

An Orion/Ares I Launch and Ascent Simulation - One Segment of the Distributed Space Exploration Simulation (DSES)

Victoria I. Chung^{*}

NASA Langley Research Center, Hampton, VA, 23681

Edwin Z. Crues[†]

NASA Johnson Space Center, Houston, TX, 77058

Mike G. Blum[‡]

Science Applications International Corporation, Moffett Field, CA, 94035

Cathy Alofs[§]

Teledyne Brown Engineering, Huntsville, AL, 35805

Juan Busto[¶]

John F. Kenney Space Center, Kennedy Space Center, FL, 32899

This paper describes the architecture and implementation of a distributed launch and ascent simulation of NASA's Orion spacecraft and Ares I launch vehicle. This simulation is one segment of the Distributed Space Exploration Simulation (DSES) Project. The DSES project is a research and development collaboration between NASA centers which investigates technologies and processes for distributed simulation of complex space systems in support of NASA's Exploration Initiative. DSES is developing an integrated end-to-end simulation capability to support NASA development and deployment of new exploration spacecraft and missions. This paper describes the first in a collection of simulation capabilities that DSES will support.

I. Introduction

NASA has embarked on its' new Vision for Space Exploration (VSE) initiative. In order to deliver products and achieve VSE objectives, NASA has established a development structure that empowers multiple NASA centers to lead the architecture and implementation of the various space exploration components. The NASA centers have roles and responsibilities for the delivery of the following products: Thermal Protection System by the Ames Research Center (ARC); Crew Module (CM) of Orion by the NASA Johnson Space Center (JSC); Ares I (a crew launch vehicle) and Ares V (a cargo launch vehicle) by the Marshall Space Flight Center (MSFC); ground launch operation by the Kenney Flight Center (KSC); Service Module (SM) of the Orion by the Glenn Research Center (GRC); and Launch Abort System (LAS) of Orion by the Langley Research Center (LaRC). By policy, the development and testing infrastructure for these essential space exploration components is situated at different geographical locations across the continental United States. Simulation is an essential part of the development lifecycle of this new space exploration endeavor and requires integration of all the individual simulations components that reside at the different NASA centers.

The traditional approach to the simulation of space vehicles has been through disjoint collections of simulations. Each individual simulation usually focuses on a specific domain aspect of a space vehicle and is developed, maintained,

^{*} Assistant Branch Head, Simulation Development and Analysis Branch, Mail Stop 125B, AIAA Senior Member.

[†] DSES Project Lead, AST Robotics Specialist, Simulation and Graphics Branch, Mail Stop ER7, AIAA Senior Member.

[‡] Principal Engineer, Space Simulation Development, Science Applications International Corporation, Mail Stop 243-6, AIAA Member.

[§] Senior Systems Analyst, Teledyne Brown Engineering.

[¶] Project Engineer, IT-C2.

and executed within the confines of the facility having the particular domain expertise. While this provides high fidelity modeling in a particular area, other domain areas are usually modeled to a lower fidelity. When a simulation needing high-fidelity models across the mission is required, a larger and more complex simulation is built; these simulations often re-implement the existing domain specific models from other locations in the local simulation architecture.

Fortunately, a new approach to building large scale simulation has emerged. This is the area of distributed simulation. In this case, the simulation is a collaborative association of multiple, geographically dispersed simulations that work in coordination to simulate the system. This approach has a number of benefits. For instance, organizations with specialized domain expertise can contribute to the aggregate simulation, combining their individual expertise into a larger body. Another benefit is the ability to restrict access to or hide the details of proprietary technologies or algorithms while making the service available to the aggregate simulation. Yet another benefit is the ability to distribute the cost of development across multiple organizations and utilize the shared computer resources and development tools of those organizations. As a result, substantial return of investment will be gained due to reduced duplication of effort, reduced time to prototype, discovering problems earlier in development, reduced long term development risks, and early definition and testing of system interfaces.

The Distributed Space Exploration Simulation (DSES) is a NASA research and development project focusing on the technologies and processes for the collaborative, interoperable, and distributed simulation of complex space systems involved in the exploration of our solar system. One of the leading distributed simulation technologies in use today is the High Level Architecture (HLA) which was originally developed and deployed by the U.S. Department of Defense (DoD) Defense Modeling and Simulation Office (DMSO). This is an evolution of previous DoD distributed simulation technologies and has now been adopted in a modified form as an international standard by the Institute of Electrical and Electronics Engineers (IEEE) as the IEEE 1516 HLA standard.¹⁻⁴

This paper presents an overview of a distributed Orion/Ares I Launch and Ascent simulation. This simulation is the first instantiation of a DSES federation.[#] It is composed of the following 5 active federates:

- GO – Ground Operations elements for pre-launch and launch activities.
- Ares I – 1st and 2nd stage elements of the Ares I launch vehicle.
- Orion CM and SM – Crew and Service module elements of the Orion spacecraft.
- Orion LAS – Launch Abort System element of the Orion Spacecraft.
- Crew – Crew interface elements for the Orion Spacecraft.

As the DSES project advances along with the Constellation Program (CxP), additional mission elements (federates) will be added. As an example, an International Space Station (ISS) federate will be added to support the ISS servicing mission and lunar element federates will be added to support lunar exploration missions.^{**}

II. DSES Overview

The principal objective of the DSES project is to investigate the use of distributed simulation in general (initially the use of HLA) and build large scale integrated space vehicle simulations in support of the Constellation Program (CxP). The Constellation Program is the organization responsible for implementing the vision of the Exploration Systems Mission Directorate (ESMD), which leads NASA's new Vision for Space Exploration initiative. The Vision for Space Exploration involves the technical expertise and active participation of scientists and engineers from all 10 of NASA's research and flight facilities.

