1-1-1992

Communication Architecture For Distributed Interactive Simulation (CADIS): Rationale Document Draft

University of Central Florida Institute for Simulation and Training

INSTITUTE FOR SIMULATION AND TRAINING

[redacted]

SEPTEMBER 4, 1992

RATIONALE DOCUMENT (DRAFT)

COMMUNICATION ARCHITECTURE FOR

DISTRIBUTED INTERACTIVE SIMULATION (CADIS)

IST-CR-92-7

[redacted]

IST

INSTITUTE FOR SIMULATION AND TRAINING

Contract Number N61339-91-C-0091
STRICOM
DMSO

September 4, 1992

Rationale Document Draft

Communication Architecture for Distributed Interactive Simulation (CADIS)

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

University of Central Florida
Division of Sponsored Research

IST-CR-92-7

RATIONALE DOCUMENT

COMMUNICATION ARCHITECTURE

FOR

DISTRIBUTED INTERACTIVE SIMULATION

[CADIS]

RATIONALE DOCUMENT

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Figures

1. INTRODUCTION

1.1 Forward

The purpose of the communication subsystem for Distributed Interactive Simulation (DIS) is to provide an appropriate interconnected environment for effective integration of locally and globally distributed simulation entities. There are many diverse aspects of this integration, ranging from the nature of the entities represented within the common simulated environment, to the common communication interface used for receiving packets of information from other simulators. The standard addressed by this Rationale Document is concerned only with the necessary communication system standards which must be accepted and adopted for supporting the integrated framework.

The Protocol Data Units (PDUs) defined in the DIS Standard are the "lingua franca" by which any two simulators or simulation sites can communicate. This includes simulators of different and unrelated design and architecture. No restriction is placed on what the participating simulator or site is, only on the way it communicates with the outside world.

Where the DIS PDUs define the information passed between simulators and simulation sites, this standard will define how those simulators, simulation sites, and other DIS entities can be connected in a modular fashion to facilitate the communication at the local and global levels. This will be done through the required use of communications standards which promote interoperability, such as the International Organization for Standardization (ISO) Open Systems Interconnection (OSI) reference model and the Government OSI Profile (GOSIP).

This standard describes the communication architecture subsystem that will support DIS exercises and activities. The DIS PDU standard describes the format of the application protocol data units that contain the entity, environment, and simulation management information that will be carried on the network. This standard describes the structure and use of the network to carry that information. This document describes the rational behind the requirements and specifications in the communication architecture standard. The guidance document (TBD) will describe how to use the information in creating a communication subsystem to support DIS activity.

1.1.1 Background

The current work on standards began in August 1989 with the First Workshop on Standards for the Interoperability of Defense Simulations. Using the work of SIMNET as a baseline and considering recommendations made in workshop meetings and position papers, IST developed a first draft for a military

standard which describes the form and types of messages to be exchanged between simulated entities in a Distributed Interactive Simulation. The workshops also provided for discussion in other areas associated with DIS such as environment, fidelity and exercise control and feedback, and communication architecture and security. Through the meetings of the workshops, based on discussions and individual input, the first draft of the COMMUNICATION ARCHITECTURE FOR DISTRIBUTED INTERACTIVE SIMULATION (CADIS) military standard has been developed. This rationale document addresses this first draft of the communication architecture/security standard.

1.2 Scope

This document contains extensive rationale supporting the choice of key items that have become part of the draft military standard entitled COMMUNICATION ARCHITECTURE FOR DISTRIBUTED INTERACTIVE SIMULATION . This rationale is intended to give the system designer a better understanding of why some choices were made and what impact deviation from them might have on the communication architecture system being designed. The communication architecture defined in the above mentioned draft military standard encompasses layers 1 through 5 of International Organization for Standardization's (ISO) Open Systems Interconnection (OSI) Reference Model (ISORM).

1.2.1 Intended use

The intended use for this rationale document is as follows:

a. To define the service and performance requirements of a communication architecture to support DIS applications.

b. To recommend standard, non-proprietary protocols to be used in the communication architecture which will support the above requirements.

c. To recommend interim protocols to be used in the communication architecture for those requirements that cannot be met by existing standardized protocols.

d. To present issues (interoperability, security, management) that are related to the communication architecture specified for DIS applications as they appear in position papers and working group recommendations.

1.2.3 Future Goals

The standard for communication architecture for DIS has been created to meet the program needs of those programs using or scheduled to use DIS. The phased approach to the protocol suites

is one attempt to progress to communication technology which does not exist today but must be developed to meet the service requirements. This section describes some of those technologies/issues which are not currently addressed in this standard but will need to be addressed in the future.

1.2.3.1 Video Conferencing

A number of DIS documents, including the concept of operations, have identified a video conferencing requirement. This is to support exercise planning, briefing, and debriefing, but specific requirements (e.g. number of sites, functionality) have not been identified. The communications industry is creating new ways to achieve such video conferencing, but mature products are not yet available. Video conferencing is very demanding of network capabilities and will have a major impact on any DIS network design. Because the requirements for video conferencing are not clearly identified and because industry offerings are not stable, video conferencing is not addressed in this document. This requirement will however be addressed in future versions as the requirements and available services become better understood.

1.2.3.2 Interface to C^3I systems

It is anticipated that DIS will interface to Communication, Command, Control and Intelligence (C^3I) systems in the future. This issue, however, will require considerable study before any actions can be taken.

1.2.3.3 Interface to Field Instrumentation

DIS exercises will include participation of Field Instrumentation (FI) through the development of interfaces between DIS networks and FI equipment.

1.2.3.4 Interface to High Order Models (e.g. ALSP)

DIS will be applied to wargame simulations and other high order models in the future. Eventually the goal is for next generation high order models (e.g. WARSIM 2000) to be DIS compliant and link directly to other DIS entities. An interim step is necessary to link DIS with existing high order models. An interim step is necessary to link DIS with existing high order models. This may be done by creating an application gateway between DIS and the Aggregate Level Simulation Protocol (ALSP), the mechanism that now links major wargame simulations.

1.2.3.5 Emerging Technologies

DIS will be flexible enough to take advantage of emerging technologies, such as Asynchronous Transfer Mode (ATM), Synchronous Optical NETwork (SONET), Frame Relay, and emerging

gigabit technologies. These technologies will be included in the standard as the need for more encompassing communication services dictate their use. These technologies will not be included in this standard unless they are standardized, but they will not be excluded from implementations if they are not standardized.

1.3 Assumptions

1.3.1 Layers 6 & 7 Usage

The DIS standard for protocol data units describes and specifies the services of layers 6 and 7, the Presentation and Application layers of the protocol stack. The standard addressed by this rationale document will therefore not address the services of these layers unless they are needed to describe services/requirements needed in the lower layers.

1.3.2 Open Architecture

The architecture defined in this standard will be open via the use of commercial standards and protocols. Nothing proprietary will be specified.

1.3.3 Scalability/Extensibility

The architecture will be specified such that it is scalable and extensible. This will allow DIS systems to be designed to expand to meet more encompassing needs and to take advantage of emerging technologies.

1.3.4 Other Uses of the Same Network

The underlying communication networks used for DIS exercises via PDU traffic will also be used for video conferencing, bulk data transfer, voice and video.

1.3.5 Programs (i.e. Gov't Programs)

There are three categories of DIS applications: simulations, which include both manned simulators and Computer Generated Forces (CGF); instrumentation, which brings real hardware into the loop; and wargames, which incorporates aggregate level entities. For all categories, there are both existing DIS applications, which will require retro-fitting for the new standard and new procurements, which have been called out in the DIS standard. Each application has different bandwidth, PDU, and entity requirements.

The communication architecture requirements specified in the standard addressed by this rationale document will be utilized by the following programs:

Close Combat Tactical Trainer (CCTT)
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)

1.3.6 Compression

This standard will not specify any means of data compression other than what is included in specified protocols.

1.3.7 Simulation vs Network Management

For this standard, the distinction is made between simulation management and network management. Simulation management will not be specified by this standard. Network management will be covered by the specification of network management protocols.

1.3.8 Long Haul Connection

Simulators at different sites shall be connected via a Wide Area Network (WAN). The standard addressed by this document defines the functional and performance characteristics which shall be satisfied by the communications service, including the WAN. It is the goal of this communications architecture that the WAN be based on standards such as frame relay, Switched Multimegabit Data Service (SMDS), Broadband Integrated Services Digital Network (ISDN), and Synchronous Optical NETwork (SONET). The provision of the WAN will depend on the evolution of these high speed communications services in the marketplace and the particular organization using the DIS applications.

Wide area networks today do not in general support multicasting. If two or three sites using DIS are to participate in a demonstration or exercise, they could be interconnected by point-to-point circuits or by a network with sufficient capacity to support repeated transmission to each site. This, however, would become uneconomical for a larger number of sites.

The nature and development of WANs for DIS application is taking two distinct paths. The first is the establishment of a permanent infrastructure that will connect all DIS sites. Although physically one large network, it will support multiple exercises via the creation of individual logical networks for each exercise. This approach is called the Defense Simulation Internet. The second approach is the establishment of Ad Hoc WANs as necessary to support exercises and tests. The primary mechanism for this is the bandwidth-on-demands services starting to be offered by the major communications suppliers (e.g. AT&T, MCI, Sprint). The concept is that a network connecting any set of DIS sites can be created quickly and efficiently from

commercial services without the cost of maintaining a permanent infrastructure. The Advanced Distributed Simulator Technology (ADST) program is exploring this approach. This document does not assume either of these approaches and will support both of them.

2. COMMUNICATION FEATURES / SERVICES

2.1 Communication Service Requirements

Distributed simulation environment support requires various types of communication. The communication requirements encompass control and data. Data communications may be with or without real time requirements and will likely be augmented to include such things as voice, video and other forms of pictorial information. Upon the introduction of each of these forms of traffic, they shall share communications facilities instead of having disjoint facilities for each.

A summary of the communication service requirements is shown in Table I.

TABLE I. DIS Communication Service Requirements

Unicast
Multicast
Broadcast
Real Time Operating Speeds
Non-Real Time
Small Packets
Bulk Transfer
Reliable
Unreliable
Low Interpacket Dispersion for Voice/Video
Multicast Implementation
Multicast Management
Authentication/Access Control
Non-Blocking Interface
Flow Control
Low Latency Packet Delivery
Security
Flexible Entity Naming & Addressing
High Throughput

2.1.1 Service Requirements of PDUs.

Each DIS PDU requires certain services to make its communication practical. These services are grouped into broad classes of operation for DIS.

2.1.1.1 Communication Classes Based on Requirements.

This section establishes DIS communication classes based on the application service characteristics for both the required and recommended interim DIS PDUs. Each DIS PDU requires certain service characteristics to make its communication practical. These characteristics are grouped into broad classes of operation for DIS.

2.1.1.1.1 Application Requirements.

The DIS application (PDUs) has been characterized using the following subset of communication service requirements: unicast, multicast, broadcast, reliable, unreliable, real time, non-real time, packet size, and bulk transfer. The application service characteristics are used to define a service model necessary to support DIS communication. The service model developed from the PDU characterization shall be used to develop the interface to the application and lower layers.

2.1.1.1.2 DIS PDU Service Characterization.

DIS functional requirements are to provide: Entity Information, Entity Interaction, DIS Management, and Environment Information. Within each functional category, PDUs have been defined or recommended to satisfy specific requirements. The October 1991 version of the DIS standard defines ten required PDUs and six recommended interim PDUs.[1] The application services for required and recommended DIS PDUs are defined in Tables II and III, respectively.

Although packet size and bulk transfer are included as application requirements in 2.1.1.1.1, it is not presented in the summary tables for the following reason. 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 octets, and constitute the majority of PDU traffic. All PDUs listed in Table II and III 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. The bulk transfers fall into the Network and/or Simulation Management functions, but there are currently no PDUs

[1] The October version of the DIS standard specifies three recommended PDUs for Update Threshold Control. As of this writing, those PDUs have been removed from the standard and, therefore, will not be included in this characterization.

which reflect this type of interaction. Consequently, bulk transfer is considered a special case.

TABLE II. Required DIS PDU Communication Services

	Reliable	Best Effort	BC	MC	UC	Real Time
Entity State		*		*		*
Fire	future	*		*		*
Detonation	future	*		*		*
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 III. Recommended DIS PDU Communication Services

	Reliable	Best Effort	BC	MC	UC	Real Time
Emitter	*desired*	*		*		*
Laser	*desired*	*		*		*
Activate Request		*			*	
Activate Response		*			*	
Deactivate Request		*			*	
Deactivate Response		*			*	

Legend: BC-Broadcast, MC-Multicast, UC-Unicast

DIS Management will require additional capability beyond the activation and deactivation PDUs. Although these capabilities have not yet specified, Table IV projects additional application requirements for these areas.

TABLE IV. DIS Functional Requirements Communication Services

	Reliable	Best Effort	BC	MC	UC	Real Time
Network Management		*			*	
Simulation Management	*			*desired*	*	

2.1.1.1.2.1 Entity Information.

The Entity State PDU (ESPDU) constitutes the bulk of network traffic for a simulation exercise. Currently, the appearance updates represented by the ESPDU are of most interest to exercise participants within a limited radius of the initiating entity. Any exercise participant which is not in the area of interest, but receives the ESPDU, will have to filter out this unwanted information. Therefore, Entity State has a strong requirement for multiple multicast interactions. Multicast interactions deliver identical packets to multiple recipients as part of a single sender operation. A multicast data transfer provides co-located entity groups the capability of communicating state information based on locale in the simulated exercise.

In addition to their multicast requirements, ESPDUs must be delivered in real time but do not need to be transmitted reliably. Dead Reckoning (DR) algorithms are used to predict the entity's position over time in order to preserve network bandwidth by reducing the frequency at which state information is required. Reliability need only be a best effort. If an ESPDU is lost, the DR models used to reduce network traffic may also be able to account for the lost packet.

2.1.1.1.2.2 Entity Interaction.

Entity Interaction PDUs have varied characteristics. Within the Weapons Fire category, the Fire PDU (FPDU) and the Detonation PDU (DPDU) have the same service characterization. Similar to the ESPDU, both the FPDU and the DPDU have a strong multicast requirement. This requirement allows only those entities within the area of interest to receive information about weapons firing and detonation.

