DIS - The Next Generation:
A Perspective for European Space Programmes

Hans Jense[1]
Nico Kuijpers

TNO Physics and Electronics Laboratory
P.O. Box 96864, 2509 JG The Hague, The Netherlands
Phone ++31 70 374 0000, Fax ++31 70 328 0961
E-mail: jense@eos.arc.nasa.gov, kuijpers@fel.tno.nl

Abstract

During the past decade, requirements within the US defence community for simulation-based tools to support acquisition, planning, (team) training, and analysis, have resulted in a standard for distributed simulation known as DIS (Distributed Interactive Simulation). The potential advantages of distributed simulation technologies are evident: increased flexibility, building on existing software and communications standards, maximization of the use of existing simulation assets, and thus reduced costs.

Two important trends have recently emerged in the DIS community: a drive towards the use of DIS technology outside the area of defence applications, and a recognition of the fact that, while the DIS protocols constitute the low-level basis for distributed simulation, a set of standards and tools at a higher abstraction level is required to enable the timely and effective development of applications.

This paper summarizes current developments in the DIS community: the move towards the proposed High Level Architecture (HLA), and the relevance of both the existing DIS standard and the emerging HLA for space applications. These developments are illustrated by examples from ongoing R&D work at TNO-FEL on:

- an environment that supports scenario management for large scale distributed simulations;
- the development of a generic Advanced Simulation Framework (ASF);
- a feasibility study into the suitability of DIS and HLA for highly interactive distributed virtual environments.

In particular the paper aims to illustrate the relevance of this work for applications in the space domain.

1. Introduction

Motivated by a requirement for simulation-based tools to support acquisition, planning, training and analysis efforts, the defence community - most notably the US Department Of Defence (DOD) - has heavily invested in R&D programmes on distributed simulation. This has resulted in a suite of protocols, collectively known as DIS (Distributed Interactive Simulation) as a standard for interconnecting large numbers of heterogeneous simulators across local and wide area networks. The DIS protocols allow the combination of such diverse elements as real-time interactive human-in-the-loop simulators, computer generated autonomous agents, numerical simulations of physical processes, and live instrumented participants, to operate together in one shared synthetic environment.

---

[1] Hans Jense is currently on leave at: NASA Ames Research Center, Human and Systems Technology Branch, Mail Stop 262-2, Moffett Field, CA 94035-1000, USA

In this paper we first briefly review the key concepts underlying the DIS standards and illustrate their use in a real-world application: the training of Forward Air Controllers in an Electronic Battlefield Facility (Section 2). Although this is a military application, it illustrates the combined use of real-time human-in-the-loop simulators, computer generated entities, and various simulation management tools, all interoperating through their DIS interfaces. Section 2 concludes with a discussion of the advantages and shortcomings of the current DIS standards. We proceed in Section 3 by describing the proposed successor to DIS: the DOD High Level Architecture. The definition of HLA addresses a number of concerns and requirements of the space community, and these aspects will be highlighted in particular. As the first HLA implementations are highly experimental, and the HLA definition is far from being frozen in its final form, many users are looking at ways to overcome the problems with DIS now through other means. The approach taken by TNO-FEL is through the development of an Advanced Simulation Framework (ASF), a generic software architecture that provides an abstract high-level interface between DIS applications and the underlying communications protocols. Like HLA, ASF greatly enhances the interoperability of existing simulations and promotes the reuse of existing simulation software components. Once a suitable implementation becomes available, ASF based simulations can be made HLA compliant in a relatively straight-forward manner. In Section 4 we indicate a number of requirements for distributed simulation in the space domain that might potentially be addressed by the DIS and HLA standards. Finally, Section 5 provides conclusions and some tentative recommendations regarding the use of DIS/HLA for space applications.

