stored in either ASCII or EBCDIC. To overcome the incompatibilities resulting from these differences, interoperable DIS communication systems can employ a common *abstract syntax* that is independent of machine architectures as well as of compilers used to produce the application code on the different nodes. Using the common abstract syntax, two dissimilar DIS nodes can exchange PDUs as follows:

- 1) The PDU is first transformed from its local representation at the sending host to a transfer syntax (i.e., a representation based on the common abstract syntax). This transformation is performed using a set of rules called the *encoding rules*.
- 2) The PDU, in transfer syntax, is transmitted down the communication stack of the sending host across a physical link, up the communication stack of the receiving host, and is delivered in the same transfer syntax to the Presentation layer of the receiving host.
- 3) At the receiving host, the PDU is transformed to the local representation using a set of rules called the *decoding rules*.

For OSI compliant networks, two standards have been adopted<sup>9</sup>: i) The Abstract Syntax Notation One (ASN.1) [17] is used to solve the problem of heterogeneous (dissimilar) local representations

---

<sup>9</sup> While ISO has adopted the standards, it does not require their use.



of data and ii) ASN.1 Basic Encoding Rules (BER) [18] are a set of encoding rules used to produce a transfer syntax for the exchanged data based on ASN.1.

To assess the impact of using ASN.1 in the DIS environment, three experiments have been conducted [15] using the ASN.1 tools of the ISO Development Environment (ISODE). The first experiment, the Isolation Model, computed the encoding and decoding overhead associated with ASN.1 in DIS simulators. In the Single-Host Model experiment, the BER routines were executed within the ISODE system running on Ethernet. The sender total overhead with respect to a single host was measured. In the last experiment, the Network Model, the end-to-end delay between two hosts was measured.

To determine the feasibility of using ASN.1 in the DIS environment, the results of the Single-Host Model experiment were compared to the latency data presented in [2]. CADIS states that the maximum latency between the application and physical layers of any DIS simulator shall be 10 milliseconds (or 20 milliseconds total for host and destination). As shown in Table 9, the total ASN.1 overhead associated with the encoding and decoding ranges from 1.89 milliseconds (Resupply Cancel PDU) to 13.79 milliseconds (Entity State PDU). The time required to encode/decode the ESPDU is over half the total time (20 milliseconds) allocated for sending/receiving the data. From this experiment, ASN.1 is too costly for the real-time nature of DIS. A close examination of the results indicates a direct relationship between ASN.1 overhead and the PDU complexity. Most DIS PDUs contain many different type fields (integers, booleans, real numbers, and short character strings). This makes BER encoding (decoding) of a DIS PDU more complex and tends to increase time for the encoding (decoding). It should be noted that the results presented in Table 9 reflect the ASN.1 overhead measured on a Sparc1 workstation using a RISC processor.

PDU	Overhead with ASN.1	Overhead without ASN.1	ASN.1 Overhead
Entity State	21.76	7.97	13.79
Fire	16.28	7.27	9.01
Detonation	21.69	7.94	13.75
Collision	13.37	7.39	5.98
Service Request	12.07	7.1	4.79
Resupply Offer	12.54	7.1	5.44
Resupply Received	15.94	7.81	8.13
Resupply Cancel	9.221	7.33	1.89
Repair Complete	9.881	7.32	2.56
Repair Response	9.689	7.04	2.65

Table 9  
ASN.1 Overhead (in milliseconds)



The research documented in [15] demonstrated the importance of abstract syntaxes in distributed applications and described the basic steps in applying ASN.1 to the transmission of a DIS PDU. Based on the experiments, it was observed that ASN.1 is platform independent; the Network Model experiment exchanged the encoded (decoded) data among three different platforms (i.e., Sparc1, Motorola, and Harris). The platform independent nature of ASN.1 could eventually be a useful tool for simulator interoperability.

One of the main concerns about ASN.1 is whether it is feasible or practical to use in real-time applications. In particular, the overhead due to encoding and decoding processing in ASN.1 can become too substantial and time consuming for time-critical applications. ASN.1 has significant processing overhead which could be reduced by using improved versions of encoding/decoding syntaxes<sup>10</sup> and newer and faster processors could minimize the effects of processing overhead for real-time applications. Currently, however, the overhead of ASN.1 is unacceptable for DIS.

## 5.2 Estelle Formal Description

In an effort to test and validate DIS, IST has developed a formal specification of the DIS draft standard using the Estelle FDT and the Portable Estelle Translator - Distributed Implementation Generator (PET-DINGO) compiler. Estelle is a standard FDT [19] used to specify distributed concurrent information processing systems. The benefit of using Estelle is that it offers a structure for describing DIS in a manner that is meaningful, without the ambiguities of the protocol description. A good structure increases the readability, understandability, flexibility, analyzability, and maintainability of system descriptions.

There are two aspects of abstraction that Estelle offers. First, it is completely independent of the implementation methods, so the technique itself does not provide any undue constraints on the implementer. Secondly, Estelle offers an abstraction from the details of local system. Abstraction provides DIS with a means to specify entity interactions (i.e., behavior models) without saying anything about the actual implementation of a simulator.

A FDT, such as Estelle, is used to specify the behavior of a protocol. Traditional communication protocols (e.g., TCP or CLNP) operate in a consistent manner in all hosts. Since a formal description represents a protocol's "true" behavior, it can be used to generate automatic test suites for conformance testing and rapid prototyping. A formal description can be a great advantage when implementing the protocol. For example, it has been shown that implementing a protocol from the standard (e.g., TCP) takes 18 months, whereas implementing the same protocol from a formal description takes only 6 weeks.

The DIS protocol specified in version 1.0 is not a traditional communication protocol. The version 1.0 environment allows each simulator to receive PDUs and independently determine the

---

<sup>10</sup> Alternative syntaxes, such as the light weight syntax, were investigated and are documented in [15].

resulting behavior. Only the logistics PDUs have predetermined State Transition Models (STMs). The remaining DIS PDUs can be interpreted in a variety of ways. IST's Estelle specification is one interpretation of DIS behavior.

The approach taken to specify DIS interactions followed an initiator/responder model. (The Repair and Resupply activities have an initiator/responder type of interaction; however, other activities such as Fire and Collision are essentially non-replied interactions and do not follow the initiator/responder model.) This model uses a general view of a simulation entity from a driver's perspective, which follows a hierarchical structure. The upper modules initiate the process by requesting action; the lower modules make appropriate choices based on the request.