These PDUs are also desired to have a real time requirement in the future, and should be as reliable as ESPDUs. Whereas ESPDUs can rely on DR to extrapolate position after packet loss, FPDUs and DPDUs are not as robust. When a weapon impacts, it is crucial that everyone in the multicast group receive that information so "killed" targets do not continue to play in the exercise. A high degree of reliability is desired for the FPDUs and DPDUs, however current multicast protocols do not provide this service. Therefore, FPDUs and DPDUs must use a best effort realtime multicast service.

The Logistics Support PDUs (i.e., Service Request, Resupply Offer, Resupply Received, Resupply Cancel, Repair Complete, and Repair Response) represent activities which, although long in duration, do not require real time service. The resupply and repair interactions require a simple reliable transaction (request/reply) paradigm. This reliability is built into the application by pairing the acknowledgement (or reply) PDU with the request (e.g., Service Request and Resupply Offer PDUs). The Logistics Support PDUs do not require multicast, because only the entities involved in the service are interested. Therefore, the Logistics Support PDUs are characterized as requiring an unreliable unicast service.

The last required category of PDUs in Entity Interaction is Collisions. Collision PDUs require a real time, unicast service. Again, only the entities involved in the collision will be interested in this information. Changes in entity appearance resulting from the collision will be communicated using ESPDUs.

The only category of PDUs not required for Entity Interaction is Electromagnetic Interaction. Electromagnetic Interaction currently consists of two recommended PDUs, Emitter and Laser. Both PDUs desire a reliable real time multicast transmission but, as stated before, this is not available. Therefore, these PDUs are characterized as requiring best effort real time multicast.

2.1.1.1.2.3 DIS Management.

There are no PDUs specified for Network Management. Network management will be handled by a standard network management protocol (e.g., Simple Network Management Protocol or Common Management Information Protocol) and will not require DIS PDUs to accomplish the management of the physical network. Network management is accomplished with an unreliable unicast service.

The Simulation Management category of PDUs is responsible for the activation and deactivation of simulation players. The request to activate or deactivate entities in a simulation exercise requires a simple reliable transaction (request/reply) paradigm. The reliability is built into the application by pairing the acknowledgement (or reply) PDU with the request. This service is

characterized as non-real time unicast. Other possible functions of Simulation Management include management and control messages spanning multiple exercises. This type of service desires a reliable multicast transmission, however reliable multicast is not currently available. Therefore, this type of service is characterized as reliable unicast. In addition to the packet form of interaction, there are situations which mandate non-real time, point-to-point, reliable bulk transfer. Such situations arise when moving large items such as databases or video images. Standard file transfer protocols such as File Transfer Protocol (FTP) or File Transfer Access and Management (FTAM) will be used.

There are no PDUs required or recommended for Performance Measures. If PDUs are developed for this functional area, the required services will fall into one of the established service classes.

2.1.1.1.2.4 Environment Information.

There are no PDUs required or recommended for Environment Information. If PDUs are developed for this functional area, the required services will fall into one of the established service classes.

2.1.2 Communication Classes.

From the previously stated rationale, 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 V.

TABLE V. DIS Application Service Model

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

2.2 Communications Models

One of the important tasks facing the DIS standards community is determining the services DIS requires from the communication systems with which a simulator is implemented. For such a determination to take place, certain terms and classes of service must be defined, then the advantages and/or limitations of each class of service must be described. The goal of this section is to provide a high-level view of the different services under consideration for DIS .

In this section, a connection is a named association of endpoints and communications resources. For correct network operation, this name must be unique across a communications network at any one point in time. The term "connection" is used in reference both to the logical endpoint association and to the association's physical realization in network state and topology. An open connection is one which has undergone initial setup and whose name has been specified. Closing a connection dissolves the association of endpoints and releases the connection's resources. A persistent connection requires an explicit setup procedure, but can later be referenced by name with little management overhead. TCP opens persistent connections (from a user perspective, at least). A transient or connectionless connection exists for the life of a single message; subsequent messages between the same endpoints might require further setup overhead (though some implementations attempt to alleviate this drawback). Some uses of UDP open transient connections.

The process of associating an endpoint with a connection is called adding the endpoint into the connection. There are two kinds of adding: a join begins with a request from the endpoint to the connection, and an invite begins with a request from the connection to the endpoint. Disassociating an endpoint from a connection is called dropping the endpoint from the connection.

An example network which is referenced in the following discussion is shown in Figure 1, along with a very "generic" connection. The network consists of nine simulators at four sites. Bold lines indicate links which are associated with the connection. Arrows indicate a direction of data flow; numbers adjacent to arrows indicate some arbitrary measure of necessary resources ("bandwidth"). The connection shows eight participant simulators, each with different transmit and receive resource requirements. Note that simulator **b** is receive-only. The bidirectional sum of necessary resources for a connection is identical for all associated links, and is equal to the sum of the transmit resources of all associated endpoints.

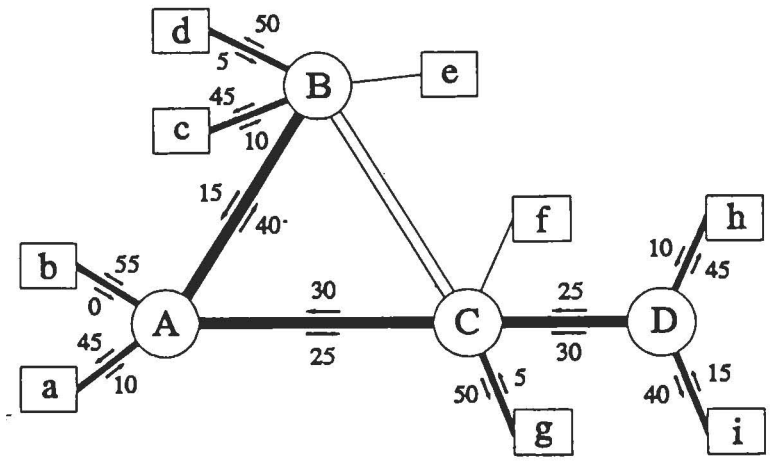

Figure 1. Example Network With Generic Connection

2.2.1. Number of Endpoints in a Connection

Connections are often characterized by the number of endpoints with which they are associated. Perhaps the most common connection is between two endpoints. This is referred to as unicast. Two unicast connections are shown in Figure 2. The connection between **b** and **d** is two-way unicast, since both transmit; the connection between **g** and **h** is one-way unicast, since only **g** is transmitting.

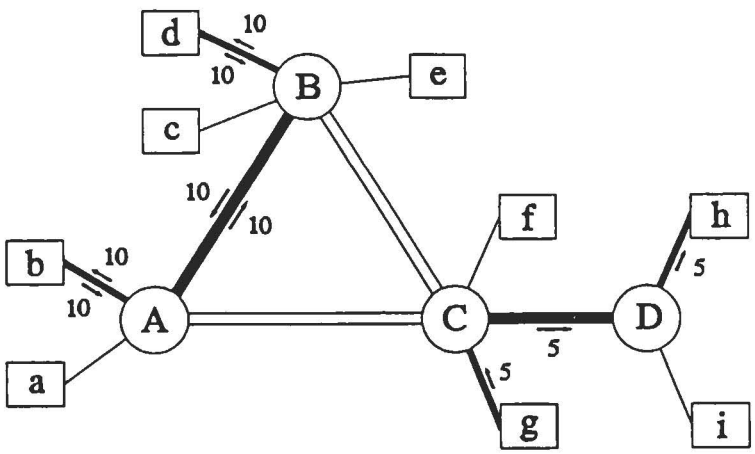

Figure 2. Unicast Connections

Connections between a subset of all possible endpoints are multicast. Figure 3 illustrates two multicast connections. The left-hand connection is many-to-many, since all endpoints transmit. The right-hand connection is one-to-many, since only f transmits.

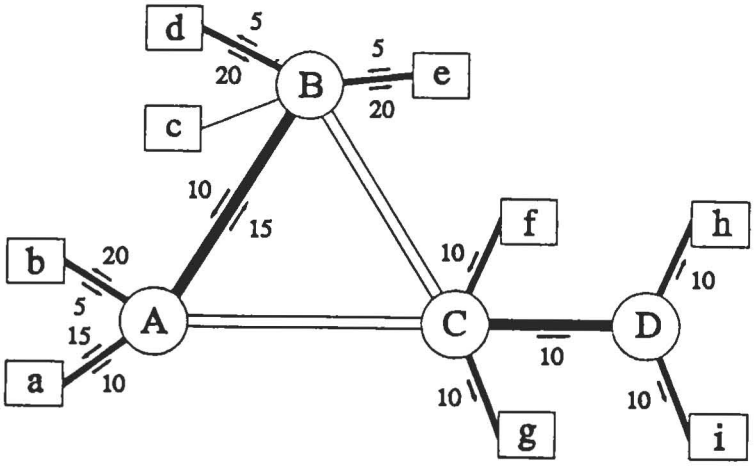

Figure 3. Multicast Connections

When all possible endpoints are associated with a connection, the connection is broadcast. A one-to-all broadcast connection is presented in Figure 4; only e is transmitting. If all endpoints were transmitting, the broadcast would be all-to-all.

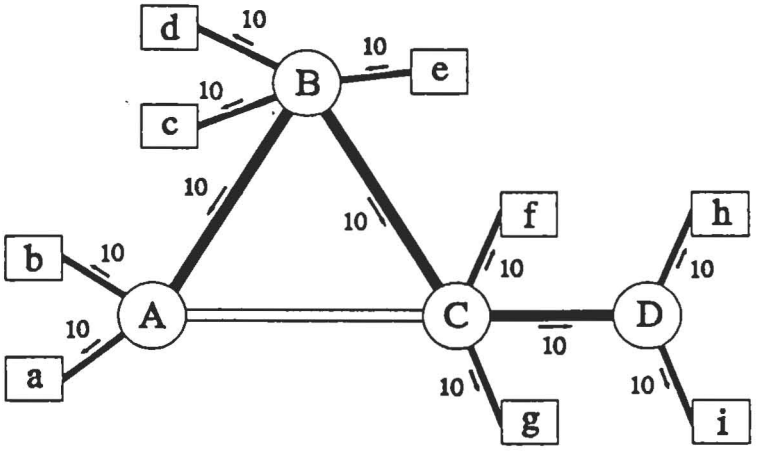

Figure 4. Broadcast Connection

It should be noted that both unicast and broadcast are special cases of multicast. Since it is rarely the case that all endpoints attached to a network are associated with a connection (in general, only network management functions might require broadcast), most DIS connections are multicast, with a few unicast connections.

2.2.2 Connection Resource Allocation

The resources (and endpoints, since they require resources) allocated to a persistent connection can be *static*, or fixed, at the time of connection setup, or they can be *dynamic*, changeable during the life of the connection. Static vs. dynamic resource allocation has implications with respect to:

- connection control/ownership,
- communications link routing, and
- bandwidth allocation.

Protocols exist where some of these three properties are static, while others are dynamic.

The "owning," or controlling, endpoint of a connection is quite important for most protocols. The owner might direct which other endpoints join or drop out of the connection, might specify the resources requested for the connection, is generally billed for the connection, and is usually designated somehow within the name of the connection. With such single-owner protocols, if the owner endpoint fails or needs to drop out of connection, the entire connection might be closed. Some protocols allow for such occurrences by providing a mechanism for ownership "handoff" to another endpoint in the connection.

Though routing and bandwidth allocation are strongly coupled, most protocol implementations make routing the more inflexible resource, once established. Dynamic adding of new endpoints into a connection can make bandwidth demands which invalidate previous routing decisions: a link which can support 5 endpoints might not support 10 endpoints. If a protocol is not capable of re-routing under such circumstances, it can not guarantee service to added endpoints without worst-case bandwidth allocation at the time of connection setup.

2.2.3 Per-Message Reliability

A reliable connection provides a mechanism to guarantee that each message is delivered and delivered intact. Protocols supporting such connections require some form of acknowledgement and retransmission facility. Unreliable connections make no delivery guarantees.

Reliable protocols are well-understood for unicast connections. Mechanisms for reliable multicast do not exist. The overhead

incurred by a reliable multicast transmitter while processing the acknowledgement/retransmission data for all receiving endpoints can be unacceptably high. Furthermore, the definition of "all receiving endpoints" is a problem for connections which allow dynamic adding and dropping of endpoints. How does an endpoint acquire a list of the other endpoints in a connection, then maintain that list up-to-date? A related and similarly handled problem, that of determining whether or not a connection is open, is discussed in more detail in the next subsection.

2.2.4 Associating Endpoints with Multicast Connections

Let us begin with the fundamental question "How does a new endpoint **n** become associated with a connection *C*?" The endpoint might be associated for the duration of an exercise, or for only a short portion of an exercise. The endpoint might need to be associated just with *C*, or with *C* along with several other connections related, in some manner, to *C*. Our question is tied to several issues:

- ownership of connections,
- what happens to a connection when the "last" endpoint drops out, and
- sets of related connections.

Given the operations defined previously in this section, there are three ways for **n** to become associated with *C*:

1) **n** can be invited into *C* by some endpoint **e** already in *C*,
2) **n** can join *C*, or
3) **n** can open *C*.

The first possibility assumes that some endpoint **e** in *C* knows that **n** should be invited into *C*. This assumption is valid for connections with a static set of endpoints, but not for connections with endpoints dynamically joining. The other two possibilities are alternatives depending on whether or not *C* is open. Actions take place based on that knowledge: the answer to the query "is *C* open?" can not change between the time **n** poses the query and acts on the result. Some means must exist to assure that *C* should not be closed or will not be opened by some other endpoint. The query is but one phase of a full-fledged distributed database transaction.

In order to determine if *C* is open, **n** must route to some information base with knowledge about *C*. A multicast connection associates a set of endpoints, not just two. As such, it is not always the best choice to name a connection by distinguishing one endpoint (which would usually be the owner). The obvious information base with knowledge about *C* is some endpoint already in *C*. If no such endpoint is identified through the connection name, how can **n** get data about *C*?

Three approaches can resolve this situation:

1) *C* is always open and a route to it is well-known,
2) some well-known endpoint always knows if *C* is open, or
3) all endpoints can be queried to determine if *C* is open.