2. Distributed Interactive Simulation

2.1 The DIS approach

The underlying technical approach of the DIS protocols is based on the following principles (Ref. 2):

- object/event architecture: all dynamic entities in the synthetic environment inform all other entities of their status and actions through the transmission of standard information packets, so-called Protocol Data Units (PDUs);
- autonomy of simulation nodes: all events caused by an individual simulator are broadcast and available to all other simulators. It is the receiving simulator's responsibility to determine the effects on its own state caused by the event;
- transmission of state change information only: simulators transmit only changes in the state of the entity they simulate. Continuous activities are transmitted at a reduced update rate. Each simulator locally extrapolates the last reported states of other simulators until the next state update is received. This is referred to as dead-reckoning.

These principles allow simulators to minimize network traffic, make optimal use of local processing power, and enter and leave the synthetic environment without disrupting it.

2.2 The Electronic Battlefield Facility

The Electronic Battlefield Facility (EBF), developed at TNO-FEL, provides simulation services for development, research and training purposes (Ref. 3). The facility provides interoperability of simulators based on the DIS protocol and is able to integrate both high and low-echelon command and control aspects. A simulation exercise, or session, consists of a synthetic environment, a number of participants and a scenario in which the mission tasks for each participant are defined. The preparation and coordination of a session is called *scenario management*. The EBF supports scenario management by means of control, analysis and presentation tools such as a 2D overview map and 3D visualization. Manned simulators and unmanned computer driven simulations of tanks, aircraft and infantry are part of the electronic battlefield. Simulators from outside the facility can join in a session. Links with several national and international institutes and industries have already taken place. A linkage with the US Army Defence Simulation Internet (DSI) creates the possibility of a world-wide electronic battlefield exercise. Figure 1 gives a schematic overview of the Electronic Battlefield Facility.

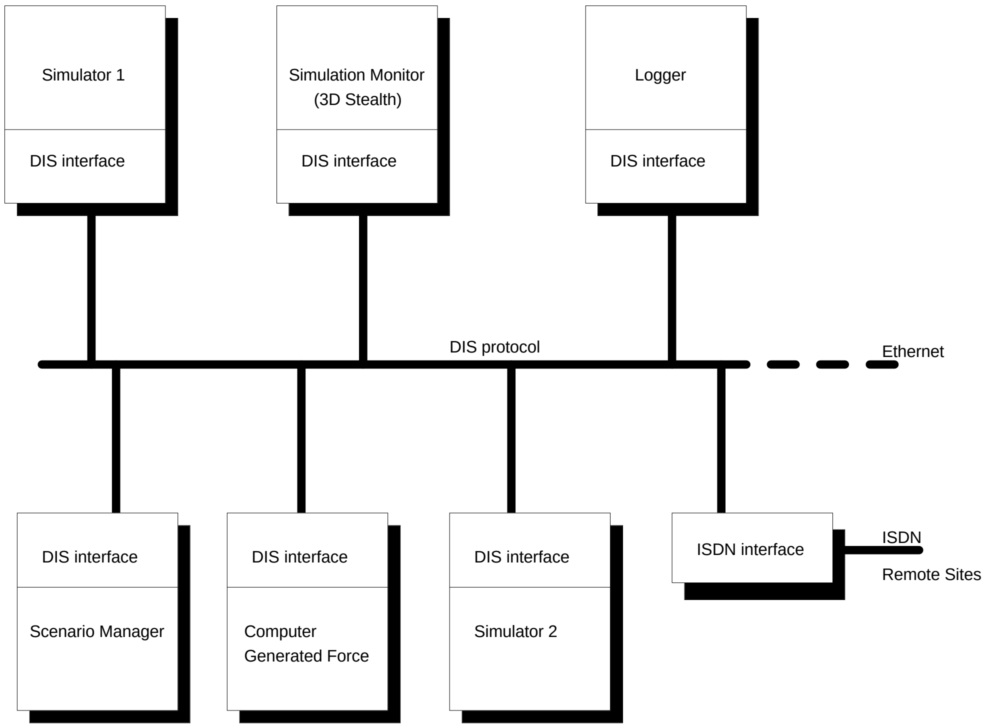

Figure 1: The key components of the Electronic Battlefield Facility

The Electronic Battlefield Facility can be used to support prototypes of training applications based on distributed simulation and virtual environments. One example is a training facility for a Forward Air Controller (FAC). A FAC controls and monitors an air-to-ground attack against a specific target, for example an enemy tank or rocket launcher, from a position on the ground in unfamiliar terrain, communicating with the attacking aircraft via radio. Since training in the field is hardly possible and expensive, simulation based training is being called for. To this end, a demonstrator is currently being built for the initial training of student FACs. A future upgrade of the simulator to make it suitable for proficiency upkeep training of operationally qualified FACs is planned. The demonstrator consists of a number of simulation modules, operating together in one simulated exercise in which a pilot, a student FAC and an instructor can operate together in a synthetic environment.