The model includes four basic modules: *Driver*, which is responsible for starting a protocol process and communicating appropriate decisions; *Core*, models the DIS PDU interactions and includes five embedded modules (Splitter, Combiner, Logistics, Fire, and Assess); *Network*, which models the physical/logical linkage between protocol entities; and *Network Switch*, which allows the user control of the network (e.g., the ability to interrupt the message transmission).

Within the Core module are independent sub-modules which models the behavior of DIS:

- Logistics:** The Logistics module models both the repair and resupply events of DIS. It can act as either an initiator or responder, but not both. This module uses six types of PDUs to inform the peer entity of the action taken. It incorporates reliability features by using some of the PDUs to acknowledge other PDUs. Such use of PDUs can be classified as application level acknowledgement.
- Fire:** This module is responsible for the fire and detonation events. It uses the Fire and Detonation PDUs to convey related information to peers. The Fire and Detonation PDUs do not require acknowledgement from their intended target.
- Assess:** The Assess module updates the internal representation of a simulation. For example, the Assess module represents updates to an entity's appearance during a simulation exercise when it receives an Entity State PDU. It is also responsible for representing the fire event and damage assessment caused by a munition detonation or a collision. As far as the specification is concerned, the assessment means that the module transitions from an idle state to a particular type of assessment state and then back to the idle state.

The formal description of DIS was based on a set of assumptions and constraints which can be found in [16]. The following assumptions are examples of the model's scope.

- The repair and resupply activities do not take place concurrently. This allows all four state transition models defined in the standard associated with these activities to be combined into a single state transition model.



- An entity cannot repair or resupply while in motion. This assumption allows the inclusion of collision and movement events in the logistic module.
- The interval between a weapon fire and the detonation of the fired munition is 2 seconds.

Two types of tests were conducted: Valid and Inopportune. Valid tests are those where the tester sends messages at times and in sequences that are expected or normal for the implementation under test's state. Inopportune test are those where the tester sends messages at times when they should not occur or are out of sequence.

As a result of the testing, two inconsistencies were found in the Logistics module (see [16]). The Fire and Assess state transition models were checked based on the assumptions and no ambiguities were found. The provision for the loss of packets due to network failure has been tested for the Repair/Resupply activities. The timeout mechanism in the standard has been verified as adequate. The current standard is not fault tolerant, i.e. the protocol can present misbehavior caused by network failures; however, the protocol works well in an environment of low network failure rate.

### 5.3 Multipeer/Multicast

This contract initiated work in the American National Standards Institute (ANSI), the International Organization for Standardization (ISO), and the International Telephone and Telegraph Consultative Committee (CCITT) to develop a full range of multicast functionalities. Initially, the work has concentrated in three main areas: the *Multipeer Addendum* to the OSI Reference Model, to provide an overall framework for multicast services; the *Transport Layer*, to provide end-to-end reliability; and the *Network Layer*, to provide a basic multicast data transfer facility. This work supports an overall strategy for multicast in OSI.

This work was subcontracted to two consultants: Jim Moulton of Open Network Solutions (ONS) and Joel Snyder of Opus One. ONS was responsible for developing multicast service definitions and protocol specifications related to connectionless standards; Opus One concentrated on connection-oriented multicast. The multipeer architecture work was also developed by ONS. This work is being progressed in three ANSI working groups, two ISO Sub Committees (SC), and one CCITT Study Group (SG). The ANSI working groups are X3T5.1 (OSI Architecture), X3S3.3 (Network and Transport Layers), and X3S3.7 (Public Data Networks). ISO SC 21 is responsible for standards on information retrieval, transfer, and management for OSI; this is where the OSI Architecture work is being progressed. The Network and Transport layer work from the X3S3.3 and X3S3.7 working groups is progressed through ISO SC 6, which is responsible for telecommunications and information exchange between systems. The connection-oriented multicast work from the X3S3.7 working group is introduced into CCITT through SG VII.



Due to the nature of standards work, this task has not been completed. In fact, adoption of the Network and Transport protocols are expected to take three years. Standards development is an iterative process. Technical contributions are submitted to appropriate standards committees and, in turn, the committee members comment on the technical merit of the ideas. The contribution is then revised, based on the comments of the task group, and submitted again for further review.

The standards development work funded by this contract consisted of 13 tasks. These tasks are described below. Tasks 1-7 were performed by ONS, tasks 8-13 by Opus One.

**Task 1: Develop a Multicast Connectionless Network Service Definition**

This task will be based on the existing OSI Connectionless Network Service Definition (ISO 8348 Add 1).

**Task 2: Develop a Multicast Connectionless Transport Service Definition**

This task will produce a potentially new OSI standard based upon the existing Transport Service Definition (ISO 8072 Add 1).

**Task 3: Develop a Multicast Connection-Oriented Transport Service Definition**

This task will be based on the existing OSI Transport Service Definition (ISO 8072).

**Task 4: Develop a Multicast Connectionless Network Protocol Specification**

This task will be based upon extensions/modifications to the ISO 8473 (ISO IP) and the existing OSI Connectionless Network Protocol Specification.

**Task 5: Develop a Multicast Connectionless Transport Protocol Specification**

This task will evaluate extensions to the existing OSI protocol (ISO 8602) and be based on the existing OSI Connectionless Transport Protocol Specification.

**Task 6: Develop a Multicast Connection-Oriented Transport Protocol Specification**

This task will be based upon either the existing OSI Connection-Oriented Transport Protocol (e.g., Class 4 of ISO 8073) or on emerging optimized protocols (e.g., VMTP, XTP) and will be capable of utilizing other multicast Network Layer protocols.

**Task 7: Develop a Multipeer Data Transmission Addendum for the OSI Reference Model**

This task develops the necessary changes to the OSI Reference Model to support multicast applications (ISO 7498 Add. 1). The method of change will be to develop an addendum to the Model that incorporates the architectural requirements for multicast. The contractor will utilize the work previously submitted to ISO committees in developing the multipeer addendum.

**Task 8: Develop a Multicast Connection-Oriented Network Service Definition**

This task develops a Recommendation for an X.25 multicast service called X.pms (Packet Multicast Service) or X.6. (Note that this is only one form of connection-oriented multicast.)

**Task 9: Develop a Multicast Connection-Oriented Network Protocol Specification**

This task will be based on the OSI Connection-mode Network Service (ISO 8878) and CCITT Recommendation X.25.