Like most projects, the DSES project did not emerge fully formed in either scope or participation. A number of NASA centers have been investigating distributed simulation technologies. One precursor to the DSES project was the DIstributed Simulation (DIS) project. This project's purpose is to build a distributed simulation for use in flight procedures development and training of the HII-A Transfer Vehicle (HTV) and its operations in proximity of the International Space Station (ISS). This simulation is referred to as the HTV Flight Controller Trainer (FCT) simulation. This is a collaborative effort between the Japanese Aerospace eXploration Agency (JAXA) and the NASA Johnson Space Center (JSC) Mission Operations Directorate.⁵⁻⁶ In this simulation, the HTV components run at the JAXA facilities in Tsukuba, Japan and the ISS components run at JSC in Houston, Texas. A graphic scene generated from this

[#] A federation is the HLA term for a collection of individual simulations (federates) that run collaboratively and exchange data and coordinate execution using the HLA interoperability infrastructure.

^{**} The Constellation Program is the organization in NASA that is responsible for guiding and coordinating the activities required to implement NASA Vision for Space Exploration. One of the Constellation Program's responsibilities is to oversee the Orion and Ares I Projects and is the sponsoring organization for the DSES project.

simulation is shown below in Figure 1. Much of the knowledge and technologies used in the HTV FCT have been directly applied to the DSES project.



Figure 1. The HTV FCT Simulation

Initially, the DSES project was referred to as a “coalition of the willing” in the sense that small groups of NASA scientists and engineers with similar interests and very little funding met on a regular basis to exchange information and ideas on new approaches to simulation. These groups located at NASA JSC, NASA Ames Research Center (ARC) and NASA Langley Research Center (LaRC) configured a small informal network of computers and began testing out some simple distributed simulations. As the DSES work progressed, interest began to spread to other centers. Another interested group at NASA Marshall Space Flight Center (MSFC) soon joined the growing DSES team. Recently, a group at NASA Kennedy Space Center (KSC) has joined in on the DSES work. NASA Glenn Research Center (GRC) will soon join the DSES team and further extend the scope and breadth of its capabilities. This will bring the count to six NASA centers with plans to extend to all NASA centers in the coming years.

In Fiscal Year 2007, DSES is primarily a product development project that focuses in three principal areas: Network Infrastructure, Software Infrastructure, and Simulation Development. The following paragraphs provide a high level description of the three principal work areas for DSES. The details of these areas are covered in a companion paper.⁷

A. Network Infrastructure

The DSES network infrastructure is currently based on an interconnection of facility local area networks through the NASA Integrated Services Network (NISN). The NISN backbone along with a combination of externally accessible IP addresses and access allocations through the facility network firewalls forms the NASA Distributed Simulation Network (DSNet). While the current network has limited levels of service guarantees, the determination of the form and necessity of network guarantees is an area of DSES investigation. Figure 2 shows the 5 currently active DSES NASA centers connected through the DSNet: ARC, JSC, KSC, LaRC and MSFC. There are plans to connect the remaining 5 NASA center as project participation expands.

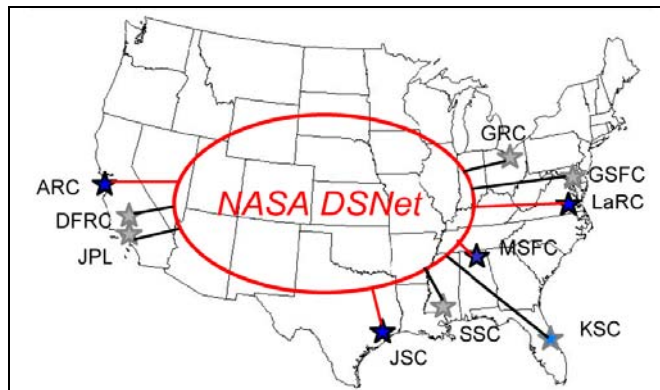


Figure 2. Distributed Simulation Network

While the DSNet does provide the necessary connectivity for distributed simulation, pure connectivity is not sufficient; it is also important to understand both the status and performance of the network. To gain insight into both the current status and performance of the network connections, the DSES project is creating a collection of DSNet status and performance tools.

B. Software Infrastructure

The DSES software infrastructure consists of identifiable collections of software assets located at each of the participating NASA facilities. These software assets include a common distributed simulation software framework. For the initial phases of the DSES project, this framework will be the High Level Architecture. However, other distributed simulation software infrastructures are being investigated. Each NASA facility provides simulation components for the distributed simulation using in-house simulation software and systems but rely on the common distribution framework for external interfaces, timing and execution control.

DSES project simulations are currently HLA based; they depend on a common description of interoperable data types that are defined in the DSES Federation Object Model (FOM). The FOM currently consists of a simple set of objects that are used to exchange space vehicle state information like position, velocity and acceleration. The DSES FOM also defines a collection of Interactions or message definitions that are used to exchange vehicle command and status information.

Since many of the simulations being developed for the Constellation Program are built in a common simulation development environment called Trick, the DSES project is also creating a generalized set of HLA classes. These Trick HLA classes take advantage of some of Trick's generalized IO and variable access capabilities to provide a model that can be included into almost any Trick based simulation to allow it to join into almost any DSES Federation. The mapping of FOM objects, attributes, interactions and parameters are defined in data at run time. In this way, many of the details and intricacies of HLA development will be hidden from the developers.