Using Virtual Environments technology such as a Head Mounted Display (HMD), the student FAC is immersed in the synthetic battlefield. Moving tanks are simulated by so-called Computer Generated Forces (CGF) modules. CGFs are often used in DIS exercises to generate (enemy) forces, without the need of a human in the simulation loop to control them. Furthermore, there is a simple F16 simulator to simulate the aircraft that delivers the close air support, and there is an instructor console. The FAC and the pilot communicate through radio. Practicing communication procedures is an essential part of the training and the trainee must give fast, clear and concise responses. The radio is also simulated by a special audio simulator that operates within a DIS environment. It enables the simulation of such features as background noise, lineof-sight and over-the-horizon effects with respect to signal strength, etc. All DIS PDUs (including voice) are recorded. This allows the instructor to give feedback to the student since the entire exercise can be evaluated by replaying the recorded scenario.

2.3 Strengths and weaknesses of the DIS approach

DIS is extensively used as standard protocol for simulator interoperability. An advantage of the DIS concept is that all DIS compliant simulators can operate within one simulated exercise. Stand-alone simulators become open systems. However, DIS has some drawbacks. Two features in the underlying data transport mechanism cause problems. Firstly, messages can get lost or arrive in the wrong order which makes the system unreliable, as there are no provisions in the DIS standard for time-management. Secondly, the messages sent are part of standardized, fixed-sized Protocol Data Units (PDUs). In the case that simulation experiments have to be repeatable, reliable data transfer is crucial. Since DIS was originally developed for training purposes, unreliability was never recognized as a problem (Ref. 4).

The fixed-sized PDUs in combination with the dead-reckoning algorithms are well-suited to communicate state-updates and events between simulators (short messages). However, PDUs are generally ill suited to transfer environmental state updates, as these often involve the transfer of large amounts of data. For instance, if weather plays a role in a distributed simulation, changes in wind direction and force are broadcast as meshes of wind vectors. This has lead to the current situation where new PDUs are being defined for all kinds of data exchange, resulting in an explosion of PDU formats.

Basically, DIS is nothing more than a protocol. No requirements for the software architecture of DIS compliant simulators are defined. Software packages can be purchased which provide the programmer with primitives for DIS based communication. However, using this type of software, specific in-depth knowledge about DIS is still required. A basic software layer which takes care of the communication and with which a simulation can communicate on a higher level would be a solution for programmers who are not familiar with DIS.

The above mentioned shortcomings of the DIS standard, i.e., the limited time-management functionality, and rigid definition of the PDUs, and the close coupling between the applications and the underlying communication mechanism, are being addressed in the next section, which describes the proposed successor to DIS: the High Level Architecture (HLA).

3. A Generic Approach to Distributed Simulation: the High Level Architecture

Over the past few years, the DIS protocol standard has provided a solid base for real-time distributed simulation. DIS has been widely accepted, both by industrial and governmental users. The Aggregate Level Simulation Protocol (ALSP), which supports event-driven simulation, has been successfully applied in the field of Wargaming (Ref. 6). In order to unify these domains and extend the scope of the existing standards the US Defence Modeling and Simulation Office (DMSO) in 1995 initiated a development to establish a new standard for modeling and simulation. This new standard, called High Level Architecture (HLA), is currently being defined. The HLA development is targeted at two main objectives. The first is to facilitate interoperability amongst heterogeneous simulations, e.g., real-time virtual simulations, aggregate level event driven simulations, non real-time constructive and live instrumented entities. The second objective is to promote reuse of simulations and their components. HLA is an attempt to capture the best from both DIS and ALSP and provide a standard architecture for simulator software at the same time.