**Task 10: Develop a Server Description**

This task develops a common set of operations, definitions, responses, and protocol elements which are needed for a multicast server. This description will apply to a server whether it is implicit in the transmission medium (such as in many LANs), internal to the network (as proposed in a layer 3/4 multicast service), or external to the network (as a multicast service at higher layers might use).

**Task 11: Develop an Inter-Server Protocol**

This task develops an inter-server protocol which does not involve additional inter-network protocol elements, but is placed on top of existing protocols. In the situation where a server is internal to a network providing a multicast service, cross-network multicast needs additional protocol and architectural support.

**Task 12: Develop a Connection-Oriented/Connectionless Multicast Interworking Description**

This task shall create a description of interworking between Connection-Oriented and Connectionless multicast environments.

**Task 13: Network Vendor Interface**

This task will provide for communications with various network vendors, enlisting their support in this program through protocol implementation, assistance in design, and marketing of the service to the DoD.

The status of these tasks will be covered in detail in the Multicast Final Report.

## 6. TRANSITION PLAN

The evolution of the DIS communication architecture will need to occur in phases to allow sufficient time for users and implementors to gain experience with current requirements before new services are introduced. This strategy will be accomplished with a Transition Plan which synchronizes interim and long-range services with the phases of the architecture. The Transition Plan should not only identify the services and base stacks for each phase, but should also postulate the time frame in which the transitions should occur. The approach to defining a Transition Plan consists of three steps: 1) establish goals for the communication architecture; 2) select a COTS protocol suite as a starting point; and 3) based on the Applications and standards evolution, establish a time frame for the transition of the architecture to long-range goals.

In previous sections we defined the Applications; determined the service requirements; identified the communication protocols which satisfy each requirement; examined COTS standards for use by the Applications; and, if COTS products did not exist, classified the requirement as long-range. This section attempts to coalesce this information into a Transition Plan which, based on the starting point recommended by CASS, defines a strategy to evolve the communication architecture from interim to long-range services through research which emphasizes the coexistence of standards.

The goals of the Transition Plan are:

- To identify research necessary to satisfy long-range requirements,
- To avoid reimplementations at each phase,
- To reduce risk and cost.

### 6.1 Areas of Development

For the architecture to evolve gracefully, research on long-range requirements must continue. This work should proceed with the continuation of the multicast standards work started in Year One as well as starting work on the DIS Application structure and the multicast group management protocol. It is important that this work be flexible enough to begin prototyping on the interim architecture, accelerating the transition to subsequent phases.

#### 6.1.1 Upper Layers

##### Multipeer Architecture

The further into the development of a multipeer addendum to the OSI Reference Model we get, the more the similarity between the multicast and the traditional unicast communication becomes. In many places the language needs to be generalized, but in most cases it is simple as saying "among" and not "between" and not referring to "two" or "pairs". The fundamental concepts



not really different. In multicast some aspects do take on more importance, e.g. enrollment. Because this similarity has been suspected, it has been surprising just how similar unicast and multicast are and how few special cases are required. This should lead to a much simpler architecture and much simpler protocol designs and implementations to accommodate multicast.

One area that must be developed and clearly understood is the broad range of characteristics of multicast communication that have a profound affect on the variety of functions found in protocols. Early discussions showed that virtually all combinations of these characteristics could have some economically viable applicability. This raises the specter either of complex (and inefficient) protocols that accommodate many characteristics or of combinatorial (many multipeer) protocols applicable to very specific (small) markets. As briefly described below, it appears that an approach has been found that makes it possible to avoid both of these extremes which may be able to broaden the applicability, i.e. market, of the specific multicast protocols that are developed.

### 5 Application Structure

The application structure for DIS Applications in a multipeer<sup>11</sup> environment will need to be developed. The application interface will be based on the OSI Extended Application Layer structure (XALS) [13] and will allow real-time simulation applications to select Transport services through extensions to the Association Control Service Element (ACSE). The interface will be designed with maximum flexibility for growth of the DIS Application Layer structure. The DIS application layer will be required to support a variety of simulated and real entities. To make efficient use of the lower-level services, a mechanism which permits these entities to be multiplexed on (or share) lower-level connections (i.e., the Communication Classes) must be developed. Maximum commonality will be promoted by using XALS to partition DIS application entities into generic and specific Application Service Elements (ASEs) and Application Service Objects (ASOs). DIS will make minimum use of Presentation services and will not use any Session functional units. This will allow the upper layer architecture implementation to be very compact, along the lines of the Skinny Stack [20]; thereby minimizing processing overhead.

### Group Management

The foremost application layer capability that can be currently identified is a group management protocol. DIS will need a group management protocol to provide group membership management, group initiation, and group communication termination. It is preferable that this protocol be "generic" and therefore standardized for all distributed simulation applications; however, it is too early to tell if this can be accomplished. Because this protocol will reside in

---

<sup>11</sup>In this context, Multippeer includes multicast, broadcast, and unicast transmissions.

both simulators and CIUs/CAUs, it will be important to consider resiliency techniques to support reliability between group managers in the CIUs/CAUs.

### 6.1.2 Lower Layers

#### Transport Layer Multicast

The architecture of the reliable multicast protocol, TP5, is based on the existing OSI Transport Layer protocol TP4. The goal is to develop TP5 along the lines of the existing OSI architecture. This idea has been favorably accepted by the ANSI X3S3.3 task group. TP5 was proposed in February 1992 at the Tucson meeting and, after only 4 months, had its service definition forwarded to the July ISO SC 6 meeting as an official U.S. position. The TP5 protocol specification was taken to the ISO SC 6 meeting as an "Expert Paper".

For the coming year work needs to continue on the definition of TP5. The current draft of TP5 lays a foundation for developing a multipeer transport protocol that is capable of providing a wide range of services. However, significant development work is still needed to: map service types to protocol mechanisms, develop semantics for reliable multipeer, develop mechanisms for flow control and sequencing, and to fit the mechanisms into the overall TP5 infrastructure.

TP5 is now defined with a common infrastructure (header format) that allows for the definition of protocol mechanism tailored for specific purposes. That is, when a "connectionless multicast" protocol is needed, TP5 without reliability mechanisms is used. As more robust services are needed, mechanism within the protocol may be called that are internally consistent with the connectionless operation. By employing the add-in approach a single protocol can be used to provide the wide-ranging services needed in a multipeer environment. Future work will be geared to defining the set of add-in mechanisms that provide the different classes of service required by DIS and other multipeer services.