If *C* is always open, **n** must still be able to route to it for a join to take place. The most common example of this arrangement is that messages in *C* are actually a "filtered" subset of the messages of some general, statically-opened connection *G*. "Joining" *C* simply means that **n**'s *G*-filter is modified to accept another type of message.

Approach 2 implies the existence of a database server with global knowledge of all open connections and *some* means to route to them. One logical choice for this a statically-open Global Exercise Manager, about which all joining endpoints would have enough data so that they could route to and query it.

Both approaches 1 and 2 allow *C* to remain open across points in time when, temporarily, no platform-simulation endpoints are associated with *C*. For approach 1, *G* is open independently of *C*. For approach 2, if the database server or some other statically-added endpoint owns the connection, it logically acts as "the last endpoint" from a protocol point of view.

Approach 3 requires a query of all possible endpoints which might be associated with or own *C*. First, this implies broadcast, at least within the scope of <u>all possible</u> — not just current — DIS exercise participants. Second, and more important, is the transaction nature of associating **n** with *C*. *C* should not close while **n** is attempting to join it, but some other endpoint *must* not open *C* while **n** is trying to open *C*. This is a classic "distributed consensus" or "distributed snapshot" problem. Algorithms to resolve such problems are known, but are complex and not implementable across even local area networks within the real-time latency limits identified by CASS. This approach is thus of questionable merit for connections requiring dynamic joining and/or dropping of endpoints.

Back to the original list of issues above, the final issue concerns sets of "related" connections. For instance, **n** might need to join connections *C*, *D*, *E*, and *F*. A very dynamic example of this arises from the "segmented battlefield" concept for defining multicast connections as representing geographic areas. Targeting handoff could bring several areas, and thus connections, into a platform's field of interest simultaneously. High-range sensors might deal best with much larger segments of geographic area than do short-range sensors, lest they be required to listen to literally hundreds of connections at once. Both real-time multiple-connection joining and hierarchial connections are most easily implemented with a message-filter approach.

2.2.5 Multicast Resource Allocation and Routing Policy

Protocols which implement multicast connections generally handle the case of multiple transmitters in one of two ways. These approaches are:

- allocate resources for the connection as a whole, or
- allocate resources on a per-transmitter basis.

These two approaches can be effectively the same for connections with a static set of endpoints; they might exhibit tradeoffs only for connections which support dynamic adding and dropping of endpoints.

Protocols which treat a connection as a whole can route and allocate resources more rapidly than those which independently route and allocate for each transmitter. Potential disadvantages, however, begin to be apparent when one starts adding endpoints. Either sufficient "worst case" bandwidth must have been allocated to the connection at setup time, or additional bandwidth must be allocated for the new transmitters. This brings up the possibility of forcing a re-route, or of refusing service to the new endpoints.

Those whole-connection protocols which support route reconfiguration per transmitter can avoid service refusal in this case. Even those which can re-route do not necessarily establish an optimal route. Whole-connection allocation generally overallocates bidirectional bandwidth on internal network links. Whereas, in the optimal state, the sum of bandwidth in *both* directions along a link equals the total transmitter bandwidth, whole-connection allocation is usually defined so that the bandwidth in *each* direction is set to the total transmitter bandwidth. Requested, and thus billed, bandwidth is twice the optimal requirement. Due to such bandwidth overallocation and due to routing all transmitters through the same links, routes are theoretically harder to find through congested networks.

Protocols which always allocate and route per-transmitter can allocate bandwidth exactly and can more easily route around network congestion points. Connection management for adding and dropping endpoints is much more difficult, though. Messages for the "same" connection can come in from different links. Processing and hardware overhead exists for maintaining and merging the different physical connections into one logical connection. Each incoming link can exhibit different latency properties, so messages from different simulators at the same site can arrive at quite different times. Whenever an endpoint **n** adds in or drops out, all other endpoints in the connection must be updated to connect to or disconnect from **n**. For a connection *C*, this is a distributed transaction problem of similar complexity to the "is *C* open?" query, but requires processing by all endpoints in *C*, not just by *C*'s owner or some database

server.

As a distributed transaction, the add/drop problem is amenable to either a database server or distributed snapshot solution. Complex, protocol-specific endpoint management appears to be outside the scope of CASS' task and against the "open architecture" premise of DIS. Protocols which do not perform multi-transmitter endpoint management themselves are thus not suitable for connections requiring dynamic adding and dropping of endpoints, though are suitable for connections whose endpoints are statically determined.

The issue of efficiency in per-transmitter bandwidth allocation and routing does not exist if service must be guaranteed for the full duration of an exercise. For an endpoint not to be refused a join into a connection, bandwidth for that endpoint must be available. This can not be guaranteed unless sufficient bandwidth for all potential endpoints is reserved at the time of exercise setup (as is the case for leased lines, but not necessarily for commercial service where other users are also on the network). If worst-case resources *must* be pre-allocated, per-transmitter allocation provides no savings.

2.3 Grouping of PDUs

Non-contention digital communications systems operate most efficiently (i.e. have the greatest throughput) when the packets that they handle are at or near the basic maximum length for which they were designed (e.g. 4352 octets for FDDI). This is due to the fact that overhead portions of the packet are of constant length and the processing time for each packet is fairly constant. Therefore the ratio of user data to overhead increases as the length of packet increases. If, however, message length becomes greater than the basic maximum packet length, the communications system must break the message up into smaller units. Such activity increases overhead and reduces efficiency.

The PDUs defined in the DIS program are relatively small compared to the maximum data area of a typical packet (e.g. with frame size for IP=20, UDP=8, and TCP=20, Ethernet data area for DIS PDUs is 1472 octets long for UDP+IP and 1460 octets long for TCP+IP). If each PDU is sent via a separate packet, the overhead ratio would be high and the throughput would be limited. One method of improving the situation is to pack multiple PDUs into a single communications packet. To this end we recommend that:

1. A single platform simulator group all the PDUs generated by a single iteration of its model(s) into packets. This may result in entity state, emissions, fire, and voice PDUs in a single packet. However, PDUs should not be "collected" from iteration to iteration of the models just to make communications more efficient. To do so would create

excessive delays between the time the PDU is created and the time it is sent.

2. A Computer Generated Force (CGF) unit group as many entity state (and other PDUs) as possible into each communications packet.

3. A gateway or router consolidate those PDUs arriving within a short time interval (e.g. 10 to 20 milliseconds) into maximum sized communications packets.

Concatenating moderately sized PDUs within LANs is likely to improve bandwidth utilization at a cost in increased latency. One negative impact of concatenation on latency is the increased processing time required to examine queues for pending transmittals with the same destination. With a frame size limit of 1500 octets, Ethernet LANs are poor candidates for concatenation of moderately sized DIS PDUs. While the frame size in FDDI is considerably larger (4352 octets) and is fixed length, its transmission rate is an order of magnitude greater than Ethernet. Thus, the token holding period expires quickly and the node can easily lose its transmission window while trying to pack additional PDUs into the frame. A second negative impact on latency comes from the increase probability of collisions in a contention environment (e.g. Ethernet, packet radio)--increase packet size results in a greater probability that some portion of the packet will collide with another packet.

It is in long haul where the benefits of concatenation usually outweigh the cost. Encryption overhead is applied to each packet regardless of size. Each router/gateway connects to a dedicated link and generally more limited bandwidth (a T1 provides from 0.1 to 0.01 the bandwidth of the LANs it connects), latency issues may become secondary to efficient use of bandwidth. It should also be noted that the LAN(s) at either end of the gateway will have already filtered out packets with destinations that they can handle, so the gateway parses a more limited subset of destinations.

It is most important to remember that all PDUs put into the same communications packet will be sent to the same destination. Therefore PDUs with different destinations should never be put into the same packet.

The maximum number of octets available for PDUs as viewed from layer seven (application) is a function of the maximum packet size of the level two protocol used (e.g. Ethernet), less the overhead (packet headers and trailers) used by the intervening layers.

The packing of multiple PDUs into communications packets must be done at the application layer, for there is no provision for

doing so in the COTS protocol suites defined in for phase 0,1 or 2. The mechanism for packing PDUs is left to the developer of the application layer software. We do recommend that this function be provided in third party Network Interface Units (NIU) being developed for the DIS market.

There is no mechanism specified as to how the PDUs are to be packed (they are simply concatenated in a buffer) and there is no indication in the communications packet that it contains multiple PDUs. A recommendation for a Concatenation PDU to make the situation explicit has floated around the DIS community for some time but has been rejected. For this reason we strongly recommend that the input processing software (commercial NIUs included) assume that there are multiple PDUs in each received packet.

Multiple PDUs should be concatenated into a single UDP datagram for Phase 0. No extra framing or encapsulation is needed.

2.4 Packet Length

Packet length in DIS will be largely determined by the specific PDU length and required protocol headers. In general, the characteristics of the architecture will determine whether extremely large PDUs or moderately sized PDUs (500 to 1000 octets) are optimal. Small PDUs (less than 100 octets) are never optimal simply because the ratio of header overhead to user data is excessive (e.g. 54 additional octets in the case of an 802.3/IP/UDP LAN).

For Phase 0, the IP data portion of a packet can, in theory, be up to 64K octets, however, transmitting PDUs in excess of 1500 octets is a less efficient use of bandwidth and processor capacity for Ethernet LANs.

2.4.2 Fragmentation

At least one DIS PDU has already been defined to exceed some LAN limits (in the worst case, the variable length Emitter PDU may be 9632 octets, see section 4.1.1). This would require fragmentation in some LANs. Since all IP implementations are required to support reassembly but not fragmentation (see RFC 1122), any host IP implementation to be used by DIS should be required to support both fragmentation and reassembly, with a maximum reassembled datagram size of at least 10000 octets, and preferably unlimited. The size of the individual fragments, before reassembly, will vary according to the limits on the various LANs and WANs in the path that a PDU takes. Many popular IP implementations refuse to broadcast (or multicast) packets that require fragmentation; the ability to broadcast and multicast fragmented datagrams should be required for any fragmentation on the local network.

2.5 Bandwidth Reservation / Guarantee

The need for a reservations service is tied directly to the offered load of the network in relation to peak utilization. In an undersubscribed network, the need for a reservations service is negligible--in a heavily oversubscribed network, the need for reservations may be substantial, but the cost will also be substantial. For DIS configurations, as is typical of many large networks, the need for reservations will increase with increased distance, number of links and number LANs. To ensure the availability of capacity for exercises involving longhaul, it will be necessary to support a reservation service by Phase 1 of DIS.

In an oversubscribed network with both reservations and demand assigned, a reservations strategy will either require that:

1. the reservation can force the clear down of no-reservation allocations to obtain the necessary end-to-end capacity;

2. the reservation can negotiate with Local Exercise Managers (LEMs) to impose flow control, in a manner that restricts flow on the non-reservations allocations up to the point that the reservations connections can be satisfied;

3. the reservation can wait for currently allocated bandwidth to be released--this strategy is most satisfactory when there are traffic statistics which can be used to estimate the amount of time prior to a reservation that allocations must be blocked (unavailable to any requester other than the reservation) to ensure that capacity is available at the start time of the reservation.

A quick assessment of the above strategies will show that (1) is really the only "Guaranteed" allocation, and is both brutal and simple. Strategy (2) results in degraded (but not interrupted) service for the non-reservations subscriber. It is an elegant solution with a hint of danger (lacks robustness). Finally, (3) is the classic solution for circuit switched common carrier networks that also offer premium services (e.g. video teleconferencing). Given extremely large capacity networks of demand assigned subscribers and a small percentage of reservations subscribers, this approach (3) is almost guaranteed. It is also the most wasteful, in terms of unused and unavailable bandwidth and requires a very centralized, statistically based implementation. Thus, strategy (1) is recommended for DIS and may be implemented at either the GEM (Global Exercise Manager) or at a LEM.

2.5.1 Method of Allocation

A reservations strategy is similar to a priority scheme. For

allocations involving multiple physical links over multiple subnets, the best point at which to process reservation control messages is in the transport layer. Defined in the OSI Transport Layer Service Definition, is the Quality of Service (QOS) parameter list. One of the QOS control parameters is 'priority'; it is recommended that this field be made available to a reservation service. An associated parameter is 'throughput' which is further subdivided into directional and measured (vs. allocated) throughput. Coupled with each priority designation, then, would be the allocated throughput.

For Phase 0, the IP TOS field can be used. This field includes flags which request low delay, high throughput, high reliability, or low cost (no combinations are allowed), as well as a three-bit priority field.

2.5.2 Start Time and Duration

Exercises are initiated and controlled at the GEM or LEM. Reservations are explicitly part of an exercise. At some appropriate interval prior to the reservation's desired start time, connections will be requested for the participating nodes (hosts). GEM or LEM processors will initiate requests to connect between the logically assigned nodes. The exercise manager will also initiate disconnects when the exercise is completed. Subnets are assumed to be asynchronous, so commands from the Exercise Manager are executed as received.

A LEM or GEM should place an upper limit in the total capacity available for reservations. In a token bus or token ring network with no demand assigned (non-reserved allocations), it may be possible to set this threshold at 90% capacity, however, in a mixed network with one or more contention subnets (e.g. ALOHA, CSMA/CD), the threshold may be anywhere from 18% to 50% of total capacity in the subnet.

2.5.3 Congestion Control

The triggering of congestion control should be infrequent in Phase 1 or 2 of DIS, since most allocations will be connection oriented. Nonetheless, equipment failure and the mixing of processors and communications links of widely varying capacities, will necessitate a congestion control mechanism. Nodes with reservation should honor choke messages, however, centrally issued choke messages (e.g. from a GEM or LEM or LAN node controller) should be ordered such that non-reservations connections to the congested node are sent choke packets before reservations connections. If a congested node or gateway is the transmitter of choke packets, it should throttle non-reservations connections before reservations connections.

2.5.4 Flow Control and Alternate Path Routing

Reservation allocations should not be decreased by imposing flow control. Conversely, a reservation should not be allowed to increase its allocation after initialization by using flow control requests.

A reservation implies a static environment with connectivity completed at initialization. However, if there is a problem at a gateway or along a long haul path, and if alternate paths exist with sufficient capacity for the reservation, the reservation should be dynamically routed to the alternate path.