C. Simulation Development

The combination of DSES network infrastructure and software solutions are the essential elements for the successful development and deployment of interoperable CxP mission segment simulations. The DSES project is using an evolutionary development process. In this process the various models and systems required for a working distributed simulation of a space exploration vehicle are developed in phases. The early phases have concentrated on technology demonstrations and simulation architecture development. The next phases will concentrate on modeling capabilities, data exchange, model exchange and coordinated execution. The final phases will concentrate on integrated system execution and analysis products. The Orion and Ares I Launch and Ascent simulation is the first simulation product from this DSES simulation development activity.

III. Orion/Ares I Launch and Ascent Simulation Overview

There are a number of mission segments associated with a complete Orion mission. The Orion and Ares I launch and ascent segment was the first selected for development as a DSES federation. This mission segment was chosen because it brings all the distributed interoperable simulation pieces together into a working system. In addition, this segment demonstrates a logical distribution of space exploration components: KSC for Mobil Launcher (ML) and Launch Control System (LCS); MSFC for Ares I Stage 1 and Stage 2; JSC for Orion CM and SM; LaRC for Orion LAS; and ARC for crew monitoring and commanding. It also provides logical potential for growth in the future that includes having GRC simulate the Orion SM which will be spun off from JSC's Orion simulation and extending the distributed but interoperable simulation for Lunar Systems and Mars Systems.

In September of 2006, the DSES project successfully demonstrated a distributed version of the Orion/Ares I launch vehicle (see Figure 3 below). After this initial demonstration, the DSES team continues to develop and improve the network, software, and simulation work areas. As the DSES project advances, interest in the potential benefits to the Constellation Program continues to grow. This has, and is, resulting in subsequent DSES capabilities demonstrations.

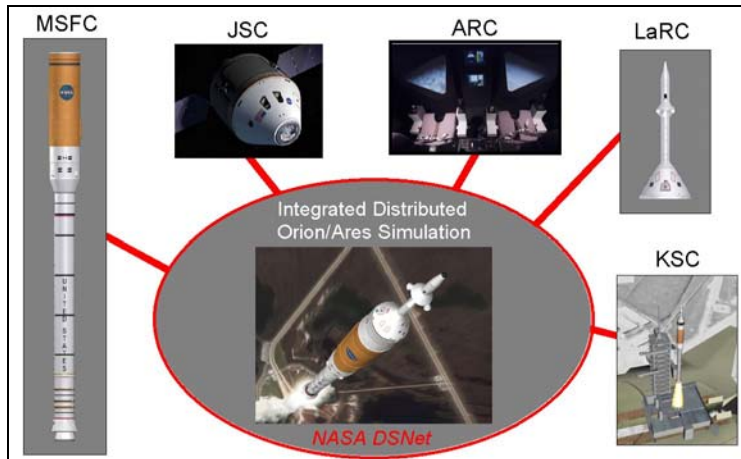


Figure 3. Orion/Ares I Launch Simulation

The current Orion and Ares I launch and ascent simulation is composed of five federates, and each federate provides functionality for important subsystems in the vehicle stack. The ground launch operation federate is a KSC simulation. The Ares I launch vehicle federate is a MSFC simulation. The Orion Command Module (CM) and Service Module (SM) federate is a JSC simulation. The Orion Launch Abort Systems (LAS) federate is a LaRC simulation. The final federate is the crew interaction simulation which runs at ARC. All these federates are run in a coordinated simulation, called a federation execution, to provide a distributed yet integrated simulation of the Orion/Ares I vehicle from launch, thru LAS jettison, to stage 2 separation and on to orbit.

The integrated and distributed Orion/Ares I launch and ascent simulation includes the nominal and abort launch scenarios. The nominal launch scenario takes about 11 minutes (see Figure 4 below). The scenario begins with a full stack of vehicles that consists of the Ares I Stage 1, Ares I Stage 2, Orion SM, Orion CM, and Orion LAS. KSC is the first simulation to start the launch process from the ground. Then, the simulation ownership is transferred to MSFC for its Ares I Stage 1 ignition and lift off phase. After 2.1 minutes, the Ares I cuts off its Stage 1 engine, and it engages the Stage 1 separation of the solid rocket booster from the Orion/Ares I stack. One second later, the Ares I Stage 2 engine ignites. The LAS simulation receives the simulation ownership at 2.7 minutes after the Ares I/Orion lift off phase; starts the LAS jettison; and separates itself from the Ares I/Orion stack. About seven minutes later, the Ares I Stage 2 engine cuts off and separates itself from the Orion. Then, the nominal launch and ascent scenario is completed.

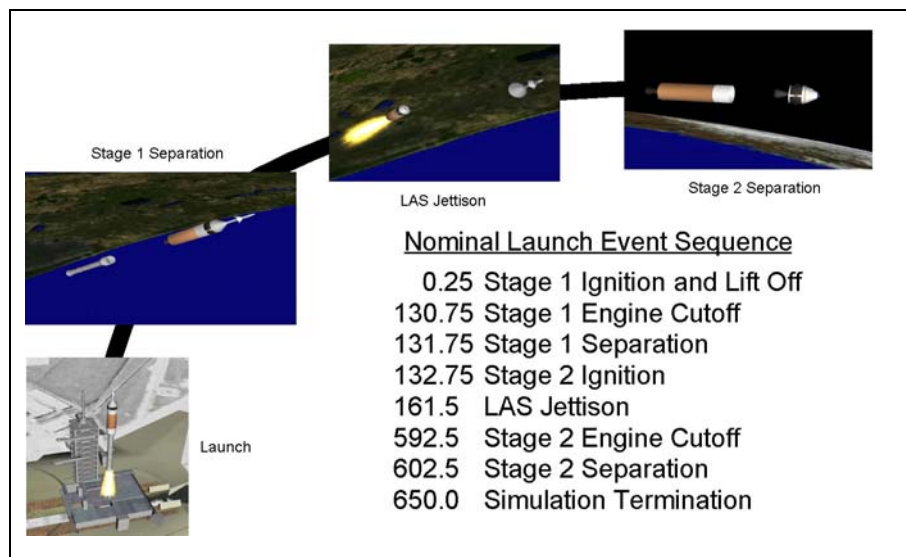


Figure 4. DSES Orion/Ares I Launch and Ascent Nominal Launch Scenario

The abort launch scenario takes about 2.5 minutes (see Figure 5 below). This scenario also starts with the full stack of Orion/Ares I vehicles launching from the ground at KSC. After 1.6 minutes, the Ares I Stage 1 engine cuts off. The