#### Network Layer Multicast

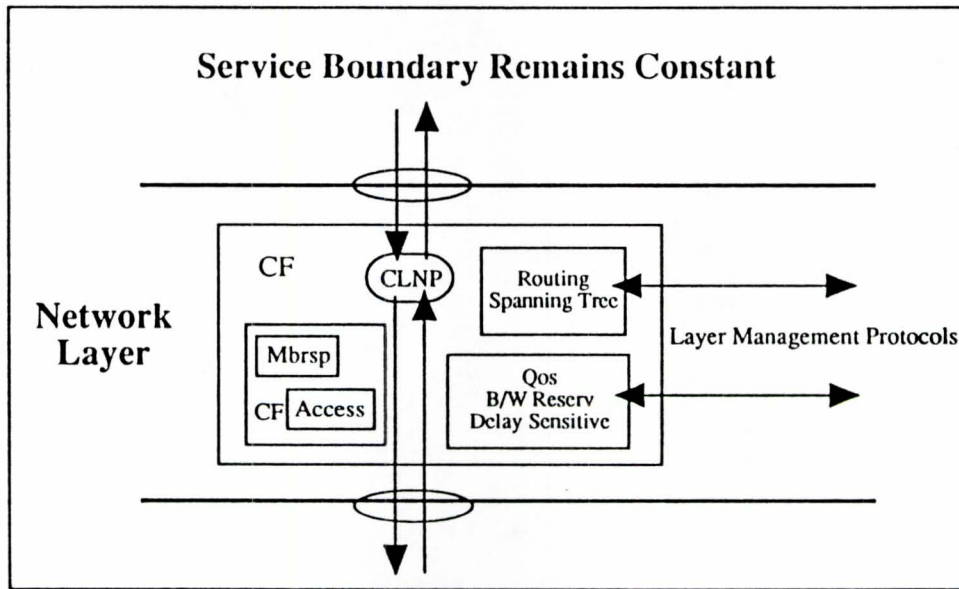
There are a wide range of applications requiring Network layer multicast. Each application will likely require a different type of service. Examples of multicast characteristics include static vs. dynamic group membership, centralized vs. distributed transmission, and sender-directed vs. receiver-directed scope control. It seems undesirable to choose one form of multicast over another; yet attempting to implement all services in one protocol will be complex and inefficient. Further, one protocol for each combination of characteristics is no more desirable. Therefore, the goal for Network layer multicast will be to support the widest possible range of services with the lowest complexity and cost.

At the April meeting of X3S3.3, a proposal for solving Network layer multicast was presented [8]. The approach is based on a modular architecture which allows multiple types of multicast



service to coexist. The result are modules which can be combined in a variety of ways to create a wide range of multicast services. These modules are called Layer Functional Modules (LFMs) and are based on the XALS concepts originally developed for the upper layers. A concept paper on LFMs was forwarded to the July ISO SC 6 meeting as an official U.S. position and was favorably received. The LFM approach is gaining international acceptance and shows technical promise for solving the multicast architecture in the Network layer.

The LFMs can be used as "building blocks" to create any form of multicast service required. Careful design can lead to a small set of LFMs that can provide a wide variety of multicast services without requiring a separate protocol for every combination of characteristics. According to [8], there are at least four general functions required to provide multicast operation in the Network layer: group management (e.g., membership and access control), addressing, routing (e.g., initialization and maintenance), and data forwarding. There are also generic LFMs which can be used for multicast or unicast transmissions, e.g., quality of service. Each of these functions will be an independent LFM and will likely contain one or more additional LFMs to perform specific operations. Figure 3 (taken from [8]) illustrates how this architecture can mix and match Network layer mechanisms without disturbing the Transport layer interface.



0407-2937

Figure 3. Decomposing the Network Layer into Layer Functional Modules



It has been proposed in [14] that all services, including multicast, progress through the same five phases of operation during the lifetime of a service. The phases may be broadly defined as follows: *Enrollment*, creates everything necessary to enter into communication; *Allocation*, when the participant requests the allocation of communication resources such that it is ready for data transfer; *Data Transfer*, when the actual transfer of data is effected according to the requested quality of service among addresses specified during either of the previous two phases; *De-Allocation*, when the participant has completed the data transfer phase and de-allocates any shared state established during the Allocation phase; and *De-Enrollment*, when the communication is unavailable or when the registration is terminated;

Using these phases, we now have a structure for organizing and defining the types of LFMs required by DIS. The general multicast functions defined above can be aligned with the phases as follows:

**Enrollment/De-Enrollment**

Group Membership: definition and activation  
Access Control

**Allocation/De-Allocation**

Resource Allocation: route initialization and maintenance  
Group Population: static/dynamic/unknown

**Data Transfer**

Data Forwarding: CLNP  
Multicast Distribution: centralized (exploder techniques) or distributed (spanning tree techniques)  
Flow/Error Control  
Resource Management: bandwidth reservation  
Sequencing: pairwise or global

DIS will need LFMs from each of the phases defined above; however, the nature of the LFMs has not been determined. Now is the time to begin addressing these issues and defining the functions required by DIS.

The LFM approach is fundamentally pluralistic and side steps the emotional issue of "my multicast" or "your multicast" by letting everyone have "their multicast". Any technique can be cast easily and simply into this structure. Not only does this approach provide a powerful tool for organizing the multicast standards development, it also provides a powerful structure for experimenting with multicast techniques. One benefit of this architecture is its "protocol independent" nature. While the basic multicast data forwarding will be provided by CLNP, interim DIS prototyping could initially build experimental LFMs around IP. Over time, the experimental LFMs can be replaced with standards. This would allow a phased evolution of multicast for prototyping, experimentation, and standardization. Prototyping the LFMs and experimenting with LFMs individually in the interim architecture avoids a complete

reimplementation for the OSI/GOSIP architecture. The prototypes could be deployed as early as 12-18 months, especially if the interim architecture utilizes existing technology, i.e. IPMC and ST-II. In this case, an immediate first step is the decomposition of IPMC and ST-II into LFM's, and aligning the functions with the phases described above. These LFM's will serve as the basis for prototyping, experimenting, and standardization.