3.1 The High Level Architecture

The HLA standard promotes reuse of simulations and their components. It attempts to specify the general structure of the interfaces between simulations without making specific demands on the implementation of each simulation. The standard is being developed in a cooperative, consensus-based forum of developers (Ref. 1).

Recently, the HLA baseline documents have become available in the USA. In this baseline, four basic concepts are defined:

1. The *Run-Time Infrastructure* (RTI) is an implementation of a distributed operating system which will be the basis software layer for future HLA applications. The software layer takes care of communication between all simulation models.
2. The *Interface Specification* is a formal, functional description of the interface between the HLA application and the Run-Time Infrastructure.
3. A set of technical *rules* is defined to which an HLA participant has to comply.
4. The *Object Model Templates* are standardized formats to define the functionality of simulation models and the interaction between models. In this way the capabilities of all simulation models are defined before the actual simulation takes place.

We will give a concise description of these fundamental concepts in the following sections, along with additional HLA terminology.

3.2 HLA Concepts and Terminology

A number of *federations*, each concentrating on a specific area of simulation, are envisaged. Each federation contains a number of members called *federates*. Federates may be simulation models, data collectors, simulators, computer generated forces or passive viewers. A simulation session, in which a number of federates participate, is called an *execution*. Simulated entities, such as tanks or aircraft, are referred to as *objects*. Live entities can be mapped onto objects (by using simulation surrogates), so that they can appear identical to simulated entities. All possible interactions between the federates of a federation are defined in a so-called *Federation Object Model* (FOM). The capabilities of a federate are defined in a so-called *Simulation Object Model* (SOM). The SOM is introduced to encourage reuse of simulation models.

The state of each object is defined by its *attributes*. Attribute values can be passed from one object to another. Objects interact with each other and with the environment via *interactions* which may be viewed as unique events, such as the firing of a weapon, or a collision between vehicles. Initially, an attribute is controlled by the federate that instantiated the object whose state the attribute is part of. However, attribute ownership may change during the execution. This mechanism allows, for instance, the fidelity of a vehicle simulation to be adjusted by passing its position attribute from one kinematic simulation to another.

In order to reduce network traffic and limit the amount of computation each federate has to perform, the HLA provides a mechanism of *publication* and *subscription.* Upon initialization of a federation execution, each simulation registers (with the RTI) which objects and which attributes it will represent (publication). It also registers which attributes and interactions it needs in order to be able to perform its task (subscription). Note that a federate can not only subscribe to attribute types, but also to (ranges of) attribute values. The aim here is to filter as much information as possible at the source. Both publications and subscriptions are dynamic and may be changed during a session.

The Run-Time Infrastructure supports HLA compliant simulations through a number of services. The main categories of services are:

- Federation Management: create/destroy execution (of a federation), join/resign execution (of a federate), pause/resume execution.
- Declaration Management: subscription/publication
- Object Management: instantiate/delete (an object), update/reflect (an attribute), request/provide (an attribute), send/receive (an interaction)
- Ownership Management: transfer ownership (of an attribute) from one object to another.
- Time Management: handling of messages in different ways, depending on the requirements of their destinations, e.g., in receive order (appropriate for real-time human-in-the-loop simulators), in priority order, in causal order, or in timestamp order. These last three methods are typically suitable for non real-time event driven simulations for, e.g., analysis purposes.

The current, (prototype) implementation of the RTI is based on the Common Object Request Broker Architecture (CORBA), specifically the ORBIX implementation of CORBA. Although most users so far have reported that the functionality of this implementation is sufficient for their immediate needs, most have indicated to be dissatisfied with its performance. It is therefore likely that future versions of the RTI will be based on other CORBA implementations, or may not be based on CORBA at all. In particular CORBA's Interface Definition Language (IDL) will be replaced by a C++ Applications Programmers Interface.