Crew simulation from ARC detects this situation and quickly sends a command to start the LAS abort process. Stage 1 Separation initiates immediately. A second later, Ares I Stage 2 separates from the Orion vehicle. The LAS ignites and pulls the Orion Crew Module off the remaining Service Module. The LAS abort motors burnout after three seconds, and the LAS jettison occurs five seconds later. After the LAS is separated from the Orion CM, the Orion CM returns safely to the Earth atmosphere. This simulation ends at 2.5 minutes after the Orion/Ares I lift off.

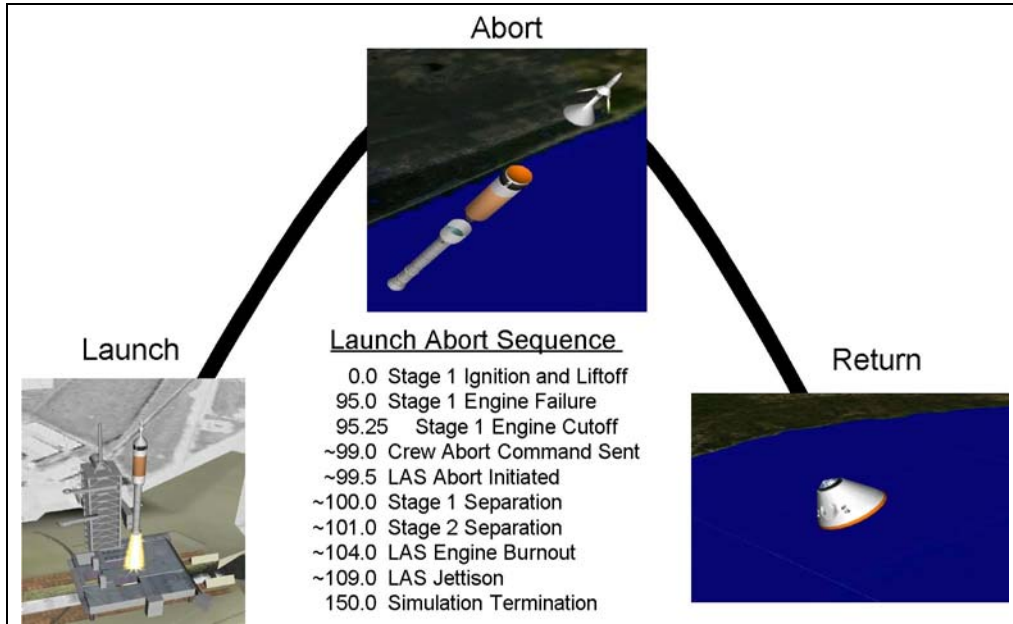


Figure 5. DSES Orion/Ares I Launch and Ascent Launch Abort Scenario

IV. Definitions, Processes and Fidelity

While it has been stated that HLA is the underlying infrastructure for this and other DSES federations, there are some additional important definitions, processes and comparisons that need to be established between the DSES participants: the development of the HLA Federation Object Model (FOM), the definition of an initialization/startup process, and a comparison of the fidelity of the dynamic responses of the individual vehicle dynamics models.

A. Federation Object Module (FOM)

HLA provides the procedural infrastructure for DSES simulation interoperability. However, HLA also requires the common definition, prior to federation execution, of all types of data that will be exchanged through the HLA interfaces. The construct that holds these type definitions is called the Federation Object Model or FOM. All participating federates must agree on and use the same FOM for a successful federation execution. As a result, all DSES participants at each of the NASA centers have agreed on a common FOM. See Figure 6 below for an example of the SpaceVehicle object, one of the principal objects exchanges between DSES federates. Object instances of this persistent type contain the space vehicle name, ownership, and states such as position, velocity, acceleration, force, attitude, and mass. In addition to persistent objects, like those of type SpaceVehicle, the FOM also defines transient or message like objects called Interactions. These are used for vehicle control messages such as command, source, and destination of federate; and timing related data.

SpaceVehicle		
vehicle_name	HLAunicodeString	ps ts
owner	HLAunicodeString	ps ts
status	HLAunicodeString	ps ts
position	SV3Vector	ps ts
velocity	SV3Vector	ps ts
acceleration	SV3Vector	ps ts
force	SV3Vector	ps ts
attitude	SVQuaternion	ps ts
rotational_velocity	SV3Vector	ps ts
rotational_acceleration	SV3Vector	ps ts
torque	SV3Vector	ps ts
mass	HLAfloat64LE	ps ts
mass_rate	HLAfloat64LE	ps ts
center_of_mass	SV3Vector	ps ts
inertia	SV3x3Matrix	ps ts
year	HLAinteger32LE	ps ts
time	HLAfloat64LE	ps ts
sim_elapsed_time	HLAfloat64LE	ps ts
system_time	HLAinteger64LE	ps ts

Figure 6. DSES FOM Space Vehicle Object

B. DSES Initialization and Startup Process

While HLA provides the interoperability infrastructure for DSES and enables the coordinated executions of distributed simulation components, HLA does not provide all the sufficient coordinating mechanisms for a successful DSES federation execution. One of the challenges faced by the DSES team was how best to coordinate the initialization and startup of the collection of federates in any given DSES Federation. We determined the need for a generic initialization sequence that can be used to create a federate which can:

- 1) Properly initialize all HLA objects, object instances, interactions, and time management
- 2) Check for the presence of all federates
- 3) Coordinate startup with other federates
- 4) Robustly initialize and share initial object instance data with other federates.