## 6.2 Three Phases

The CASS strategy for OSI/GOSIP compliance is based on a phased, evolutionary approach. The first step to this evolution is the recommendation of an interim architecture, based on available network products and services, which is capable of supporting current exercises and communications experiments. The interim architecture will then transition to OSI/GOSIP standards over a period of years, as multicast communications protocol standards are adopted to support DIS.

To realize a graceful evolution of the architecture, the CASS is recommending a three phase approach. Phase 0, also known as the interim architecture, mandates COTS products with no development or customization. The protocols which make up this phase are basic communication services and will provide an infrastructure for proof-of-concept communication experiments. Due to the maturity of IPS products, Phase 0 will use Internet standards. Since the multicast requirement cannot be satisfied with a COTS product, CASS is leaving the choice "open" rather than requiring a particular protocol. The Phase 0 architecture is a proof-of-concept of the DIS OSI communication infrastructure. In addition to the basic communication protocols identified in the Phase 0 architecture, Phase 1 requires additional multicast capabilities. The last phase of the architecture, Phase 2, is the GOSIP compliant architecture, based upon lessons learned in Phase 1, added functionality, and final versions of OSI/GOSIP multicast protocols. The phases and corresponding base stacks are shown in Figure 4 and defined in the following sections.

SNMP	Telenet	FTP	NTP
UDP		TCP	
IP	IPMC w/LFMS	Open	
IEEE 802.3			

0407-2935

Phase 0:  
Internet Architecture

DIS-AP	CMIP	VTP	FTAM	DIS-GMP
CLTP		TP4		
CLNP	MC-CLNP	LFMs		
IEEE 802.3		FDDI		

Phase 1:  
OSI Architecture

DIS-AP	CMIP	VTP	FTAM	DIS-GMP
CLTP	TP4	TP5		
CLNP	MC-CLNP	LFMs		
IEEE 802.3	FDDI	Future		

Phase 2:  
GOSIP Architecture

Figure 4. Three Phase Communication Architecture



### 6.2.1 Phase 0: Internet Architecture

The *base stack* for the Phase 0 architecture consists of the basic communication services described in Section 4.2. For consistency with subsequent phases of the architecture, Phase 0 will be described in terms of the OSI seven layer model. At the Application layer, five protocols are specified: DIS Application Protocol, Simple Network Management Protocol (SNMP), Telnet (a terminal protocol), File Transfer Protocol (FTP), and Network Time Protocol (NTP). SNMP and Telnet will be used to meet the Network Management service requirement; SNMP will provide network monitoring while Telnet will be used to establish terminal sessions for remote debugging and network management. FTP will be used to satisfy the file transfer requirement by providing a bulk transfer service (i.e., retrieval of databases). The NTP will be used to meet the synchronization requirement.

At the Transport Layer, the architecture is based on the User Datagram Protocol (UDP) for unreliable unicast (or datagram) service and the Transmission Control Protocol (TCP) for reliable data transfer. At the Network Layer, the architecture specifies the Internet Protocol (IP) for seamless local/global communication. Aligning the interim architecture with the Communication Classes approach described in Section 3.1.3, Table 10 characterizes the interim services.

	Class 1: Unreliable Multicast	Class 2: Unreliable Unicast	Class 3: Reliable Unicast
Application	DIS PDUs	DIS PDUs SNMP NTP	DIS PDUs FTP Telnet
Transport	UDP	UDP	TCP
Network	IP ST-II LFMs	IP	IP

Table 10  
Phase 0 Service Characterization

As part of the standard operation of IP-over-Ethernet, the mapping between IP-addresses and the corresponding local Ethernet addresses, is handled by the Address Resolution Protocol (ARP). The routing protocols used with this protocol suite are Exterior Gateway Protocol (EGP), Border Gateway Protocol (BGP) and Open Shortest Path First (OSPF).

As discussed in Section 3.2, the DIS requirement for multicast can not be met with any COTS product and a specific multicast protocol has not been recommended by CASS. There are two candidates which can meet interim requirements, ST-II or XTP. ST-II is the better choice because it provides the necessary resource reservation required for real-time delivery. It has also



been proven to work for DIS applications and provides a better migration to the LFM's being proposed in ANSI/ISO. Prototyping the LFM's should begin in this Phase by decomposing the Internet Group Multicast Protocol (IGMP) and ST-II, using IP as the basic data forwarding service. All experience gained with LFM's at this stage of development will be directly transferable to subsequent phases of the architecture. This is not true for experience gained with XTP. At this time CASS is leaving the selection of the multicast protocol to the individuals requiring it for an exercise; however, this Transition Plan recommends the use of ST-II.

The Phase 0 architecture is based on WAN's interconnecting Ethernet (IEEE 802.3) LAN's with a local broadcast capability at each site.

The standards<sup>12</sup> for Phase 0 are listed below:

<b>SNMP</b>	Simple Network Management Protocol (RFC 1157)
<b>Telnet</b>	(Terminal Protocol) (RFC 854)
<b>FTP</b>	File Transfer Protocol (RFC 959)
<b>NTP</b>	Network Time Protocol (RFC 1119)
<b>UDP</b>	User Datagram Protocol (RFC 768)
<b>TCP</b>	Transmission Control Protocol (RFC 793)
<b>IP</b>	Internet Protocol (RFC 791)
<b>ICMP</b>	Internet Control Message Protocol (RFC 792)
<b>ST-II</b>	Stream-II (RFC 1190)
<b>IGMP</b>	Internet Group Multicast Protocol (RFC 1112)
<b>LFM's</b>	Layer Functional Modules - Group Management, Routing, Data Forwarding, Addressing, and Quality of Service
<b>ARP</b>	Address Resolution Protocol (RFC 826)
<b>RARP</b>	A Reverse Address Resolution Protocol (RFC 903)
<b>EGP</b>	Exterior Gateway Protocol (RFC 904)
<b>BGP</b>	Border Gateway Protocol (RFC 1163, 1164)
<b>OSPF</b>	Open Shortest Path First Routing (RFC 1131)
<b>CSMA/CD</b>	Carrier Sense Multiple Access with Collision Detection (IEEE 802.3)

### 6.2.2 Phase 1: OSI Architecture

The proposed Phase 1 architecture incorporates the OSI seven layer stack to facilitate the migration to a full GOSIP compliant network. The *base stack* for Phase 1 combines the OSI basic communication services with the required multicast services. The Application Layer specifies five protocols: DIS Application Protocol (DIS-AP), Group Management Protocol (DIS-GMP), Common Management Information Protocol (CMIP), Virtual Terminal Protocol (VTP), and File Transfer Access and Management (FTAM). The DIS-AP will allow PDUs to select the

---

<sup>12</sup> The Internet standards are called Request For Comments (RFCs).

required Transport services (i.e., the appropriate communication class). The DIS-GMP will provide the capability to specify group membership management, group initiation, and group communication termination. To satisfy the network management requirement, CMIP and VTP will be used. CMIP provides network management and monitoring and VTP will be used where ever terminal sessions are needed. FTAM will be used to satisfy the file transfer requirement by providing a bulk transfer service (i.e., retrieval of databases). The synchronization requirement is being developed within the OSI program of work [21].