2.5.5 Contiguous Allocations

For some non-digitized video, audio and sensor transmissions, it is not possible to incur delays in between packets without a noticeable breakup on the receive side. For that reason, one of the possible uses of reservations will be to allocate a connection as contiguous bandwidth in blocks of 32 kbps, 64 kbps, 384 kbps, etc. Such allocations require access to the node's station management functions (at the MAC and PHY layers) using the MIB (Management Information Base) at these layers as well as the transport/network layer MIB. Node to node (or GEM/LEM) control is initiated using SNMP messages. It may be necessary to force reduced traffic loading (via flow control messages) on contention LANs or WANs, if these long sequences of frames are to avoid collisions. Use of flow control packets in this case would be restricted to reservations initialization and to contention type subnets.

2.5.6 Problems with Concurrent Reservation Initialization

LEMs and the GEM must be able to communicate with each other during the implementation of a reservation. While a reservation is being initialized from one Exercise Manager, all the other Exercise Managers should inhibit any reservations implementation of their own. This avoids dual seizure conditions and partial allocation deadlocks.

Two or more LEMs establishing connections concurrently for two or more reservations with similar start times, may result in partial and incomplete allocations. For example, if along LinkA only 500 kbps is available and within LinkB only 400 kbps is available and both LEM1 and LEM2 need 300 kbps on LinkA and LinkB to satisfy the two different reservations, the following could occur if concurrent reservation initialization is allowed:

> LEM1 allocates and holds 300 kbps along LinkA and LEM2 allocates and holds 300 kbps along LinkB. Now each Exercise Manager attempts to complete the reservation but both find that there is insufficient capacity available at the other

link they need. Neither can satisfy their reservation so both reservations are denied.

In fact, one of the reservations could have been satisfied if LEM1 had been allowed to allocate a complete reservation before LEM2 initialized its reservation request.

Two or more LEMs establishing a connection concurrently for two or more reservations, may also result in a blockage referred to as a dual seizure. For example, LEM1 begins establishing a connection by allocating the remaining capacity from point A to B to C to D. At the same time, LEM2 attempts to establish a connection from point D to C to B to A. Both Exercise Managers reach an impasse at the B/C boundary coming from opposite directions. One must back off to let the other complete the connection. The problem is easily avoided if concurrent reservation initialization is prohibited.

2.5.7 Advance Reservation Logging

Exercise Managers should provide a service which stores a reservation in advance. This allows two important features to be implemented at the GEM/LEMs:

1. Negotiation rather than denial -- The Exercise Manager can implement a dialogue which examines alternative capacity, start time, duration and node connectivity values with the requester, if the original reservation request is likely to be denied.

2. Efficient use of capacity -- Advance Reservations can result in capacity utilization which approaches the efficiency of token ring or TDM strategies rather than that of contention techniques.

The above, of course, assumes that there is no reservations override option and that reservations are honored on a first come first served basis.

3. ARCHITECTURE

3.1 Topology and Components

The basic job of the communication subsystem is to provide an application interface for the DIS protocol with interconnection between each of the participating simulation and simulation support entities. The environment is heterogeneous, multi-vendor, multiple developers, and multiple owning or operating agencies. Heterogeneity extends not only to the collection of participating hosts, but also to the variety of communication medium, various operating systems, and various languages for software development. The diagram in Figure 5, illustrates the

communications subsystem as a protocol stack of 7 layers. The DIS PDU's are application messages which connect with the Application Programs. The applications contain entities, environmental objects and other objects such as simulation support services.

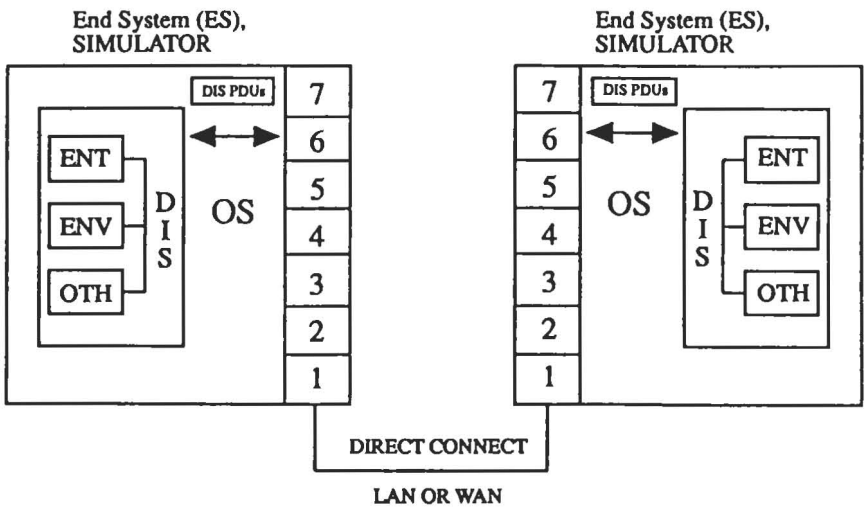

Figure 5: Communications Subsystem Architecture

Various types of communication facilities can be used to form the virtual network as illustrated in Figure 6. The communications medium, a the lowest level, include wire, fiber-optic, satellite, micro-wave, etc. These medium may be used interchangeably, as performance characteristics permit. Communication services should be independent of the means of communication, to the maximum extent possible. Figure 5 also illustrates that applications include manned battlefield simulators, Computer Generated Forces and real instrumented platforms. A Cell is a homogenous set of simulators which can be distributed in a variety of ways. Cell Adapter Units (CAUs) interface between non-DIS compliant applications and the DIS protocol. Hosts are defined by interfaces with the network.

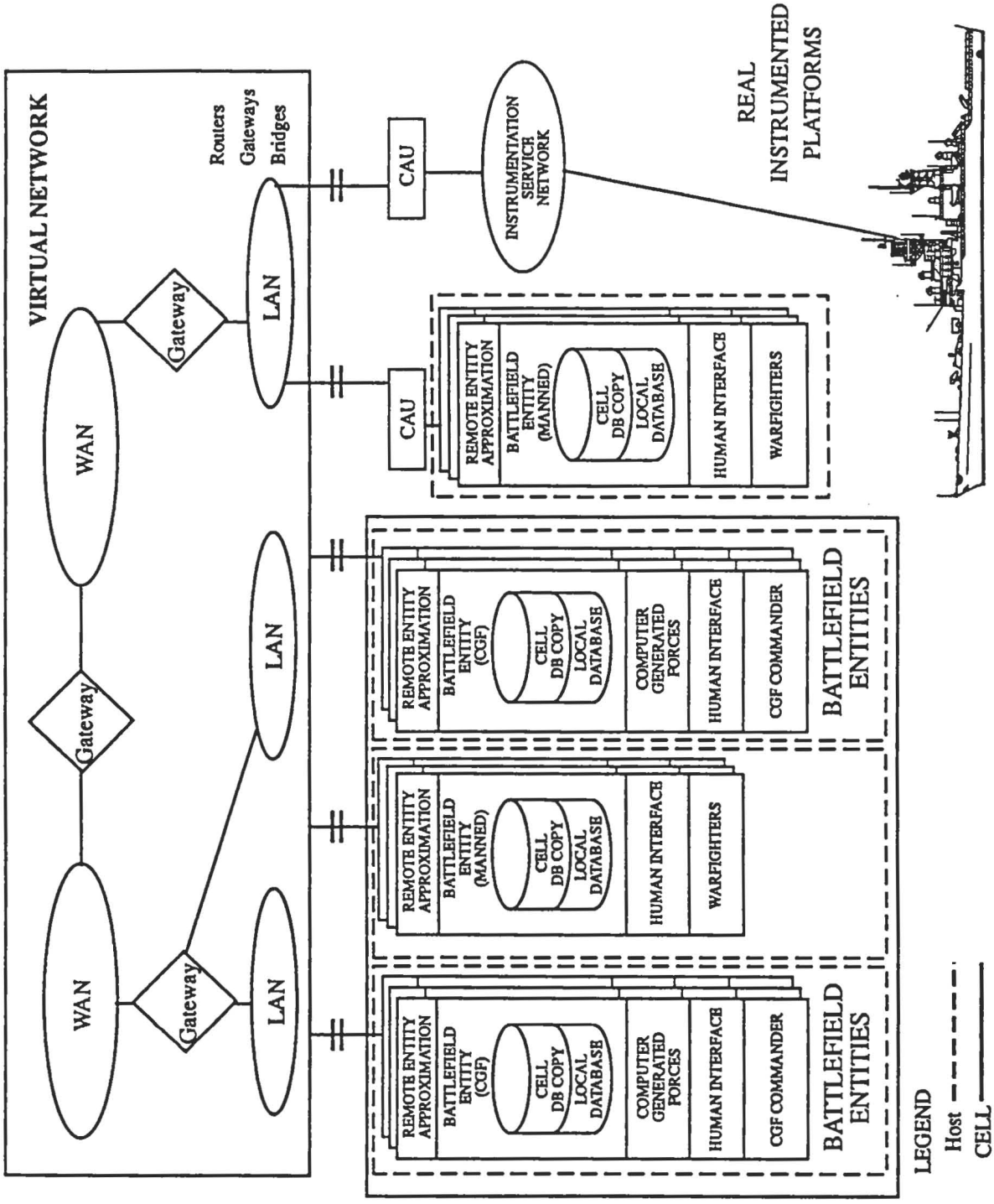

Figure 6 A Variety of Interconnection Modes For The DIS Architecture Virtual Network

Various types of communication are needed to support a distributed simulation environment. These currently include control, data, voice, real time, non-real time, and will likely be augmented in the future to include such things video and other forms of pictorial information. It is desirable from a usage and communication management perspective for these various forms of traffic to share communications facilities, instead of having different and disjoint facilities for each type of communication. Man-in-the-loop simulator based training and experimentation are the domains of interest. Thus, a lower performance bound is set by certain interactions which must proceed at human reaction rates which reflect the situation being simulated.

It is anticipated that multiple, simultaneous, independent training sessions will take place even on a single instance of a DIS facility. Therefore, mechanisms must be provided to ensure the separation and non-interference of these potentially conflicting activities. Similarly, there is a need for including some simulation components whose operating characteristics are classified. Mechanisms are required to ensure the separation of secure and non-secure parts of a simulation activity. In lieu of such mechanisms, entire simulation exercises must be insecure or secure at the same level.

The network design issues are naming, addressing, routing, flow control and congestion control. Real time, low latency traffic and non-real time traffic will have different requirements/tolerance of the performance impact of flow control mechanisms.

3.1.1 Naming

Communication functions include a means for naming the entities participating in the communication. Naming functions are distinct and separable from addressing functions, which help to route messages to their proper (named) destination. Addressing functions are most often associated with an architecture or communication system design, while naming is often more related to the application of the communication. For simulation, names need to be assigned not only to hosts, but to simulated entities and perhaps their parts, and other services which populate the simulation environment.

Flexibility in the naming of the communicating entities will support flexibility and modularity in the application designs and implementations utilizing the communication. Flexible naming is one aspect of moving toward a more "object-oriented" system paradigm. Additionally, as the simulation environment gets more global and far reaching, it also gets more complex. This increases the necessity to separate the relatively infrequently changing name structure, from the relatively more frequently changing address structure. Naming functions also include group

naming in support of multicast operations. In DIS phase 0 there is no specific name service required.

3.1.2 Addressing

Inter-entity communication in a distributed interactive simulation environment consists largely of packets sent between two or more of the simulation participants. Messages are exchanged largely within an individual exercise, however some management and control type of communication could span multiple exercises. Additionally, although it is not currently the case, certain "servers" which may be costly to replicate could service clients in multiple exercise simultaneously.

Addresses exist at each level of the ISO model. At level 7, the entities are addressed. An entity contains three fields: Entity ID, site and host. The Entity ID is an unique address for the exercise. Entities can fire weapons as "events". These events exist in an address space relative to the originating entity. The Site is one interface on the Wide Area Network. The host is the simulator or Computer Generated Forces computer. Cells and exercises define groups of entities, but they do not have a network address. In the future, it is likely that entities can migrate between hosts or even between sites. This means that the Layer 7 address should be kept separate from any lower layer address.

The distributed simulation environment has a requirement for multiple 1-to-many interactions to maintain a shared notion of system state. These types of interactions, frequently referred to as multicast, deliver identical PDU's to multiple recipients, as part of a single operation on the part of the sender. In some cases, the many is a large group, typically all entities participating in a particular exercise (e.g. the entity state PDU). Many of these participants will be both sources and sinks of multicast activity.

Current implementations of multicast addresses occur at Layers 2, 3 and 4. At layer 2 the Local Area Network, IEEE 802.2, standard multicast addressing is used. The network layer multicast address, i.e., Internet Protocol in Phase 0, is used to map into the link layer address at Layer 2. The Internet Activities Board administrate the internet address space. Multicast addressing have already been allocated for special needs. For the DIS protocol, the network layer multicast is allocated to the Exercise ID and the Protocol Version number. The Layer 4 protocol provides the transport service. The address is the port ID. Additional multicast addressing can occur for ports. The DIS protocol is mapped into one preassigned port ID for the User Datagram Protocol in Phase 0.

In some cases the group associated with the multicast may be dynamic, with entities coming and going during the course of an exercise. In other cases, the groups may be static and setup prior to the execution of the exercise. It is anticipated, based on current experience, that a large number of multicast groups will eventually be needed.

3.1.3 Routing

Routing is performed by the gateway system. The gateway routing minimizes the number of "hops" to reduce latency, and routes multicast addressed PDU's to their destinations. The gateways can choose, independently from the DIS application, to use a connection-oriented or a connectionless protocol. In the connection-oriented approach, virtual circuits are established between the source site and all destination sites in the exercise. PDU's are then copied on each virtual circuit. The gateway effectively operate as a virtual Bridge.

In the connectionless approach, gateways route PDU's to minimize the number of hops. A desired property for the connectionless approach in DIS is that it also minimize the number of packet copies. Example protocols are the Internet Activity Boards Multicast OSPF and ST-II.