The following list summarizes the major differences between the new HLA standard and its DIS predecessor:

- HLA is an architecture (allowing the use of different components, such as communication protocols); DIS specifies one single protocol;
- HLA is much more scaleable than DIS, because network traffic is greatly reduced due to its declaration management facilities (publication/subscription);
- HLA supports a much wider range of simulations, e.g., real-time virtual (human-in-the-loop) simulations, constructive (non real-time) simulations, aggregate level (event driven non real-time) simulations, and instrumented live entities, where DIS was chiefly targeted at the first category only;
- HLA decouples the simulation application from the underlying communication protocol. Thus, HLA simulations can be developed by experts in the application domain, rather than by communication (DIS) gurus;
- HLA is based on the much more efficient point-to-point communication mechanism; HLA aims at using multicast communication, while DIS relies on broadcast communication;
- HLA still has to prove itself, whereas DIS already has a track record of numerous successful implementations and applications;
- It remains to be seen when the HLA implementations will become available for non-US users. DIS implementations, both public-domain and commercial, are freely available.

These comparisons indicate that HLA will be a major step forward with respect to DIS, but also that the transition from DIS to HLA is going to be a gradual one. The problem then becomes: how to make this transition as painless as possible in terms of technological risks and development effort. One approach towards this goal is described in the next section.

3.3 Advanced Simulation Framework

Currently the HLA standard is far from being frozen in its final form. Moreover, the first implementation of the RTI, although its functionality is at an acceptable level, still leaves a lot to be desired in terms of performance. One of the aims of the HLA is to provide a much cleaner separation between simulation application software and the underlying communication protocols. In anticipation of this, TNO-FEL decided it would be sensible to prepare for the introduction of new and rapidly changing simulation standards by providing an abstraction layer (or “middleware”) that for the moment offers a DIS-compliant interface, whilst enabling the migration to a HLA-compliant interface sometime in the next few years.

We can illustrate this by looking at the EBF application described in the previous section. From an architectural point of view, the various EBF software modules have much functionality in common regarding the communication of state information. All modules need an up-to-date representation of the state of the simulated environment at each point in time during a session. This state information is kept up-to-date by communicating DIS PDUs. A software framework was developed at TNO-FEL which encapsulates this common communication functionality. The so-called Advanced Simulation Framework (ASF) was designed using an object-oriented method in order to achieve reusability and extensibility. As was indicated before, the development of the ASF was inspired by the emerging HLA standard. Although DIS is being used as communication protocol, the ASF is designed such that it can easily be adapted to become HLA compliant. In this way all EBF modules built upon ASF can be made HLA-compliant as soon as this becomes feasible.

The Advanced Simulation Framework (ASF) is a generic software architecture to facilitate interoperability between simulations and promote reuse of simulators and their components. All future EBF applications will be built on top of this software framework. The ASF supports the reuse of models of entity components such as sensors, weapons, communication devices, behavior models, etc. It has been set up in such a way that simulation models can easily be replaced, e.g., several types of combat vehicles can be created in a short time, that differ, for example, in the type of weapons or sensors. A component library is being set up that contains many simulation models that can be reused for new applications.

In the ASF two subsystems can be distinguished. The *Environment* realizes a protocol-independent overview of the simulated battlefield. It reflects the up-to-date status of the simulation objects (or entities) and allows the application to add new objects. Furthermore, the application can subscribe to relevant information, thus enabling the system to discard non-relevant information at an early stage. The application can receive and issue unique events such as simulation management events. The Environment shields the application from the partially protocol-dependent ObjectServer.

The *ObjectServer* represents the underlying run-time infrastructure. It exchanges object information with other platforms and provides the Environment with up-to-date status information of the battlefield. The ObjectServer is protocol-dependent, i.e., if another protocol has to be supported, this subsystem needs to be (partially) replaced or extended. The ObjectServer currently relies on the DIS protocols, but a switch to HLA is foreseen in the near future. Figure 2 depicts the ASF layer and application modules built upon it.