The Session and Presentation Layers and ACSE will be implemented using the Skinny Stack [20] approach and will incorporate the ASO-ACSE (A<sup>2</sup>CSE) extensions.

At the Transport Layer, the architecture is based on the Connectionless Transport Protocol (CLTP) for datagram service and the Class 4 Transport Protocol (TP4) for reliable data transfer. At the Network Layer, the architecture specifies the Connectionless Network Protocol (CLNP) for seamless local/global communication. The characterization of Phase 1 protocols is shown in Table 11.

	Class 1: Unreliable Multicast	Class 2: Unreliable Unicast	Class 3: Reliable Unicast
Application	DIS-AP DIS-GMP	DIS-AP CMIP	DIS-AP CMIP FTAM VTP
Transport	CLTP	CLTP	TP4
Network	CLNP LFMs	CLNP	CLNP

Table 11  
Phase 1 Service Characterization

Routing will be based upon End System to Intermediate System (ES-IS) protocol, Intermediate System to Intermediate System (IS-IS) protocol, and Inter Domain Routing Protocol (IDRP). However, in the early implementations, static routing will be used. The ES-IS protocol provides the equivalent function as the Internet ARP protocol.

DIS will need a long-lived Network Layer multicast protocol; LFMs will be used in Phase 1 to satisfy the multicast requirement. The basic multicast data transfer will be provided by multicast CLNP (MC-CLNP), i.e. CLNP carrying group addresses. While the DIS-specific functions, such as resource reservation and multicast distribution, will be provided by individual LFMs.



Since LFM's were prototyped in Phase 0, the transition to Phase 1 will be much cleaner and will not require total reimplementatation. The transition will require only the substitution of CLNP for IP for carrying data.

The architecture will successfully operate over any type of communication subnetwork environment that meets minimum performance requirements (e.g., IEEE 802.3).

The OSI protocols are defined below:

<b>DIS-AP</b>	DIS Application Protocol (undefined)
<b>DIS-GMP</b>	DIS Group Management Protocol (undefined)
<b>A<sup>2</sup>CSE</b>	ASO-Association Control Service Element (undefined)
<b>XALS</b>	Extended Application Layer Structure (ISO 9545/DAM1)
<b>CMIP</b>	Common Management Information Protocol (ISO 9596)
<b>VTP</b>	Virtual Terminal Protocol (ISO 9041)
<b>FTAM</b>	File Transfer Access and Management (ISO 8571)
<b>CLTP</b>	Connectionless Transport Protocol (ISO 8602)
<b>TP4</b>	Transport Protocol Class 4 (ISO 8073)
<b>CLNP</b>	Connectionless Network Protocol (ISO 8473)
<b>LFMs</b>	Layer Functional Modules - Group Management, Routing, Data Forwarding, Addressing, and Quality of Service (undefined)
<b>ES-IS</b>	End System to Intermediate System Routing Protocol (ISO 9542)
<b>IS-IS</b>	Intermediate System to Intermediate System Routing Protocol (ISO 10589)
<b>IDRP</b>	Inter Domain Routing Protocol (ISO 10747)
<b>CSMA/CD</b>	Carrier Sense Multiple Access with Collision Detection (ISO 8802/3)
<b>FDDI</b>	Fiber Distributed Data Interface (ISO 9314)

### 6.2.3 Phase 2: GOSIP Architecture

The proposed Phase 2 architecture incorporates the future OSI multicast protocols into the GOSIP compliant network. The additional protocols required in Phase 2 will be the reliable multicast protocol. Although DIS does not currently require a reliable multicast protocol, it is desired for the long-term architecture. This reliability will likely be provided by a Transport Layer protocol called TP5 [9]. DIS may also want apply multicast to the network management protocol, CMIP. When DIS simulation management is defined, it will be possible to anticipate a desired use.

The Phase 2 service characterization is shown in Table 12.



	Class 1: Unreliable Multicast	Class 2: Unreliable Unicast	Class 3: Reliable Unicast	Class 4: Reliable Multicast
Application	DIS-AP DIS-GMP	DIS-AP CMIP	DIS-AP CMIP FTAM VTP	DIS-AP
Transport	CLTP	CLTP	TP4	TP5
Network	CLNP LFMs	CLNP	CLNP	CLNP LFMs

Table 12  
Phase 2 Service Characterization

The additional protocol required for Phase 2 is shown below:

**TP5**            Transport Protocol Class 5 (undefined)

### 6.3 Optimizing the Transition

The approach described in the preceding sections allows us to reduce risk and cost and, perhaps, optimize the transition from interim to long-range requirements. Since this strategy promotes the coexistence of Internet and OSI standards, it is flexible enough to allow prototyping on the interim architecture before transitioning to subsequent phases of the architecture. Because this approach does not require complete re-implementation at each phase or any wholesale changes from one stack to another, the transition can constantly balance the rate of change with risk and cost. Sometimes speeding up the transition without increasing risk. Sometimes slowing down to lower risk on aspects of the transition when tough technical problems are encountered without impacting schedule severely.

The architecture and design of LFMs is not straightforward or obvious. "Getting it right" the first time is not a given. It is very easy to do a bad decomposition. Thus, a fair amount of design effort should be allowed to minimize the risk to this aspect of the work.

Once the decomposition of LFMs is determined and the basic LFMs are identified, i.e. those that are required by everyone or nearly everyone, you can concentrate on only those LFMs you want. For example, the basic multicast data forwarding LFM will be useful to many different types of applications. However, DIS will require specific capabilities that other applications do not need, e.g. resource reservation. The LFM approach allows us to concentrate the majority of our efforts and resources on the unique LFMs, interesting other industry segments in assisting with

the development of general-purpose capabilities. The LFM's also allow us to experiment with individual components of multicast, and thereby provide DIS with a means to fine-tune our multicast architecture easily and cost effectively.