To address these requirements, the following initialization process is used in all DSES federates:

- 1) Create the federation
- 2) Publish and subscribe
- 3) Create object instances
- 4) Confirm all federates are joined
- 5) Achieve “initialize” Synchronization Point
- 6) Update object instance(s) with initial data
- 7) Wait for object instance reflections
- 8) Set up time management
- 9) Achieve “startup” Synchronization Point
- 10) Start execution

The details of the DSES initialization and startup process are beyond the scope of this paper and are covered in a companion paper.⁸

C. Dynamics Comparison

One significant requirement for a distributed or collaborative simulation is that the fidelity of the component simulations be compatible. This requires coordination between the DSES participants to assess and compare selected

aspects of the fidelities of their simulation models and environment(s). More specifically, this requires a comparison between DSES participant space environments and space vehicle dynamics.

Initially, it may seem sufficient to have each DSES participant build a 6 DOF space-based simulation, agree on initial conditions, execute the simulation and compare state and environment variable histories. However, the likelihood of getting exact or even numerically equivalent matches between these simulations is essentially zero given the diversity of model implementations and simulation environments. Therefore, we employ a multi-step testing process in order to facilitate a more systematic approach. This allows for the progressive increase in modeling complexity and for the systematic identification and categorization of the sources and sizes of comparative modeling differences.

This comparison process is divided into the following 10 principal “unit” comparisons and final integrated test case:

- Test Case 1: Earth Modeling Parameters
- Test Case 2: Keplerian Propagation
- Test Case 3: Gravity Modeling
- Test Case 4: Planetary Ephemeris
- Test Case 5: Atmospheric Modeling
- Test Case 6: External Force Affects
- Test Case 7: Combined Translational Test
- Test Case 8: Torque Free Rotation
- Test Case 9: Torque Driven Rotation
- Test Case 10: Gravity Gradient Torque
- Full Test: Integrated 6 DOF Test

The details of the dynamics comparison process are beyond the scope of this paper and will be covered in a companion paper.

V. Constituent Federates

As stated earlier in the paper, the DSES Orion and Ares I launch and ascent federation is composed of five principal federates: Ground and Launch Operations; Ares I; Orion Crew Module and Service Model; Orion Launch Abort System; and the Crew Interface. Each federate resides at different NASA centers. This section describes these federates.

A. Ground and Launch Operation (KSC)

The Ground Operations mission segment involves the ground processing of the Orion/Ares I launch vehicle in preparation for and up to launch. For the vast majority of the operational life of the Space Shuttle, KSC has used the Shuttle Ground Operations Simulation (SGOS) system for ground operations simulations. This very successful simulation combines high fidelity models with a rich control set that allowed for comprehensive software checkout and test team training. SGOS is capable of representing both the Ground Support Equipment (GSE), and the Shuttle hardware and software systems. The Ground Operations simulation being built to support Orion/Ares I will reuse, as appropriate, GSE models from the SGOS system. It is currently envisioned that KSC will use the JSC developed Trick simulation environment in conjunction with MathWorks MATLAB/Simulink model development environment for the next generation of GSE simulation at KSC. The simulations of the Orion/Ares I flight hardware and software will come directly from the NASA centers responsible for those components. KSC’s Ground Operations Simulation efforts within the DSES simulation will serve as an initial prototype of this distributed/collaborative approach and as proof of concept for further development. The first phase of KSC DSES development will consist of low fidelity simulations of the Mobile Launcher (ML) and the Launch Control System (LCS). Follow on development will increase the fidelity of those simulations corresponding to the design/development of the represented systems.

B. Ares I (MSFC)

The Ares I element is represented in DSES by two different MSFC developed simulations: the Marshall Aerospace Vehicle Representation in C II (MAVERIC-II) and the Ares Real-Time Environment for Modeling, Integration, and Simulation (ARTEMIS). The initial DSES Ares I element used the MAVERIC simulation; as DSES has evolved, the ARTEMIS simulation has been integrated.

MAVERIC is designed to rapidly create flight simulations for spacecraft and launch vehicles. It can be used for all steps of vehicle development — from concept design to actual flight of a finished vehicle. MAVERIC-II simulations can provide detailed predictions of how vehicle designs will actually perform before the vehicle is built and flown.

MAVERIC contains generic models for the propulsion system, the reaction control system, vehicle aerodynamics, the atmosphere (GRAM, US62, and US76), mass properties, and winds, and it has algorithms that emulate flight software for guidance and navigation systems. The system can be run in both 3-DOF and 6-DOF simulation modes. 3-DOF mode amounts to perfect attitude control and is typically used when design additions or changes to the guidance algorithms are being evaluated. In the DSES federation configuration, MAVERIC is executed in full 6-DOF mode.