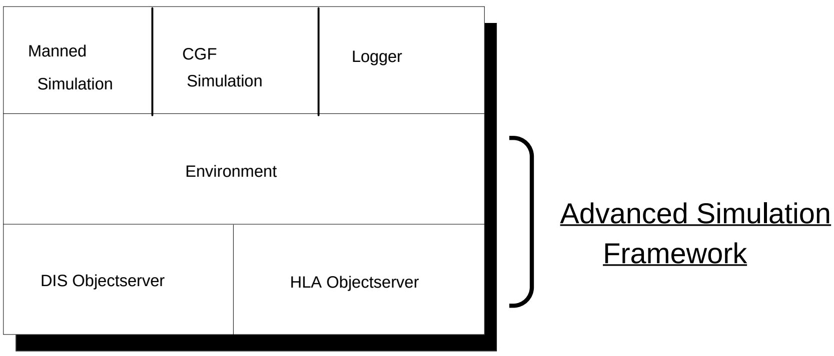

Figure 2: The Advanced Simulation Framework

The first application of the Advanced Simulation Framework will be the FAC demonstrator mentioned in Section 2. Some of the modules were already available in the Electronic Battlefield Facility, while new modules will be based upon ASF. In particular, these are the Virtual Environment for the FAC and the F16 simulator. Other ASF based modules for the EBF that are planned for the future include a number of generic visualization and analysis tools.

4. Benefits of DIS/HLA for European Space Programmes

As in the defence area, modeling and simulation are indispensable tools in the space domain. A recent paper provides an overview of the use of simulation for verification and testing tasks in space projects (Ref. 5). The paper mentions a number of tasks related to space systems engineering, such as:

- concept study;
- requirements analysis;
- design;
- system integration;
- system testing.

In all of these tasks, the use of simulation is increasing, motivated by several reasons:

- in the space domain there is a continuing trend toward more international collaboration, requiring the interoperability of independently developed simulation facilities as well as the distribution of project test-beds that currently are mostly centralized;
- several phases of space projects rely on extensive verification and validation exercises by combining hardware-in-the-loop and software based simulation;
- in the space domain there also is an increasing drive toward further cost reduction by promoting the reuse of existing simulation assets;

All of these issues have been recognized in the military domain as being crucial. They form the primary motivation behind the development of both the DIS and the HLA standards. For instance: the desire to be able to perform team-training instead of individual crew-training led to the requirement for interoperability between geographically dispersed training simulators. This led to the development of the DIS standard which primarily focuses on real-time virtual (human-in-the-loop) simulations. Next, the drive towards further cost reduction by optimizing the use of existing simulation assets through reconfigurability instead of new development efforts, along with the widening of the scope of simulation types to include live entities (instrumented hardware) and non real-time simulations led to the development of the HLA.

Another relevant issue here is the use of project test-beds. In the defence community, the use of test-beds in various phases of, for instance, system acquisition or development projects, has become common. Typically, requirements for new systems are identified, based on the analysis of DIS based operational exersizes. Next, new system concepts are developed and prototyped, using the same testbed, and possibly also the same excersize scenarios. The designs are further refined and verified. During later stages of the project, system testing is conducted, either by using software simulations of the new system, or an instrumented hardware implementation. During all these phases, the underlying DIS test-bed remains the same, while components (simulation models, simulators, analysis and visualization tools, scenarios) are substituted as necessary.

One application area deserves special mention: (team) training. In the military domain this is the application area that has benefited the most from the introduction of distributed simulation. In the space domain, the training task usually involves only a few individuals (controllers, mission specialists, astronauts) and the fidelity requirements for the simulated environment are often higher than in defence applications. This is where a reconfigurable project test-bed, consisting of highly accurate constructive simulations, hardware-in-the-loop components, event-driven simulations as well as real-time virtual simulations would allow the training of operational procedures. Initial training is possible at an early stage in the project, when the space system is still in its conceptual phase. Results from these initial training exercises could feed into the requirements analysis phase. Later, as more and more components are designed and implemented, the training environment would consist of validated simulations of the final system components, including hardware. Basing the project test-bed on a distributed architecture, either based on DIS, HLA, or something in-between (such as the ASF), would in principle allow the test-bed components to be geographically dispersed, and to be easily adaptable as the project progresses.