To facilitate the prototyping of long-range protocols for Phases 1 and 2, the public domain OSI Skinny Stack implementation should be used. The Skinny Stack is based on an implementation strategy implied by the standards that combines the protocol machines of the upper three layers (Session, Presentation, and ACSE) into a single state machine rather than the naive approach of implementing a protocol machine for each layer. The resulting Skinny Stack that uses no Session Functional Units (as with DIS) is 1800 lines of C and considerably faster than other implementations. An early Skinny Stack implementation for mapping X windows over OSI shows comparable performance to X-windows over TCP. The X-windows over OSI was only 9% slower than X-windows over TCP and 3 times faster than an ISODE implementation. Skinny Stack is an OSI conformant upper layer profile/stack which provides only those facilities needed to support send-receive applications. The Skinny Stack is interoperable with other Skinny Stacks as well as with full (i.e., all the facilities of Session Layer, Presentation Layer and ACSE) OSI stacks.

#### 6.4 A Timeline

To make the Transition Plan a viable strategy, it is important to develop a timeline which identifies the transitions from one phase to the next. But how do we define these time periods? The approach used in this Plan is based on the Applications using DIS and the projected standards evolution required to satisfy long-range requirements.

As discussed in Section 2.1, there are seven new procurements or upgrades to existing systems which refer to the DIS standard: CCTT, BFTT, MAIS, TCTS, TACTS, JACTS, and NTC. Most of these projects are in initial planning stages and do not have critical design reviews until the 1994-1995 time frame, much less actual deployment. Until that time, exercises will consist mostly of homogeneous simulators or small groups of heterogeneous simulations (e.g., the I/ITSC demonstration). This will be the period when DIS is demonstrated and tested, as well as developed (e.g., versions 2.0 and 3.0 of the standard). During the next 3-4 years, DIS will begin to mature much the way traditional communication protocols do. Therefore, we can postulate that this is our interim time frame.

DIS will have to wait three to four years to get the full range of Network and Transport multicast services it requires. Since the goal of the DIS communication architecture is to move toward OSI/GOSIP, this is where the development work should focus. As discussed previously, the multicast work is being designed to be relatively protocol suite independent. This will allow DIS to begin experimentation with multicast in the interim architecture without losing the experience gained when the architecture evolves to GOSIP compliance. The LFM approach will allow DIS to move quickly from IP to CLNP and will not put DIS in a bind if the ISO standards process gets bogged down. If this strategy is sidetracked by developing unique interim solutions, it will



be at great expense to the Government and to the DIS community and will prolong the lack of marketability for the services being developed.

Since the Phase 0 (IPS) architecture has a very large experience base in industry, it will require minimal effort to learn and develop. However, the DIS community should not take this to mean that all their communication problems are solved. To the contrary, as discussed in Section 4.1.1, the IPS is only an interim solution itself since the internetworking protocol IP will have to migrate to a successor in the same interim time frame. The current disarray in the Internet community over the successor to IP makes the stable CLNP and associated protocols a much more attractive target to migrate to.

In the coming years, any new development and experience should be with the OSI protocol suite. The DIS community will have to gain OSI experience sometime, and the sooner the community begins, the sooner we can arrive at our long-term architecture. The OSI protocols are still maturing and can be molded to meet the requirements of DIS. But they won't magically get there on their own. DIS has specific multicast requirements that no other applications have, which makes commercial interest difficult. DIS should take steps to broaden the appeal of the services required by wargaming to other simulation industries (e.g., entertainment) or else commercial interest in simulation-specific services will always be difficult<sup>13</sup>. In any case, DIS must take the initiative to mature these protocols and start now. It will take a minimum of three years to develop multicast and, according to the Applications time frame, this is exactly the amount of time DIS has.

By synthesizing our knowledge of the Applications and standards evolution, it is possible to develop a timeline for the evolution of the DIS communication architecture. This timeline is shown in Figure 5. The line starts with the beginning of this contract in 1991 and ends in 1997 with the projected dates of the BFTT and TCTS contracts. Shown on the line are the projected dates of the known Applications which will use DIS along with the corresponding maximum number of entities expected for each program. Also shown is the recommendation of the Phase 0 architecture (3/92) and projected dates for LFM prototypes (9/93), the Phase 1 architecture (3/95), and the Phase 2 architecture (3/96). A prediction was made at the March 1992 CASS meeting for when DSI would support OSI traffic. This date (3/95) is also included on the line and strengthens the postulated interim time frame. The last group of dates included on the timeline are those of projected exercises (e.g., LA Maneuvers) which will potentially use DIS.

---

<sup>13</sup> However, identifying and characterizing LFMs in the lower layer and generic Application Service Objects in the application layer can do much to broaden the market and lower the cost to the DoD.