ARTEMIS is being developed by MSFC for use in the Ares System Integration Lab (SIL) hardware-in-the-loop test bed to test out Ares I avionics. ARTEMIS uses the Trick simulation environment developed by JSC and leverages off of Guidance, Navigation, and Control algorithms developed in the MAVERIC simulation. ARTEMIS is designed to model the Ares I from pre-launch through orbit insertion. There are multiple configurations of ARTEMIS – a non real-time, single platform version and a real-time, distributed version that eventually will interface with avionics hardware components. In the current DSES federation configuration, ARTEMIS is executed in the single platform configuration. The addition of ARTEMIS to the DSES configuration provides a ready pathway to allow the DSES configuration to evolve with higher fidelity Ares I configuration representations as the Ares SIL matures.

C. Orion (JSC)

Over the years, JSC has developed and used a number of simulation tools. Recently, divisions in the Engineering Directorate at JSC have been adopting a common simulation development tool: “The Trick Simulation Development Environment.”⁹⁻¹¹ Trick automates many of the tedious processes in constructing simulations and provides a number of generalized simulation capabilities. In addition, JSC has been assembling a collection of common models. These models, while hosted in the Trick environment, are not necessarily Trick dependent. This goal is to develop a suite of space systems models that can be shared across projects and facilities.

JSC DSES simulations are built using Trick and this collection of common models. One of these is the Advanced NASA Technology Architecture for Exploration Studies (ANTARES), a multi-purpose simulation supporting the Orion Guidance Navigation and Control (GN&C) team. ANTARES supports a wide variety of analyses, including requirements assessments, design trades, GN&C flight software evaluation and test, and pilot in the loop interactions. ANTARES supports computational and parametric studies as well as hardware and crew in the loop facilities. ANTARES is based on common model library architecture, leveraging off simulation development across several organizations at JSC. ANTARES forms the core of the JSC contribution to DSES Orion simulations, such as the Launch and Ascent simulations and the Orion/ISS rendezvous and docking simulation.

D. Launch Abort System (LaRC)

The Launch Abort System (LAS) simulation is being developed by the Simulation Development and Analysis Branch (SDAB) at the NASA Langley Research Center. The SDAB established the Langley Standard Real-Time Simulation in C++ (LaSRS++) framework in 1995. LaSRS++ is an object-oriented application framework for constructing continuous cyclic simulations. The framework was built from scratch using modern object-oriented programming and design techniques. LaSRS++ consists of a suite of software libraries that can be shared among developers, projects, and facilities. The framework provides all of the critical services required to perform simulations, allowing developers to focus only on vehicle model development. The benefits of using the common framework for simulation development include the increase in productivity and ease of maintenance due to a high degree of framework code reuse, cohesive software practices, and common coding standards for project software development.

The framework supports multiple, heterogeneous models in a simulation. While originally developed to support aircraft simulation, models are not restricted to aircraft and can represent any interactive item. Vehicle models are portable across simulators with little to no modifications to its “hardware interface” code. LaSRS++ was designed to be platform non-specific, and has been used on IRIX, Linux, Solaris, and Windows operating systems. See references¹²⁻²⁰ for detailed descriptions of the LaSRS++ framework and application software.

The LaSRS++ framework includes various world models such as Earth, Moon, and Mars for use in simulations. Several atmospheric models are also included in the framework such as the Global Reference Atmosphere Model (GRAM), Mars Global Reference Atmosphere Model (Mars-GRAM), and Marshall Engineering Thermosphere (MET). The framework supports real-time human-in-the-loop, hardware-in-the-loop, and parametric simulation and analysis. It also supports multiple simulation aspects such as Guidance, Navigation, and Controls (GNC), vehicle dynamics and behavior, trajectory analysis, vehicle performance assessment, crew displays, and flight deck avionics and layout for part task or end-to-end full mission simulation. The LaRC’s HLA interface software is being designed and developed to be independent of LaSRS++, making it available for potential reuse within other simulation architectures. The DSES project represents the first use of this developing technology.

The LaRC Launch Abort System (LAS) simulation for the DSES project is project-specific software derived from the generic spacecraft model of the LaSRS++ framework. The generic spacecraft model adds enhancements to the

LaSRS++ framework including staging and orbital mechanics calculation. Generic spacecraft models are composed of interconnected sub-models such as mass, propulsion, GNC, and aerodynamics. Mass models allow for the addition of mass properties to the spacecraft; and force models, such as rocket motors, add forces and moments to the spacecraft. The LAS simulation is composed of four force models (3 propulsion models and 1 aerodynamics model) and three mass models. The force and mass models provide simulation of the escape, pitch, and jettison motors. The aerodynamic force model applies aerodynamic forces and moments to the LAS. For the DSES project, the LAS simulation runs at 40 hertz and exchanges data with the Orion simulation at 4 hertz.

E. Crew Interaction (ARC)

The DSES team at ARC uses an in-house real-time operating system known as microTau, which was developed for high fidelity vertical motion simulation, including the Shuttle and lunar landings. The environment allows relatively fast development and deployment of new models and the addition of new computing hardware and force feedback inceptors. Simulation parameters can be changed on the fly. The overall system involves local Ethernet, SCRAMNET, and XIO memory sharing tools, and interfaces to other protocols and media.

ARC also has an extensive HLA environment bridging three facilities, adding different motion simulators, and a 360 degree panoramic control-tower-style visualization and control center to any distributed simulation. Ongoing work includes the addition, to the DSES network, of the Integrated Vehicle Health System (IVHS) and Advanced Diagnostics and Prognostics Testbed (ADAPT) Laboratories.