3.1.4 Flow Control

The flow control objective is the need to sustain real time operating speeds. Applications for training and evaluation purposes which include manned simulations need to keep pace with the real world entities they model, and with human reaction time. Our concern here is on the impact of inter-entity communication performance on network performance. Higher performance networks make more interactions per unit time feasible, and better compression techniques (e.g. dead reckoning, which compresses the number of messages needed, not the content of a message) make fewer interactions per unit time possible.

When we introduce "voice" data messages to the mix of traffic, another dimension of flow control becomes important. To be able to collect together and replay continuous voice messages, the inter-message dispersion in time of the individual parts can not be degraded too much. Current experience in this area suggests that an initial target for effective communication of continuous speech is inter-message dispersion of less than 50 milliseconds.

3.1.4.1 Error Control

Section 2.1 identifies PDUs which shall be delivered reliably. This means that each of those PDUs shall be delivered to its destination without error. Implied in this definition is that the receipt of each PDU shall be acknowledged and retransmitted

if necessary. Such acknowledgement and retransmission will be handled by the error detection/correction mechanism of the protocols used at level 4 and below. That is, there is no action required at the application level other than to indicate that a particular PDU is to be sent reliably. The receiving application can assume that all PDUs sent reliably are in order and intact.

PDUs not requiring reliable delivery shall be given best effort delivery. These PDUs make up the bulk of network traffic and include those PDUs that are multicast to all simulators in a DIS exercise. Acknowledgement and retransmission, associated with reliable delivery, is not feasible due to the additional latency and network bandwidth that would be required. There is also the possibility that a PDU with corrupted data may be received. The processing of such corrupted data may create unacceptable behavior in the receiving simulator. To prevent this, the DIS communications architecture shall include in its best effort delivery a checksum mechanism. Because this type of checksum is specific to DIS, its location in the protocol stack has not been defined. This checksum shall include the entire PDU. If a checksum error is detected in a received PDU, the PDU shall be discarded by the communications software. That is, it shall not be made visible to the application.

3.1.5 Congestion Control

Congestion occurs when the demand is greater than the available resources. The problem of congestion is not solved as resources become less expensive, such as computers, or as higher speed networks become available, such as FDDI and the Gbit networks. For example, suppose the LAN in Figure 2 is an FDDI and the WAN is the Defense Simulation Internet (DSI). The high speed LAN without proper congestion control can lead to reduced performance. With the high speed link, the arrival rate to the first gateway can become much higher than the departure rate, leading to long queues, buffer overflows, and packet losses that cause the latency and transfer time to increase.

One solution is demand reduction schemes. A slow-down control packet, known as the "source quench", is sent from the gateway to the source host. It is the hosts responsibility to reduce the speed by locating the offending entities and reducing their activity. The host could change dead-reckoning thresholds, or decrease the number of entities. The scheme must be fair. If one simulator is favored over another, it is more difficult to assure a "fair fight".

Another solution is to use prioritized traffic, so that lower priority PDUs are lost first in overflow situations. The question is how to prioritize the PDUs. The Entity State PDUs could be placed at a lower priority than other PDUs because the dead-reckoning algorithms smooth the results. The problem is that

there is a limit on how many lost Entity State PDUs can occur before visual cues such as jumps are noticed. Therefore, we do not recommend a priority scheme for congestion control.

For the phase 0 system, there are only two different speed links, the WAN and the LAN. The congestion control problem is handled using the source quench Internet Control Message Protocol (ICMP) packet. Designing a scheme that allows slower paths to be used depending upon the load levels on all paths is a topic for further study

3.1.6 Interoperability with Non-DIS Systems

There are three types of Non-DIS cells to be considered:

1. Previously Stand-alone simulators
2. Higher-Order Models
3. Live ranges and operational platforms

Previously Stand-alone simulators model the battlefield at the vehicle level. The differences in the simulator is in the Computer Image Generator (CIG), the terrain map and how it manages automated entities. Interoperability involves mapping simulator events and state into the DIS protocol and generating an accurate terrain data base for the simulation assets.

The Higher-Order models are interfaced through a Computer Generated Forces system. One of the benefits of translating these HOM's to DIS is the Plan View Display and Stealth capability of DIS.

The simulation assets of a DIS system can be used to provide range participants with the infrastructure of a larger battlefield than that possible using field equipment at the range. Interoperability involves mapping the state of a Range Control Center, the location and velocity information and events from the various platforms. An accurate terrain database of the range is needed for the simulation assets. One trade-off to be considered in translating the range protocol into DIS is the ease with which entities can join and leave the exercise. A reduced sized entity state PDU can be used for the instrumentation service network. However, simulation management must establish the appropriate databases when an entity joins.

3.1.6.1 Interoperability Requirements.

Interoperability that the DIS initiative is attempting requires that interactive operations be achieved at each site, consulting a standard document, and without communicating with each site. Such interoperability requires that the interface of each simulator to the network be specified down to the hardware plug. Simulators that comply, can be plugged in and will interoperate.

Strict interoperability requires that the standard take care of all technical aspects of linking together parts of the network. Only administrative details are left to be negotiated by the participants and the network.

Much progress has been made over the past decade on standardizing approaches to interconnecting computer systems. Three aspects of distributed interactive simulation distinguish DIS from the more general computer/communication interconnection. These are: 1) real time delivery requirements for interactive, man-in-the loop behavior 2) multicast delivery options for convenient updating of shared data items and 3) military security considerations. The approach to interoperability specified in this standard is to adopt the more general communication framework, augmenting it only as necessary to meet the specific additional requirements of distributed interactive simulation.

Any approach taken toward communication interoperability must apply to as wide a variety of existing simulators as possible, preferably all. This interoperability integration shall be possible with minimal disruption of existing simulators, even at the expense of optimality and efficiency. To accomplish this for the widest class of existing simulators, including those already interconnected and those running stand alone, only the minimum properties should be standardized to accomplish the integration. This allows as many pre-existing configurations as possible to remain compliant with the minimum change, as well as accommodating the maximum flexibility for future innovation with minimum disruption to working systems.

There are many approaches to integrating a simulator into an integrated DIS exercise which fit within the framework outlined above. From an architectural point of view, the following list enumerates a variety of possible simulator organizations, all of which are appropriate for meeting DIS interoperability requirements:

- a simulator and its DIS communication interface can coexist on a single host computer.

- a single host can run multiple simulations, using the same or different DIS host identities for these entities.

- a dedicated front end processor can be used for implementing the communication interoperability (as well as other DIS) requirements for one or more back end simulators. This approach is sometimes referred to as an "application gateway". One of the primary advantages to this approach is minimal interference with currently operational simulators. The interconnection of the application gateway with the simulator is not subject to the DIS standardization effort. A reasonable implementation of such a component might be

useable by various classes of simulators.

- a simulation implementation can span multiple computers, either as part of a multiprocessor system, locally distributed, or even with geographically distributed components. With such arrangements, from the vantage point of the network, a single component is designated as representing the simulation in its entirety. Any information distribution among the components is entirely the responsibility of the simulator.

3.2 Protocol Suites

3.2.1 Role of the Communication Architecture.

The ISO Reference Model is probably the most widely referenced model for communication architecture, and we adopt its use here. Under this model, the communication interconnection problem is broken down into seven layers, each with specific responsibility in carrying out part of the overall communication integration. The development of this reference model was in large measure motivated by and patterned after the success of the DARPA Internet program, which was the pioneer of the general machine interconnection technology base. Along with the development of the reference model, ISO has developed a series of protocols which in some cases mirror comparable entities in the Internet, and in other cases extend and formalize concepts only primitively developed by the Internet program. Currently, there are two dominant suites of protocols (Internet and ISO) which fit within the Reference Model communication architecture and are instantiations of a solution to the general communication interoperability problem. They differ in their details, in their maturity, in their number of options, in their flexibility, in their performance, in their number of currently available commercial products, in their number of fielded systems, and in their organizational support, among other factors.

Within level 3 of this reference model there is functionality which is key to a generalized interconnection model. This network level provides for packets of information to be transparently delivered from system to system across almost arbitrary interconnections of local and wide area networks. By adopting the low cost conventions of providing for remote delivery even when delivery is actually local, and through the provision of gateway processors linking the local and wide area networks, a single approach (from the application perspective) can handle both the local and global cases, as well as transparently handle any needed change from one to the other. Under this approach, any reasonable selection for the layers below will be perfectly acceptable and work. These decisions can be handled locally on a case by case basis or by policy over some administrative domain if deemed appropriate. Building to

the level three interface admits a mixing and matching approach to all of the levels below without sacrificing interoperability. Levels above do need to be matched. However, in our immediate case, handling interoperability for these functional elements has already been subsumed into the current DIS PDU standard. This approach ensures the maximum interoperability with the minimum of specification and new development.

3.2.2 Generalized Functional Architecture.

The Communications community thinks in terms of a vertical layering of communications functions. The accepted nomenclature (adopted by the International Standards Organization) refers to seven layers. Table VIII identifies the levels and illustrates their meaning in the context of the networking of simulators.

TABLE VIII: Seven Layer OSI model applied to Simulation

Number	Name	Content
7	Application	Kind of data exchanged (position, orientation,...) Dead reckoning rules. Rules on determining hit or miss and damage.
6	Presentation	Representation of position (local vs geocentric coordinates), orientation (Euler angles, Quaternions, SPV), units (English, metric, degrees, BAMs..), and encoding (integer vs float, big vs little endian).
5	Session	Procedure for starting and ending an exercise. Rules for joining and leaving an exercise. Freeze.
4	Transport	Addressing from end user to end user. Assuring communications reliability, if required.
3	Network	Addressing information from node to node.
2	Link	Framing of information on a physical link. Flags, zero bit insertion. Conflict resolution.
1	Physical	Wire, optical fiber, radio transmission. Voltage levels, impedance values, clock rates.

The DIS PDU document addresses levels 5 through 7. It does so without separating the levels. Levels 4 and below are defined in the remainder of this section.

There are a variety of existing protocols and interfaces which populate the functional areas for levels 1-4. The two most prominent suites of protocols which are collectively put forth as solutions to the interoperability problem are the DoD (Internet) suite and the OSI (GOSIP) suite. At this stage of evolution, the two are conceptually similar, but vary considerably in the details and in maturity. Both suites emphasize the network transparency from level 3 and above, as discussed previously. This means that one simulator is completely isolated from the selections made at levels 1 and 2 for every other simulator or collection of simulators, by adopting one of the "internetwork" layer standards as the base level for interoperability. This provides the freedom to delegate to local decision making the protocols used for the lower levels (assuming the selections conform with overall, real time performance objectives). The current real work of this document focuses essentially on levels 3 and 4. A plan which starts from the more mature Internet suite and evolves as appropriate over time toward the GOSIP suite is the most prudent path at this time.

3.2.3 OSI Compatibility.

The ISORM was developed in 1977 by the International Organization for Standardization in response to the need to interconnect heterogeneous computers. OSI defines a framework for the interaction of users and applications in a distributed environment.

The Government Open Systems Interconnection Profile is the U.S. Government program for adoption of OSI across all Federal agencies. The purpose of GOSIP is to provide: networking connectivity, through GOSIP network architecture; interoperability, through standard "profiles" of OSI protocols; and competition, through focus on small number of subnetwork technologies and interoperable applications.

3.2.3.1 Benefits of DIS Compliance to the OSI/GOSIP Architecture.

DIS compliance with the OSI/GOSIP architecture provides the following benefits: reduced cost, increased interoperability, and increased application-level functionality. Efforts to ensure conformance to OSI/GOSIP standards and ensure interoperability between products of different vendors means that computer networking can be done as an integration of multi-vendor, commercial-off-the-shelf (COTS) components. Easy access to vendor interoperable COTS OSI/GOSIP products gives wider availability to networking capabilities at a reduced cost.

Not only will OSI/GOSIP standards provide interoperability between products, but international interoperability will also be increased. The OSI standards are international in scope and will be used by North Atlantic Treaty Organization (NATO) allies, among others. Using OSI standards opens the possibility that interoperation with our NATO allies will be accomplished within the framework of international standards.

3.2.4 PDU Encapsulation for Phase 0.

For the 1992 Interservice/Industry Training Systems Conference (I/ITSC), a demonstration of the use of DIS will occur. This demo will use the UDP/IP protocols for the communication architecture. The encapsulation of PDU in the UDP header was defined for the demonstration as shown below.

The UDP fields are defined as:

1 -	source port	(2 octets)	An optional field, when meaningful, indicates the port of the sending process.
2 -	destination port	(2 octets)	[DIS = 3000]
3 -	length	(2 octets)	Length of the datagram including the header and data.
4 -	checksum	(2 octets)	Verifies part of the IP header, the entire UDP header and data
5 -	data		DIS PDU data

Commonly used source and destination port numbers are available in RFC 1340 "Assigned Numbers". The number 3000 was chosen for DIS.

4. PERFORMANCE

4.1 Bandwidth

There are a number of factors which have a major influence on DIS bandwidth. At the very highest level, they include:

Total number of entities
Mixture of entity types.
Type of exercise or scenario
Choice of dead reckoning algorithm (and positional/angular thresholds)
Security requirements

For the current set of approved DIS PDUs, the majority of network traffic will be Entity State PDUs (ESPDUs). ESPDUs are required to be sent at some minimum rate (e.g. every 5 seconds) by every entity and may be sent much more frequently depending on entity dynamics. The start-up of a session will also see high traffic but that is deterministic. The PDUs used to initialize an exercise or entity (such as the recommended Activate PDUs) represent a significant amount of data to be sent via the net, but they can be transmitted at a controlled rate. In the near term, the inclusion of Emitter PDUs may add a significant traffic load to the network, depending on the degree of electronic warfare (EW) present in a given exercise. Similarly the future inclusion of simulated tactical communication links (both voice and data) will undoubtedly have a substantial impact on bandwidth.

In addition to the above there are also additional bandwidth requirements due to communications "overhead". A given PDU of "n" bits in length requires the addition of both headers and trailers in order to satisfy routing and data integrity requirements. The proposed UDP/IP protocols add 28 octets (8 for UDP and 20 for IP). The underlying media adds further overhead, such as FDDI's 20 to 28 octets of preamble, header and trailer information. A method to reduce this load is to concatenate PDUs at the application layer such that the overhead bits are applied to groups of PDUs rather than to every PDU. This approach, however, imposes an additional computational load on each host. This trade-off of processing load vs network traffic requires further study before serious recommendations can be made.