Another area where distributed simulation technology can potentially be used to great advantage is collaborative virtual environments. Collaborative VEs have been proposed for several applications, including training and design. Currently the most advanced 3D user interfaces to such collaborative applications are immersive virtual environments. Immersive VEs will probably pose severe communication requirements due to the nature of the generated network traffic. Additionally, there are a number of critical human factors that need to be assessed before this technology can be successfully applied. TNO-FEL is currently a partner in the COVEN project, carried out in the framework of the EU Advanced Communication Technology and Services (ACTS) programme. COVEN aims to develop a software platform for teleworking and virtual presence applications. Networking trials will be performed between the COVEN partners, making use of the ISDN and ATM communication infrastructure. In the framework of COVEN TNO-FEL will lead a study into the feasibility of using DIS and HLA concepts for the development of the COVEN platform.

Because of the similarities between the simulations used in the military and the space domains as well as the potential benefits to European space programmes, both the existing DIS standard and the HLA deserve a closer look. A considerable R&D effort is required to assess the exact requirements of space applications with respect to a distributed simulation approach, but this effort could at the same time yield valuable results to help steer the evolution of the existing DIS technology towards the more universal HLA framework for distributed simulation. The potential advantages of distributed simulation technologies, not limited to but including DIS and HLA, for a number of space applications are evident:

- increased flexibility;
- building on existing software and communications standards;
- maximization of the use of existing simulation assets;
- reduced costs.

5. Conclusions

We have presented an overview of the current state of the DIS standard and the development of its successor, the HLA. The use of DIS has been illustrated by means of an example from the defence area: a test-bed for development, research, and training known as the Electronic Battlefield Facility (EBF). We have given a concise description of the major concepts behind both, and compared their strengths and weaknesses. Based on the comparison of the two approaches, the need for a gradual transition from the established DIS standard to the still to be implemented HLA standard has been identified. This need is addressed by the development of the Advanced Simulation Framework (ASF), a generic software layer that provides a clean separation between the applications and the underlying communication protocols. Finally, the situation with respect to simulation in the space domain has been compared to that in the military field. In spite of there being a number of differences, several basic similarities have also been noted, including trends towards more international collaboration, the need for interoperability of heterogeneous simulators, and the drive toward cost reduction through reuse of existing simulation assets. All of these suggest an approach based on distributed simulation. Moreover, we have mentioned a number of applications in the space domain that exhibit technical characteristics and requirements that can potentially be addressed by standards for distributed simulation in some form. These similarities justify in our view a careful study of both the requirements and the capabilities of existing standards (DIS), current developments (ASF, COVEN), and emerging architectures (HLA) for distributed simulation.

Acknowledgments

The authors would like to thank Robert-Jan Elias for sharing his knowledge of the High Level Architecture, Marco Brassé for describing the Forward Air Controller application, and Wim Huiskamp for critically reading early versions of the paper. Finally, we would like to thank the Programme Committee for accepting this paper in spite of it being submitted late due to the inherent unreliable message delivery capabilities of the World Wide Web.

References

[1] Defence Modeling and Simulation Office, *HLA Management Plan High Level Architecture for Modeling and Simulation*, Version 1.6, 17 July 1995.

[2] DIS Steering Committee, *The DIS Vision, A Map to the Future of Distributed Simulation*, Version 1, Institute for Simulation & Training, May 1994, Orlando, Florida, USA, IST-SP-94-01.

[3] N.H.L. Kuijpers, R.J.D. Elias, R.G.W. Gouweleeuw, *Electronic Battlefield Facility* in Battlefield Systems International 96 'Integrated Battlefield Management', Volume 2, 4-6 June 1996, Chertsey, UK.

[4] Duncan C. Miller, *The DOD High Level Architecture And The Next Generation Of DIS*, 14th DIS Workshop on The Standards for the Interoperability of Distributed Simulation, Orlando, FL, March 1996.

[5] B. Melton and J.E. Miró, *The role of simulation in space system verification*, European Space Research and Technology Centre ESTEC, Noordwijk, The Netherlands.

[6] R. Weatherly, D. Seidel, and J. Weismann, *Aggregate Level Simulation Protocol*, Proc. 1991 Summer Comp. Sim. Conf., pp. 953-958, Baltimore, July 1991.