Using their expertise in building simulations in these systems, ARC built the crew-in-the-loop abort initiation portion of the DSES simulation. A two-step sequence was first developed, in accordance with existing spacecraft safety practices. This simulation avoids most of the probability of accidental activation. For purposes of demonstrating Integrated Vehicle Health and Navigation status, a graphical interface was later added, which displayed relevant state data and integrated an abort 'button'. For ease of use during integrated testing runs, and for consistency where required, a third feature was developed which automatically completes an abort condition after a preset amount of time should the crew be unable to respond. An interface feature currently under development aims to provide the crew with information about the status of distributed simulation elements, which for a 'real' simulation could indicate the state of any system appropriately modeled in the simulation and communication over the IP network.

VI. Future Work

The DSES team continues to increase the DSES capabilities by including the Orion and International Space Station (ISS) on-orbit operation and Orion Earth re-entry, descent and landing (EDL) in Fiscal Year 2007. While rendezvous has been demonstrated in the HTV FCT project, the DSES team has added hardware and human interaction into the simulation scope. The Orion simulation includes the vehicle systems and dynamics models, hand controller interface, and graphical views for pilot control. The ISS simulation includes the vehicle systems and dynamics models. This simulation segment will lead to end-to-end Orion ISS support mission profile which is depicted in Figure 7.

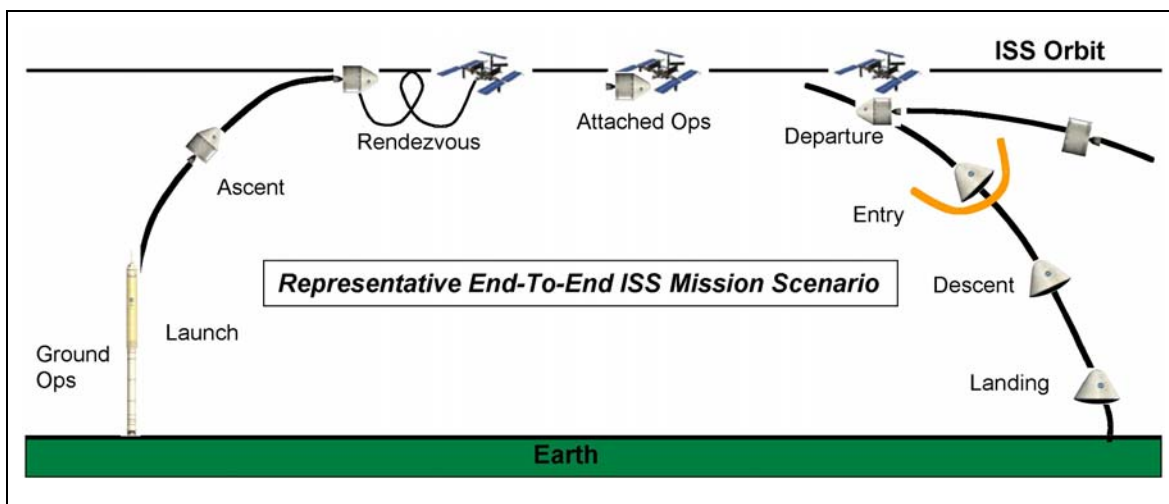


Figure 7. DSES Ares I/Orion/ISS Service Mission

Beyond Fiscal Year 2007, DSES will use the current DSES simulation as a base and extend the simulation segments to include lunar orbit and lunar landing simulation. These simulation segments will include the Command Module, Service Module, Lunar Surface Access Module (LSAM) Ascent Stage and Descent Stage. In turn, DSES will also verify extensibility of DSES concepts and software to support the end-to-end test and verification of the Constellation Program mission profile. Figure 8 is an artistic conception of the DSES Lunar Simulation Segment.



Figure 8. DSES Lunar Simulation Segment

VII. Concluding Remarks

The successful demonstration of the Orion and Ares I launch and ascent simulation showcased the significant benefits and potential of utilizing the Distributed Space Exploration Simulation (DSES) to provide an end-to-end simulation capability for the Constellation Program. For Fiscal Year 2007, this includes all mission segments for the Orion spacecraft (including Earth re-entry, descent, and landing) and its service missions to the International Space Station.

The DSES project is a collaborative project that currently includes 5 NASA centers: ARC, JSC, KSC, LaRC and MSFC. DSES utilizes and leverages technical domain expertise and resources from each of these centers to effectively support the Constellation Program with its simulation needs. The expected benefits include schedule efficiency and cost effectiveness in integrated mission simulation development, verification, validation, and accreditation that ultimately increases confidence in simulation results and fidelity. In the process, DSES provides a unifying approach to simulation across NASA and makes efficient use of NASA's simulation development resources. DSES also demonstrates the benefits of interoperability through distributed simulation by establishing agency wide capabilities through distributed yet integrated network and software; fostering program wide standards for simulation interoperability; and cultivating a framework for simulation and model interoperability and reuse.

As the Constellation Program advances, new mission segments will be added. This includes missions to return humans to the Moon. Ultimately, the Constellation Program will extend human presence beyond our local Earth/Moon system to our planetary neighbor Mars. As these and other mission segments evolve, DSES will evolve to provide the high fidelity mission simulation needs for the Constellation Program.

VIII. Acknowledgments

The authors would like to extend our appreciation and gratitude to the many talented scientists and engineers who have contributed to the DSES project: Paul Bielski, Dan Bowman, Mark Coffman, Chris Daum, Dan Dexter, Jim Gibson, David Hasan, Joe Hawkins, Missy Hill, Sean Kenny, Michael Madden, Jason Neuhaus, Marty Pethel, Al Ragsdale, Mark Ricci, Russ Sexton, Paul Sugden, and Jesse Trammell. These are the people who really make DSES work. It is their efforts that have ultimately contributed to the success of numerous demonstrations of the DSES Orion and Ares I Launch and Ascent simulation to the major stakeholders ranging from the various NASA Constellation Program Level 2 Systems Integration Groups to the Constellation Program Director.