Another source of "overhead" traffic are security measures. The degree of overhead depends on at what layer (of the OSI seven layer stack) the security measures are implemented.

4.1.1 Estimating Exercise Bandwidth Requirements.

In general, there is no single set of formulae for accurately estimating the bandwidth requirements of any given DIS exercise since, by nature, they have a combination of man-in-the-loop and non-deterministic simulated adversaries. As such, each entity in a given exercise generates network traffic at a varying rate. The rate varies depending on the particular involvement of that entity with others. For example any vehicle that is in transit to or from its assigned duty area will exhibit very predictable dynamics and therefore generate low network traffic. Conversely,

an entity entering into conflict or close cooperation with another will typically generate a high level of traffic. In both cases the traffic is a result of the frequency at which the PDUs are generated, while the size of the individual PDUs remain relatively stable. Estimating sizes of PDUs for selected entity types is a comparatively straightforward process while estimating the frequency at which they are generated is fairly complex and more subjective.

As stated earlier the Entity State PDU will be the main source of network traffic. There are currently nine other PDU types required by the DIS standard, with several others recommended. Of the nine required, six are related to logistics (e.g. repair and resupply) and are expected to occur so infrequently as to have little or no effect on network bandwidth requirements. Another, the Collision PDU, also falls into this category. The remaining two are the Fire PDU (FPDU) and Detonation PDU (DPDU), and conceivably can occur frequently enough at certain stages of battle to be considered in bandwidth calculations. In addition, the Emitter PDU (EPDU), one of the emerging recommended messages, is likely to be a major contributor in the near future. These four PDU types have the following formula for determining their sizes (in bits):

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

Given the above, it is possible to estimate the PDU sizes for classes of entity types. For example, for a given type of tank the minimum number of articulated part records may be 5 (azimuth and azimuth rate for turret, elevation of the barrel, and up/down position for two hatches) and the number of emitters 1 (laser range finder). For a fighter aircraft the number of articulated parts could easily be 20 (8 weapon stations, 2 drop tank stations, 6 vertical control surfaces, 2 horizontal control surfaces, landing gear, and speed brake) with 3 emitters (radar, jammer, and laser designator). Similar assumptions can be made regarding surface ships. The following table presents estimates of PDU sizing for these three classes of entities (without any overhead bits).

TABLE VI. PDU Sizing Estimates

ENTITY CLASS	A	H	E	B	T	ESPDU	FPDU	DPDU	EPDU
TANK	5	1	1	1	1	1792	704	928	752
AIRCRAFT	20	2	3	1	2	3712	704	1056	2160
SURFACE SHIP	50	5	10	1	5	7552	704	1440	9632

The next step in estimating the bandwidth requirements of a given exercise is to approximate the rates at which each entity class will issue each of the above PDU types. Since this rate can vary a great deal within a given exercise, one method of estimation is to give values representing some average low and high rates. The final step is to determine the number of each major entity type which will participate in the exercise. Given all of these factors, the determination of a range of probable network traffic can be easily calculated. Figure 7 presents an example of such an analysis for three different types of exercises. The examples include tactical voice and data links as sources of network traffic (65 Kbs for each voice channel and actual values for Link-4A, Link-11, and Link-16). Figure 8 presents the results of the same analysis in graphical format.

SAMPLE PDU SIZING

	A	H	E	B	T	ESPDU	FPDU	DPDU	EPDU
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
						OVERHEAD BITS/PDU=			428

SAMPLE RATES PER ENTITY TYPE PER PDU TYPE

	LOW RATE (HZ)				HIGH RATE (HZ)			
	ESPDU	FPDU	DPDU	EPDU	ESPDU	FPDU	DPDU	EPDU
TANK	0.2	0	0	0.2	2	0.1	0.1	1
AIRCRAFT	0.2	0	0	0.2	8	0.1	0.1	4
SURFACE SHIP	0.2	0	0	0.2	1	0.1	0.1	2

SAMPLE EXERCISE TRAFFIC ESTIMATES

% ENTITIES AT HIGH RATE	0%	20%	40%	60%	80%	100%
% ENTITIES AT LOW RATE	100%	80%	60%	40%	20%	0%
SAMPLE EXERCISE #1						
600 TANKS	408,000	1,030,656	1,653,312	2,275,968	2,898,624	3,521,280
100 AIRCRAFT	134,560	982,320	1,830,080	2,677,840	3,525,600	4,373,360
10 TACTICAL VOICE LINKS	650,000	650,000	650,000	650,000	650,000	650,000
TOTAL TRAFFIC	*1,192,560*	*2,662,976*	*4,133,392*	*5,603,808*	*7,074,224*	*8,544,640*
SAMPLE EXERCISE #2						
24 SHIPS	84,538	201,896	319,254	436,612	553,970	671,328
50 AIRCRAFT	67,280	491,160	915,040	1,338,920	1,762,800	2,186,680
3 TACTICAL DATA LINKS	237,500	237,500	237,500	237,500	237,500	237,500
TOTAL TRAFFIC	*389,318*	*930,556*	*1,471,794*	*2,013,032*	*2,554,270*	*3,095,508*
SAMPLE EXERCISE #3						
200 TANKS	136,000	343,552	551,104	758,656	966,208	1,173,760
100 AIRCRAFT	134,560	982,320	1,830,080	2,677,840	3,525,600	4,373,360
5 TACTICAL VOICE LINKS	325,000	325,000	325,000	325,000	325,000	325,000
3 TACTICAL DATA LINKS	237,500	237,500	237,500	237,500	237,500	237,500
TOTAL TRAFFIC	*833,060*	*1,888,372*	*2,943,684*	*3,998,996*	*5,054,308*	*6,109,620*

Figure 7. Sample Network Traffic Analysis

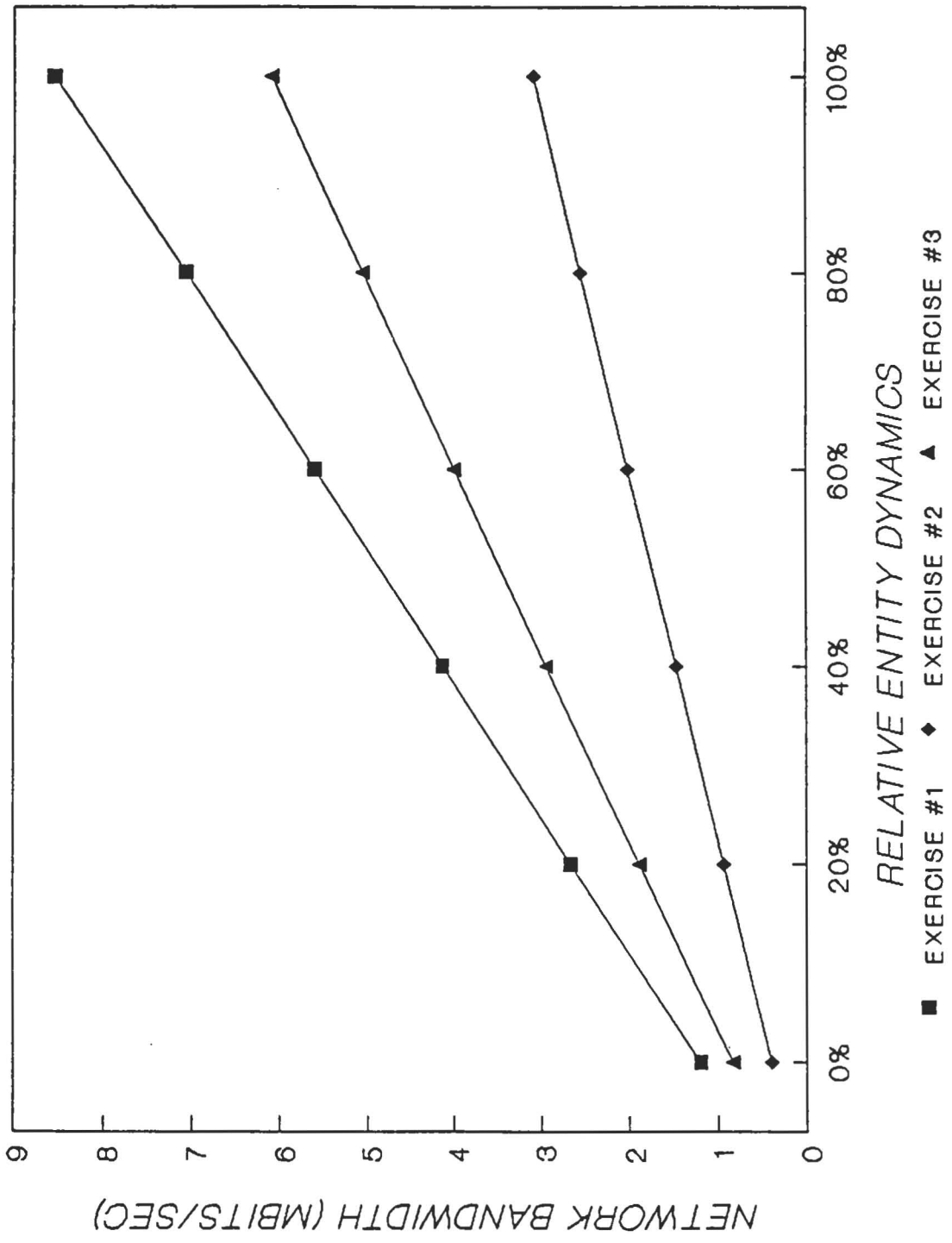

Figure 8. Sample Exercise Bandwidth

As shown in the figures, the network traffic can vary as much as ten to one depending on the relative number of entities which are in a high dynamic environment. The low end of the charts are certainly the minimum bandwidth requirements since they are based on all entities in a quiescent mode (i.e. ESPDUs only once every 5 seconds). The high ends of the charts are more subjective since it makes assumptions as to the maximum rates each entity type will exhibit, but in any case are not probable since they represent all entities simultaneously engaged in heavy combat. Given those assumptions, such charts may be used as a guide to sizing a network for any type of exercise.

Some final points to be made about the above discussion:

- The sample bandwidth values shown are only for illustration, and should not be used in formal specifications.

- The Emitter PDU used here is in accordance with the latest format proposed by the Emissions Subgroup, not the format shown in the existing version of the DIS specification. This latest version results in less overall network traffic since it is only issued on change of the emitter data (the older version had to be issued at least as often as ESPDUs).

- The analysis does not account for the transitory existence of entities in the form of guided weapons released by various types of weapon systems. These will add still further traffic and will most likely be present during the same period of time where high vehicle dynamics are also occurring - during engagement of groups of opposing forces.

- No data compression is assumed. For reduction of PDU traffic it is not considered viable at this time due to the large computational load it would place upon each entity host computer. It should be seriously considered for tactical voice links, however, since the task is simplified by the fact that the computer does not need to know what is actually in a voice message; the compression and decompression can then be done by hardware, external to the computer system. The signal can be compressed by hardware at the source, sent over the network in its compressed form, and fed directly to decompression hardware at the listener. A variety of commercial devices currently exist to support this, some offering time stamping of the audio stream for synchronization. Standards are emerging with the growth of multimedia computing technology, and could be considered for use in the DIS application.

4.1.2 Estimating Traffic in terms of PDUs and Packets per Second.

Once it has been established that the underlying media is capable of handling network traffic (i.e. from the bits per second standpoint), the next figure of merit to analyze is that of the number of messages to be handled in a given unit of time. This factor provides a relative figure-of-merit for the type of processing power necessary for a given set of communications protocols.

Figure 9 presents another look at the sample exercise data presented earlier. Here, in addition to the total traffic for each exercise in bits per second there are two additional fields showing the number of PDUs per second as well as packets per second. The following assumptions were made in developing these estimates:

1. Packet length is the standard 1500 octet Ethernet datagram size.

2. PDUs can be concatenated such that each packet contains several PDUs. The PDU sizes here are taken to be without overhead bits; a single set of overhead bits is applied to the entire packet.

3. The "host" composing the packets always waits until the 1500 octet limit is filled. In actual practice the efficiency factor will probably be lower (to avoid excessive latency), resulting in an actual packet rate that falls somewhere between the two values (PDUs/sec and packets/sec) shown.

4. Voice packets are produced at 32 Hz, Link-11 and Link-4A at 4 Hz, and Link-16 (JTIDS) at 16 Hz.

SAMPLE PDU SIZING

	A	H	E	B	T	ESPDU	FPDU	DPDU	EPDU
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
						OVERHEAD BITS/PDU=			428

SAMPLE RATES PER ENTITY TYPE PER PDU TYPE

	LOW RATE				HIGH RATE			
	ESPDU	FPDU	DPDU	EPDU	ESPDU	FPDU	DPDU	EPDU
TANK	0.2	0	0	0.2	2	0.1	0.1	1
AIRCRAFT	0.2	0	0	0.2	8	0.1	0.1	4
SURFACE SHIP	0.2	0	0	0.2	1	0.1	0.1	2

SAMPLE EXERCISE TRAFFIC ESTIMATES

% ENTITIES AT HIGH RATE	0%	20%	40%	60%	80%	100%
% ENTITIES AT LOW RATE	100%	80%	60%	40%	20%	0%
SAMPLE EXERCISE #1						
600 TANKS	408,000	1,030,656	1,653,312	2,275,968	2,898,624	3,521,280
100 AIRCRAFT	134,560	982,320	1,830,080	2,677,840	3,525,600	4,373,360
10 TACTICAL VOICE LINKS	650,000	650,000	650,000	650,000	650,000	650,000
TOTAL TRAFFIC	*1,192,560*	*2,662,976*	*4,133,392*	*5,603,808*	*7,074,224*	*8,544,640*
PDUs/SEC	*600*	*1172*	*1744*	*2316*	*2888*	*3460*
PACKETS/SEC	*81*	*187*	*293*	*399*	*505*	*610*
SAMPLE EXERCISE #2						
24 SHIPS	84,538	201,896	319,254	436,612	553,970	671,328
50 AIRCRAFT	67,280	491,160	915,040	1,338,920	1,762,800	2,186,680
3 TACTICAL DATA LINKS	237,500	237,500	237,500	237,500	237,500	237,500
TOTAL TRAFFIC	*389,318*	*930,556*	*1,471,794*	*2,013,032*	*2,554,270*	*3,095,508*
PDUs/SEC	*54*	*185*	*316*	*448*	*579*	*711*
PACKETS/SEC	*32*	*74*	*115*	*157*	*199*	*241*
SAMPLE EXERCISE #3						
200 TANKS	136,000	343,552	551,104	758,656	966,208	1,173,760
100 AIRCRAFT	134,560	982,320	1,830,080	2,677,840	3,525,600	4,373,360
5 TACTICAL VOICE LINKS	325,000	325,000	325,000	325,000	325,000	325,000
3 TACTICAL DATA LINKS	237,500	237,500	237,500	237,500	237,500	237,500
TOTAL TRAFFIC	*833,060*	*1,888,372*	*2,943,684*	*3,998,996*	*5,054,308*	*6,109,620*
PDUs/SEC	*304*	*652*	*1000*	*1348*	*1696*	*2044*
PACKETS/SEC	*61*	*139*	*217*	*296*	*374*	*452*

Figure 9. Analysis with Bits, PDUs, and Packets/Sec

4.2 Latency

Some interactions between simulated entities are very tightly coupled in time. That is, the action of an individual controlling one of the entities may be a reaction to the activity of another. How tightly these interactions are coupled in time depends on the performance of the unit being controlled. High performance units, that is those units that react quickly to a human controllers input, tend to be very tightly coupled. An example of this is one simulated fighter aircraft flying in close formation with another. Units that respond to control inputs less quickly, such as ships, are only loosely coupled.