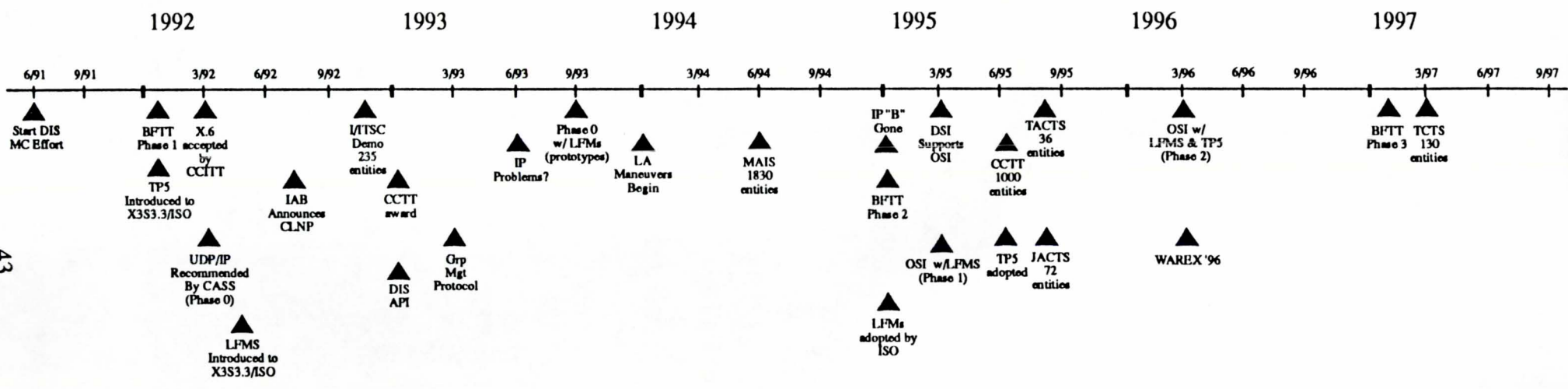


Figure 5. Timeline for DIS Applications and Standards Evolution

## 7. CONCLUSIONS

Today, there is no one architecture (Internet or OSI) which meets the total needs of DIS. This means that the DIS community will have to develop an architecture to meet its requirements. This development will need to occur in phases to allow sufficient time for users and implementors to gain experience with current requirements before introducing new services. The phases CASS is recommending provides a good structure for the DIS transition to GOSIP compliance. The Phases provide DIS with a starting place (Phase 0), an interim OSI stack (Phase 1), and a final OSI/GOSIP stack (Phase 2). The basis for starting with the IPS base stack is strengthened by the fact that the DSI currently supports only IP and ST-II traffic. However, DSI will begin routing OSI traffic in the next three years (again, our postulated interim time frame). The DIS users must take advantage of this time to gain OSI experience and begin evolving the architecture before it is required for interoperability.

The transition to OSI/GOSIP will not happen overnight. If DIS desires GOSIP compliance, the community must take the initiative to push the evolution of the protocols which meet our requirements. Until the time that all services are met by one protocol suite, the communication standards (IPS and OSI) will have to coexist. This means that while development of services and protocols are occurring on the long-term architecture, proof-of-concept testing should occur on the interim architecture. The multicast LFMs are a perfect example. This approach will reduce the risk of integrating new services and provide a graceful evolution of the architecture DIS requires.

The approach presented here is flexible enough to accommodate existing and future communications technologies, while providing a graceful growth path for the maturation of the architecture. Now is the time for the DIS community to think about the range of problems it faces and formulate a strategy for achieving the goals within the required time frame. We should not blindly react to interim solutions which, at best, patch the problem and requires re-development at a subsequent phase of the architecture. By following this strategy, DIS can focus attention on the long-range answers we need.

## GEMENT

elf in an interesting situation. My task was  
on architecture make the transition to an OSI  
oversial this would become. After my first  
est approach was solicit advice from Internet  
to be involved in communication standards  
ight be expected, each person had a different  
uld not drive the Internet versus OSI debate,  
now believe this to be the best approach for  
s report emphasizes the idea of coexistence.  
one matter: development of new protocols  
Therefore, if DIS is to eventually be OSI  
be.

rom various sources. I hope it synthesizes  
e learned through mail lists (IETF, X3S33,  
meetings (X3S3.3, X3S3.7, X3T5.1, and

ery fortunate to meet a group of people who  
ons and I would like to thank them for their  
al thanks to John Day, BBN Communications  
press my appreciation to Jim Moulton, Open  
is and Technologies; Lyman Chapin, BBN  
Corporation; Al Kerecman, USA CECOM;  
n Hares, Merit Network; Bob Cooney, Naval  
ichard desJardins, The GOSIP Institute.



## REFERENCES

- [1] Military Standard (Draft), Protocol Data Units for Entity Information and Entity Interaction in a Distributed Interactive Simulation, IST-PD-90-2 (revised), September 20, 1991.
- [2] Communication Architecture for DIS, Draft Standard, March 1992.
- [3] Summary Report, The Sixth Workshop on Standards for the Interoperability of Defense Simulations, Volume 1: Minutes and List of Attendees, March 17-19, 1992. Institute for Simulation and Training, IST-CR-92-2.
- [4] Strawman Distributed Interactive Simulation Architecture Description Document, Volume I: Summary Description, Loral Systems Company, ADST Program Office, March 31, 1992.
- [5] Communication Service Requirements for Distributed Interactive Simulation, Part 1: Application Service Characterization, M. Loper, IST-TR-91-27, October 30, 1991.
- [6] Communication Service Requirements for Distributed Interactive Simulation, Part 1: Multicast Service Characterization, M. Loper, IST-TR-92-23, July 20, 1992.
- [7] The Department of Defense Open Systems Interconnection (OSI) Implementation Strategy, DCA/DCEC, May 1988.
- [8] Layer Management Enhancements for Network-Layer Multicast: Architecture, D. Deutsch and J. Day, March 27 1992, X3S3.3/92-128.
- [9] Proposed Contribution to SC 6 on Multicast Transport Protocol, J. Moulton, March 26 1992, X3S3.3/92-142.
- [10] "Internet Architect Gives Long-Term View", Network World, May 18, 1992.
- [11] Communication Service Requirements for Aggregate Level Simulation Protocol (ALSP), MITRE Informal Report, February 12, 1992.
- [12] Distributed Interactive Simulation (DIS), Concept of Operation, Technology, System Design and Architecture, Dr. Ron Hofer, U.S. Army Simulation Training and Instrumentation Command.
- [13] Information Processing Systems - Open Systems Interconnection - Application Layer Structure, ISO 9545/DAM1.

- [14] Considerations for Multicast Architecture in the Lower Layers and Its Implications, J. Day, March 26 1992, X3S3.3/92-210.
- [15] Evaluation of Abstract Syntax Notation One for Distributed Interactive Simulation, Part 2: The Single-Host and Network Models, M. Bassiouni, G. Bulumulle, M. Chiu, M. Loper, and H. Nee, IST-TR-92-22, June 30, 1992.
- [16] Specification of Distributed Interactive Simulation in the Estelle Formal Description Technique, D. Shen and H. Ng, IST-TR-92-17, April 23, 1992.
- [17] International Organization for Standardization - Information Processing - Open System Interconnection - Specification of Abstract Syntax Notation One (ASN.1), ISO 8824, 1987.
- [18] International Organization for Standardization - Information Processing - Open System Interconnection - Specification of Basic Encoding Rules (BER) for ASN.1, ISO 8825, 1987.
- [19] International Organization for Standardization - Information Processing - Open Systems Interconnection - Estelle: A Formal Description Technique Based on an Extended State Transition Model, ISO 9047, 1989.
- [20] The OSI Skinny Stack, P. Furniss and K. Ashley, University of London Computing Centre, March 25, 1992.
- [21] Time Management Function Working Document, ISO/IEC JTC 1/SC 21 N 7125, July 3, 1992.



0000123