IX. References

¹Simulation Interoperability Standards Committee (SISC) of the IEEE Computer Society, “IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA) — Framework and Rules”, The Institute of Electrical and Electronics Engineers, 3 Park Avenue, New York, NY 10016-5997, USA, 11 December 2000, ISBN 0-7381-2620-9 SS94882.

²Simulation Interoperability Standards Committee (SISC) of the IEEE Computer Society, “IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA) Federate Interface Specification”, The Institute of Electrical and Electronics Engineers, 3 Park Avenue, New York, NY 10016-5997, USA, 9 March 2001, ISBN 0-7381-2622-5 SS94883.

³Simulation Interoperability Standards Committee (SISC) of the IEEE Computer Society, “IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA) Object Model Template (OMT) Specification”, The Institute of Electrical and Electronics Engineers, 3 Park Avenue, New York, NY 10016-5997, USA, 9 March 2001, ISBN 0-7381-2624-1 SS94884.

⁴Simulation Interoperability Standards Committee (SISC) of the IEEE Computer Society, “IEEE Recommended Practice for High Level Architecture (HLA) Federation Development and Execution Process (FEDEP)”, The Institute of Electrical and Electronics Engineers, 3 Park Avenue, New York, NY 10016-5997, USA, 23 April 2003, ISBN 0-7381-3585-2 SS95088.

⁵Lauderdale, G., Crues, E., Snyder, D., and Hasan, D., “A Feasibility Study for ISS and HTV Distributed Simulation,” AIAA Modeling and Simulation Technologies Conference and Exhibit, Austin, Texas, 11-14 August 2003.

⁶Lauderdale, G., Crues, E., Snyder, D., and Hasan, D., “Further Studies On The Feasibility Of A Distributed ISS and HTV Simulation,” Proceedings of the Fall 2003 Simulation Interoperability Workshop and Conference, Orlando, Florida, 15-18 September 2003.

⁷Crues, E., Blum, M., Chung, V., and Bowman, J., “The Distributed Space Exploration Simulation (DSES),” Proceedings of the Spring 2007 Simulation Interoperability Workshop and Conference, Norfolk, Virginia, 25-30 March 2007.

⁸Phillips, R., Dexter, D., and Crues, E., “A Coordinated Initialization Process for the Distributed Space Exploration Simulation (DSES),” Proceedings of the Spring 2007 Simulation Interoperability Workshop and Conference, Norfolk, Virginia, 25-30 March 2007.

⁹Vetter, K., and Hua, G., “The Trick User’s Guide 2005.7.0 Release,” NASA Technical Publication, NASA Johnson Space Center, June 2006.

¹⁰Vetter, K., “Trick Simulation Environment – User Training Material 2005.7.0 Release,” NASA Technical Publication, NASA Johnson Space Center, June 2006.

¹¹Paddock, E. J., Lin, A., Vetter, K., and Crues, E. Z., “Trick: A Simulation Development Toolkit,” AIAA Modeling and Simulation Technologies Conference and Exhibit, Austin, TX, 2003, AIAA 2003-5809.

¹²Madden, M. M., “Architecting a Simulation Framework for Model Rehosting,” AIAA Modeling and Simulation Technologies Conference, Providence, Rhode Island, August 16-19, 2004, AIAA 2004-4924.

¹³Neuhaus, J. “Modeling Mass Properties in an Object-Oriented Simulation,” AIAA Modeling and Simulation Technologies Conference, Providence, Rhode Island, August 16-19, 2004, AIAA 2004-5166.

¹⁴Madden, M. M., and Neuhaus, J. “A Design for Composing and Extending Vehicle Models”, AIAA Modeling and Simulation Technologies Conference, Austin, Texas, August 11-14, 2003, AIAA-2003-5458.

¹⁵Kenney, P. S., “Simulating the ARES Aircraft in the Mars Environment”, AIAA Modeling and Simulation Technologies Conference, Austin, Texas, August 11-14, 2003, AIAA-2003-6579.

¹⁶Madden, M. M. “Examining Reuse in LaSRS++-Based Projects”, AIAA Modeling and Simulation Technologies Conference, Montreal, Canada, Aug. 6-9, 2001, AIAA-2001-4119.

¹⁷Neuhaus, J. “An Object-Oriented Sensor and Sensor System Design,” AIAA Modeling and Simulation Technologies Conference, Montreal, Canada, Aug. 6-9, 2001, AIAA-2001-4123.

¹⁸Madden, M. M.; Glaab, P. C.; Cunningham, K.; Leslie, R. A.; Kenney, P. S.; and Geyer, D. W.: “Constructing a Multiple-Vehicle, Multiple-CPU Simulation Using Object-Oriented,” AIAA Modeling and Simulation Technologies Conference, Boston, Massachusetts, Aug. 9-11, 1998, AIAA-98-4530.

¹⁹Leslie, R. A.; Geyer, D. W.; Cunningham, K.; Glaab, P. C.; Kenney, P. S.; and Madden, M. M.: “LaSRS++ - An Object-Oriented Framework for Real-Time Simulation of Aircraft,” AIAA Modeling and Simulation Technologies Conference, Boston, Massachusetts, Aug. 9-11, 1998, AIAA-98-4529.

²⁰Chung, V. I., and Hutchinson, B. K., “A Unique Software System For Simulation-To-Flight Research,” AIAA Modeling and Simulation Technologies Conference & Exhibit, Montreal, Canada, August 6-9, 2001, AIAA 2001-4057.