The issue of communications latency is directly related to how tightly a simulated entity is coupled to the entity to which it is reacting. The more tightly coupled two simulated entities are, the less latency is permitted in the communications that carry the state data of each to the other. Allowable latency under different circumstances is the subject of considerable debate. Little research of the quality that can serve as the basis of standards for latency has been done. The best information available is from the flight simulator industry, which for many years has been struggling with a related issue called transport delay. Flight simulator experience provides the following:

1. Humans cannot distinguish differences in time that are less than 100 milliseconds. This is due to physiological factors of the human body. This effectively provides a floor latency/transport delay value. That is, with a human in the control loop, there is no benefit to be gained from latency/transport delay less than 100 milliseconds.

2. In situations where latency/transport delay reaches 300 milliseconds, pilots start compensating for the lag in response. The result is a phenomenon known as Pilot Induced Oscillation (PIO). Such PIO can range from a minor annoyance to total loss of control.

The flight simulation community has also experimented with schemes to compensate for transport delay by predicting the behavior of the device being controlled. This approach showed promise, but the main emphasis in dealing with transport delay has been in reducing the delay by faster processing and better communications within the simulator. The DIS community has also begun to explore prediction of position as a means to compensate for latency in tightly coupled interaction. Northrop has done the most work in this area. Studies reported to the DIS

community[2] suggest that sophisticated prediction algorithms can compensate for up to 750 milliseconds of latency in the interaction of high performance aircraft carrying out radical maneuvers.

The position of simulated vehicles is not the only consideration in dealing with latency. DIS networks will also carry voice in the simulation of tactical radio nets. A speaker's voice will be converted from analog to a digital data stream that will be treated as just another series of PDUs. At the listener's position these will be converted back into analog form and will be output to speakers and/or headphones. Latency in such voice communications carries its own considerations.

In the case of an overseas phone call that was routed via a geosynchronous satellite, latency of a half second or more is inherent in such communication. In normal conversation this is annoying but the speakers can generally adjust to it without difficulty. However, in the heat of a simulated operation such delays would render a simulated radio net unusable and would not be acceptable. Also, there is no prediction mechanism that can compensate for delays in voice traffic.

The dispersion of the arrival times of voice PDUs is also important. In the process of converting analog voice to a digital data stream, the analog signal is sampled at regular intervals and each sample is converted to a digital message. For the reconstruction of the voice back to analog these messages should ideally arrive at the same regular interval. However, due to a variety of factors, there will some dispersion of arrival times. If the dispersion is too great, voice quality will suffer and may be unintelligible. The mechanism of converting voice from digital to analog form can handle some dispersion in arrival times. It is also possible to deliberately hold incoming voice PDUs in an accumulating FIFO buffer and then meter them to the voice reconstruction mechanism at a same rate at which the voice was sampled. This technique would eliminate the effects of delay dispersion, but would do so at the cost of additional overall latency.

4.2.1 Allocation of Latency Values.

In designing systems that meet the total latency standards defined in the CADIS standard, it is important to allocate these latencies in a reasonable manner. For example, in a LAN if one designs a simulator with a latency of 45 milliseconds between the

---

[2] Position paper "Techniques for Extrapolation, Delay Compensation, and Smoothing with Preliminary Results and an Evaluation Tool," S. Goel, K. Morris, IST-CR-91-13, Summary Report: The Fifth Workshop on Standards for the Interoperability of Defense Simulations.

application layer and the media (layer 1) it will still meet the standard of 100 milliseconds for total latency with similar simulators on the same LAN. However, if it becomes part of an exercise that includes simulators from other geographic sites, the total latency will likely exceed 100 milliseconds due to the latency consumed by the WAN connecting the sites.

5. SECURITY

5.1 Introduction.

The goal of this section is to identify a number of security requirements made evident by the broad outlines of DIS, and by common understanding regarding the environment in which DIS will perform. The section will also give a thumbnail sketch of the world of information security. This section is not intended to be a comprehensive analysis of DIS security requirements. Such requirements will be a complex function of the system itself as it evolves, and of the needs of its primary intended users.

5.2 Policy.

As an Automated Information Systems (AIS), networks supporting DIS must comply with appropriate security criteria to be certified and accredited to process unclassified sensitive and classified information. The criteria encompass a wide range of security issues that impact the AIS or network. Security is usually achieved by a combination of software and hardware functions, administrative procedures, personnel clearances, and physical measures. The Designated Approving Authority (DAA) determines the required balance of automated functions and manual procedures in accordance with risk management decisions.

AIS security is an operational requirement and requires detailed planning and execution to a degree equal to or grater than any other operational requirement. Security shall be considered throughout the life cycle of an AIS of network from the beginning of concept development, though design, development, operation, and maintenance. The program manager and system developer must take steps to ensure that security considerations are addressed in each of the above referenced phases of the system life cycle.

5.2.1 Security Plan.

Security Plans for DIS networks will be based on guidance from appropriate DoD Component Heads. Security plans shall be prepared by the appropriate Information System Security Organization (ISSO) or Network Security Officer (NSO)

5.3 Security Vocabulary.

Arguably the most important task in defining a security

specification for DIS is the acceptance of a common vocabulary for discussing security issues. The most widely accepted system so far developed is the DoD Trusted Computer Systems Evaluation Criteria (TCSEC), and its follow on "interpretations" for Networking, Secure Database Standards, and Integrity Criteria. TCSEC is also known as the Orange Book, while the Trusted Network Interpretation (TNI) is known as the Red Book. These works have both popularized and made explicit such terms as "Security Policy", "Multilevel Security", "Discretionary Access Control", "Trusted Path", etc., as well as the familiar rating categories C1, C2, B1, B2, B3, A1.

The National Computer Security Center (NCSC) evaluates commercial products.

There are many good reasons for using the DoD security vocabulary. For one thing, it is fairly explicit, and addresses, in one form or another, virtually every conceivable aspect of computer security. It is not necessary to commit to the evaluation categories, or to specific formulas of risk assessment, to benefit from the vocabulary, concepts, and methodologies which have been developed. In addition, the primary clients for DIS will, at least initially, come from the DoD, and the classified nature of information exchanged on distributed simulation nets makes the DoD approach appropriate.

5.4 DIS Security Requirements.

A comprehensive list of DIS security requirements is not available, nor is there one in preparation. Yet certain specific security needs are already discernible. It is the responsibility of the network sponsor to describe the overall network security policy enforced by the Network Trusted Computing Base (NTCB). At a minimum, this policy shall include the discretionary and mandatory integrity, or both. The policy may require data secrecy, or data integrity, or both. It is essential that development of the discretionary and mandatory secrecy policy be addressed as an integral part of network design. Some of the elements that support the security policy are described briefly in the remainder of this section. The elements are merely examples, development of a security policy and security appliques for specific DIS application requires support from information security specialist within a given organization or command and may also require support from INFOSEC specialist from the National Security Agency's (NSA) Information Systems Security Organization.

5.4.1 Encryption

5.4.1.1 Confidentiality Requirements.

It is known that messages exchanged during a military simulation

will contain sensitive data regarding weapons systems characteristics and warfare tactics. A DIS exercise may also be the rehearsal of an operational mission, and as such the data exchanged will be extremely sensitive. Clearly such information must be protected from eavesdropping by simulation non-participants, much in the same manner as telemetry data is protected. Eavesdropping can occur via wiretapping, which is monitoring by entities not legitimately connected to the net, or by users who are legitimately connected but are accessing message data not intended or them.

A mechanism for thwarting eavesdroppers is encryption of messages on the network. The architectural level at which encryption/decryption occurs is significant: encryption at the link level (L2) is more efficient, while encryption at the session layer or higher (L5+) allows users to be differentiated by different encryption keys, and protects messages for a greater part of their passage through the operating system of the host. Encryption is used for other tasks as well, in particular the authentication of user identities Identification and authentication are covered in section 5.4.3 below).

5.4.1.2 Key Distribution.

Assigning and distributing cryptographic keys on a dynamic or per-session basis can be a major difficulty. However, the Joint Chiefs Of Staff (JCS) issued a Multicommand Required Operational Capability (MROC) for a Joint Key Management System (JKMS) on 28 December 1989. Further, the criticality of interoperability through electronic key distribution was underscored by joint operations in DESERT STORM, and consequently enjoys a high priority. When fielded, electronic key management will eliminate the requirement to physically deliver keying material to each DIS facility. Products such as CANEWARE and NEW are already capable of accepting electronically distributed key.

5.4.1.3 DIS Encryption.

Nodes on a DIS network will not transmit a great deal of data, but will receive data from all the other nodes in the simulation. Thus a fast algorithm is required, if only on the decoding end.

Fiber Data Distribution Interface (FDDI) fiber optics are relatively safe from wiretapping, so the need for encryption on a FDDI ring is reduced on a LAN if its configuration meets the criteria of a protected distribution system (PDS). On a LAN supporting multilevel security, encryption may be required to prevent eavesdropping by legitimately connected FDDI hosts; likewise there is an eventual need for session level isolation and access control in multi-user application gateways. Wide Area Networks (WAN) employing FDDI will require encryption due to the wiretap threat. At present, encryption systems for the 100Mb

plus data rates are under development but not currently available. Fortunately, in the short term, only single simulations will run on the DIS net, and nonparticipants can be physically excluded; thus link level encryption for FDDI can await the emergence of a suitably fast technology.

5.4.2 Access Control Issues.

DIS will support multiple simulations simultaneously on a single network. Enforcing the separation of simulations becomes a security issue when differing classification levels coexist, as, for example, when a highly classified weapons development simulation is run together with a simulated battle scenario, presumably at a lower classification level.

DIS security issues go far beyond the protection of run-time simulation messages. Computers that participate in more than one level of exercise will be required to store and to internally manipulate data of varying classifications, and to insure that only users with proper clearance can access classified data. This raises issues of Multilevel Security at both the Operating System and Database levels. The following sections discuss these issues in more detail.

5.4.2.1 Label-based Access Control Mechanisms.

Label-based security is an important requirement in the DoD TCSEC at the B1 and higher levels of assurance. The mechanisms are called Mandatory Access Controls (MAC) because data transfer is governed, in part, by the contents of subject and object sensitivity labels.

Multilevel Security is implemented by defining a class of protected data objects, and attaching security sensitivity labels to them. Autonomous entities (users and processes) are known as subjects; these also receive sensitivity labels by which their access to the protected objects is regulated. The set of subjects and objects, together with the rules for access, is carefully specified in a set of rules. The enforcement mechanism for the rules is referred to as the Reference Monitor.

The Bell-Lapadula Model is a Security Policy associated with the TCSEC. It specifies important read and write controls so that classified information cannot flow in violation of national security directives. The Bell-LaPadula model is the most widely accepted and implemented access control model used in the DoD.

5.4.3 Identity and Authentication.

In a distributed interactive simulation, it is important to guarantee that participants are, in fact, who they say they are; this is known as the Authentication problem. Identification of

entities can occur at varying levels of granularity: the level of host on a network, the level of human users on the network, or the identifications of individual processes. In the initial DIS environment, simulation hosts will participate in only one simulation at a time; it seems reasonable, therefore, to initially propose a per-node granularity of authentication.

5.4.4 Integrity.

In DIS, as in most environments, there is the need to insure that data is not corrupted, either deliberately or by accident. This issue of Integrity is an important security problem, and applies to message data, stored information, and dynamically manipulated information within an operating system. Again, cryptography plays an important role in data integrity verification (for example by checksums), and many network authentication services also support point-to-point integrity policies.

5.4.5 Audit.

A critical facility for all secure systems, including networks, is the audit facility. The audit facility maintains logs of security-relevant events in tamperproof, restricted access locations; typical examples of logged events include attempted logins and access to critical data. Commercial audit products exist.

Audit trails can be maintained on individual systems, but a network audit facility is also desirable in DIS. Coordinating a distributed audit facility can be a problem, and might require utilities like NFS and yellow pages (secure versions of which are currently under development). The main problem with audit is storing and analyzing the enormous amount of data that can be generated. The primary approach to this problem is to specify a limited set of audit events; this greatly reduces the data volume. Many audit systems will have built in "triggers", or thresholds, that expand the level of audit detail in areas where certain conditions have been exceeded. Likewise, there are processing tools to make the analysis of audit data easier, should that prove necessary.

5.4.6 Security Architecture.

Approaches to network security are dictated by a number of factors such as data rates, vulnerability, threat and availability of encryption devises. The DIS Security Models section describes a security framework and security architectures that are usable and compatible with the DIS architecture.

5.4.7 Physical Security.

Physical security consists of functions that can be performed by

"physical" mechanisms, i.e., those that are not part of the computer operating system. A list of examples might include:

1) Protected cables, locked rooms, security guards, removable media

2) Computer locks, disk drive locks, hardening against radiation leakage & radiation damage

Physical security can come from unexpected sources, for example fiber-optic networks, which almost as a bonus provide several measures of physical security. Fiber optic networks 1) are difficult to tap undetected, 2) immune to EMR damage, and 3) do not leak EMR that can be monitored.

In general, however, the methods of physical security lie outside our scope of interest.

5.5 Security Products.

A list of certified network products can be found in the Information Systems Security Products and Services Catalogue, published by the NSA. Additional information in Information Systems Security products and services may be obtained by writing to:

Director
National Security Agency
ATTN: INFOSEC Office of customer Relations
Fort George G. Meade, MD 20755-6000

or calling:

Customer Relations at (301)688-4680.

5.6 DIS Security Models **(to be added)**

5.6.1 Case 1: Single Cell - System High

5.6.2 Case 2: Multiple Cells - Same Security Level

5.6.3 Case 3: Multiple Cells - Different Security Levels

5.7 Conclusion.

The security situation for DIS is complicated by the desire for standards and interoperability, a real dearth of available products, and the inherent vulnerabilities of a distributed architecture. Implementors of critical features, such as networking, operating systems, and database security will have to confront major systems integration and standards-conformance problems. At the same time, the classified environments in which

DIS must operate will make adherence to formal standards of evaluation and certification more critical than in commercial environments.

6. NETWORK MANAGEMENT

6.1 Network Management.

The approach to network management is usually dependent upon the type of network employed. Thus, there is a generally recognized and sanctioned way to manage an OSI based network, in the form of OSI network management protocols and service definitions. Similarly, an Internet based network is typically managed by Internet network management protocols. The most prudent course of action would be to adopt the network management approach that comes with the protocol suite selected for handling interoperability.

6.1.1 Basic Functions.

Exercise communications management is a set of facilities to monitor and control the networks that join simulators and other DIS components at a site and sites with each other. Monitoring shall mean the ability to determine the status of a network component. Control shall mean the ability to set parameters of a network component. The monitoring and control of network components is often referred to as "network management".

DIS requirements for network management are essentially the same as for any other distributed application. One can think of an exercise as having two phases, initialization and operation. During the initialization phase, one would use network management monitoring facilities to check the status of lines, host interfaces, routers, and other network components required for the exercise. Control functions would be used to boot devices with the appropriate parameters, enable interfaces, and so on. The exact set of functions used would depend on the equipment being used, the extent to which its configuration can be changed, and the nature of the network or networks involved.

During operation of an exercise, network management functions would be used to detect and troubleshoot problems. Monitoring functions are used to detect apparent connectivity or equipment failures. Once a problem is detected, operators select appropriate monitoring functions to retrieve parameter values or other information needed to determine the exact cause. Finally, operators can use control functions to reboot equipment, activate alternate interfaces, or take other corrective action. As is the case for initialization, the exact functions used would depend on the nature of the problem, the equipment, and of the networks involved.

It should be noted that some facets of network operation are not typically automated or performed remotely. For example, a network operator might command the use of a dial-up line, but the use of leased lines must typically be arranged for in advance. Also, while a network operator might command the use of back-up equipment when primary equipment fails, it is sometimes necessary for a technician to remove and replace failed components.

6.1.2 Network Management Architecture

DIS shall use standard network management protocols to manage the communications infrastructure. Simple Network Management Protocol (SNMP) is a network management protocol frequently used in conjunction with the Transmission Control Protocol (TCP)/Internet Protocol (IP) stack. Common Management Information Services (CMIS)/Common Management Information Protocol (CMIP) is used in an OSI environment. The choice of network management protocol would depend on the other protocol suites (i.e. Internet or OSI) being used in the network.

With the phased migration from UDP/IP to OSI recommended by CASS, the choice of Network Management protocol would intuitively progress from SNMP to CMIS/CMIP. Some of the respective architectural features are listed below.

In the SNMP architecture, there are a number of Network Management Stations (NMS's) which gather pertinent information by communicating with Management Agents associated with each Network Element such as hosts, terminal servers, and gateways.

SNMP's support seems to be widespread and growing. There are, however, some perceived limitations

1. Poor communications between SNMP Network Management Stations.

2. There are security weaknesses, such as the lack of authentication of Set commands.

3. SNMP does not handle sub-element addressing. For example, to get at info about port #5 of a multi-port router, the net manager must go through a long process of repeatedly querying data for every instance of a variable

Work is currently ongoing to correct these problems. A proposa' to help resolve this problem will include block transfers of MIB data. The proposed revision will be called SMP.

Problems which must be addressed are information hiding; for security reasons, for example, not all network management systems should be allowed access to the complete MIB of all elements.

CMIP, like OSI in general, is not widely supported and, in fact, not totally proven. One criticism is that it uses all 7 layers of the OSI stack and consumes too much processing power. To cut the processing requirement, CMOP and CMOL operate over TCP/IP and link level respectively. These protocols are not necessarily OSI-compliant and are not widely supported. On the positive side, SNMP will run over OSI networks.

6.2 Network Management Functions

The following sections are functions that should be provided by Network Management. Concern was given that these functions will all be impacted by the decision on whether security is managed on a "cell" basis where all physical simulators and simulators are statically bound, or managed based on a conceptual "exercise".

6.2.1 Define or choose mechanism to promulgate security level of exercise.

Issues for this function are:

1. What is the system's security lattice, i.e., what are the values of the security sensitivity labels that could legally be associated with any data in the exercise, and what are the dominance relationships when comparing any two with each other?

2. What is the granularity of security sensitivity labeling?

6.2.2 Define the mapping between classified and unclassified databases.

Issues for this function are:

1. What is the granularity of security sensitivity labeling of the database, e.g., none; table-level; row-level; record-level?

2. Who is to have access to the classified and unclassified tables?

3. What are the clearance levels of those who are to have access to the tables?

6.2.3 Enumerate all hosts participating in the exercise.

Issues for this function are:

1. What is the security operating mode of the host?

2. What is the security lattice subset supported by the host?

3. What are the security characteristics associated with hosts with which any given host wishes to communicate?

6.2.3.1 Enumerate security sensitivity level of all participating hosts (in the exercise).

6.2.4 Provide a mechanism to select & distribute keying material as needed.

Security issues for this function are:

1. Centralized or distributed?
2. Manual or automated?
3. If automated, accredited?
4. What is the security policy regarding key distribution and change?

6.2.5 Choose address or addresses to be used in exercise

Should the security characteristics (security operating mode, maximum security level) of an addressee be supplied with the address?

6.2.6 Allocate bandwidth appropriately

The philosophical design permits simulators to come and go during an exercise which may last for several days. The following issues arise:

1. What bandwidth is reserved in advance ?

2. How are simulators admitted into an exercise with a high degree of confidence that performance will remain acceptable? Does the exercise management function include mechanisms for honoring or rejecting a request to enter ?

Are there run-time provisions for sensing cases in which one network element is "hogging" the medium ? It might be difficult to call a break to hunt down such a problem.

6.2.7 Use network time protocol (NTP) rather than new PDU's for time

Issues for the following function are:

1. On a dynamic network such as this, could an NTP-base synchronization subnet be properly designed ?

2. Considering the sizable investments at most training sites and the fact that PC and VME- based GPS receivers can be purchased for $5K, might it be better to sync each site to GPS ? If each site has a GPS receiver, should NTP be used on the LAN ? In a LAN, the time server simply broadcasts

3. The NTP time stamp uses 64-bit fixed-point timestamp with integer in first 32-bits and fraction in next 32 bits. Is this appropriate for all the various entities including field instrumentation entities ?

7. REFERENCES

Listed below are some of the documents referenced in this rationale document.

7.1 Standards Referenced

The following standards have been referenced in this document:

a. ISO 7498 and CCITT X.200 (ISO Reference Model).

b. Mil Std. Final Draft Protocol Data Units for Entity Information and Entity Interaction in a Distributed Interactive Simulation, October 1991.

7.2 Other Documents Referenced

The following non-standard documents have been referenced in this document:

a. Tannenbaum, Computer Networks. Prentice Hall, 1988.

APPENDICES

APPENDIX A

REPRESENTATIVE PROFILES

10 Communication Architecture Profile For Phase 0

The following is one example of how a communications profile might set up for a DIS exercise. This communication architecture uses the DoD family of protocols is based on IP, with TCP (MIL-STD-1778) for reliable communication and UDP for real-time communication.

10.1 Exercise Management. In each simulation site there are several simulators and a Local Exercise Manager (LEM), all interconnected by a LAN, to which we refer as "Ethernet", even though it may be FDDI or other LANs.

The LEM is a software module, which does not need dedicated hardware, and may be implemented on any of the simulators, for example. The LEM is in communication with other LEMs, and in particular with the Global Exercise Manager (GEM) for the purpose of coordinating the entire exercise.

After the LEMs agree about the parameters of an exercise they communicate them to all the participating simulators, using a "session-level" type communication. This setup includes the identifications of all the simulators involved in the exercise, their roles, the exact presentation schemes to be used, the exact geographic database to be used, the maximal bandwidth that each simulator is allowed to load on the network, and the IP-multicast-addresses (IPMCA) assigned to the entire exercise.

10.2 Communication Setup. The setup communication, between the LEM and the simulators is conducted by using Telnet over TCP (over IP, over Ethernet). The setup process may use both manual and automatic procedures. As a part of the general setup, database files (e.g., geographic) are loaded, by using FTP (MIL-STD-1780), from designated directories. FTP also operates over TCP (over IP, over the Ethernet). The real-time communication (e.g., of PDUs) is carried by UDP. These packets are encapsulated inside Ethernet packets. The entire configuration is managed (and verified) by using SNMP in the simulators. This allows a remote network management process to check the status of each simulator.

In each case the real-time simulation messages are sent to all the participants in the exercise, local broadcast over the Ethernet, and remote multicast over WANs.

It is expected that future simulators may require time synchronization. This may be achieved by using the Internet time synchronization protocol (a.k.a. the Network Time Protocol, NTP), over UDP, on the Ethernet. The time protocol is defined in RFC1119 and RFC1129.

The real-time communication for the support of distributed interactive simulation requires that a given bandwidth is delivered without exceeding a given delay. In practice this required both multicast and bandwidth performance guarantees. Since these issues (bandwidth+delay and multicast) are at the network level (level-3 of the ISORM) it is possible to address them at the gateway between the LAN and the WAN. If the WAN provide these services there is no need for this gateway to be involved. However, in the most general case this gateway should handle them. In cases that the Commercial Off The Shelf (COTS) gateways and WANs do not provide this functionality it can be achieved it by adding a front-end to the gateway on a general purpose computer, preferably with two Ethernet interfaces to allow inserting this front-end "in series" with the gateway.

To guarantee interoperability each simulator should comply with the Host-Requirement, as specified in "Requirements for Internet hosts - communication layers" (RFC1122) and in "Requirements for Internet hosts - application and support" (RFC1123). This would guarantee the "invisible support" as required for interoperability (including ARP, re-direct, etc.). A good source of information is "Perspective on the Host Requirements RFCs" (RFC1127).

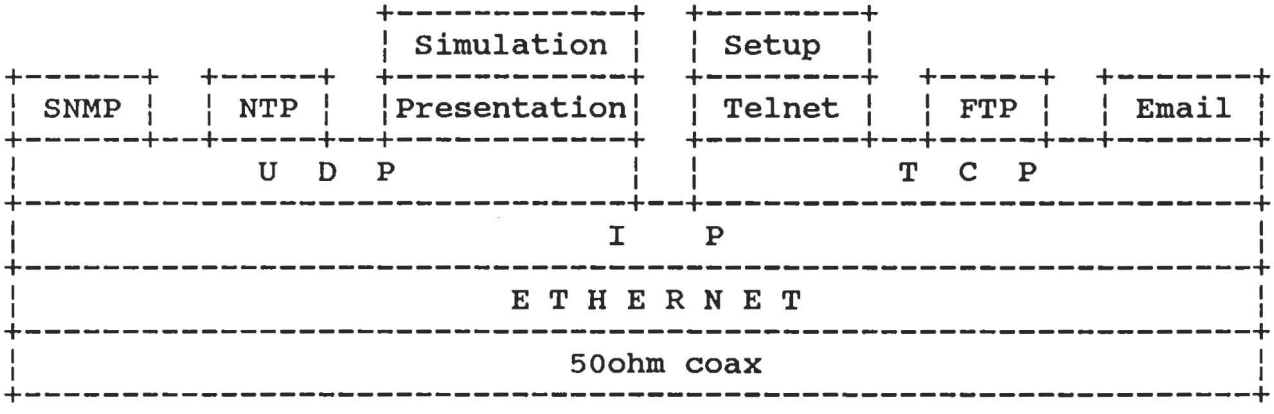

Figure 10. The Protocol Structure in the Simulators

Notes:

* The ISORM level of the Ethernet is 2, of IP is 3, and of TCP and UDP is 4. The ISO level of the simulation is 7, and its presentation level is 6.

* The simulation session level, 5, does not show explicitly.

* Each of Telnet and FTP span levels 5 through 7.

Do you have any comments?

See reverse side for instructions.

Name and position _____

Organization _____

Address _____

City, State, Zip _____

Work Telephone (Include area code) _____ *Submission date* _____

Comments on the Document *(You may add pages as needed and send comments in an envelope to the address listed on the reverse side.)*

Problem areas

Section name and wording

Recommended Wording

Reason/rationale for recommendation

Other suggestions

In a continuing effort to make our standardization documents better, IST has provided this form for use in submitting comments and suggestions. This form may not be used to request copies of documents, waivers, deviations or clarification of specification requirements on currents contracts. Comments submitted on this form do not constitute or imply authorization to waive any portion of the referenced document or to amend contractual requirements.

If you have any suggestions for improving or adding to the document, fill out the form below. Be as specific as possible. You can mail this page by removing it from the document, fold along the lines indicated on the other side, and tape along the open edges. Place a stamp where indicated.

(fold along this line)

(fold along this line)

Place stamp here

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

ATTN: Amy Vanzant-Hodge