

# Evolution of A Distributed Live, Virtual, Constructive Environment for Human in the Loop Unmanned Aircraft Testing

James R. Murphy\*

*NASA Ames Research Center, Moffett Field, CA, 94035*

Neil M. Otto†

*Flight Research Associates, Moffett Field, CA, 94035*

NASA's Unmanned Aircraft Systems Integration in the National Airspace System Project is conducting human in the loop simulations and flight testing intended to reduce barriers associated with enabling routine airspace access for unmanned aircraft. The primary focus of these tests is interaction of the unmanned aircraft pilot with the display of detect and avoid alerting and guidance information. The project's integrated test and evaluation team was charged with developing the test infrastructure. As with any development effort, compromises in the underlying system architecture and design were made to allow for the rapid prototyping and open-ended nature of the research. In order to accommodate these design choices, a distributed test environment was developed incorporating Live, Virtual, Constructive, (LVC) concepts. The LVC components form the core infrastructure support simulation of UAS operations by integrating live and virtual aircraft in a realistic air traffic environment. This LVC infrastructure enables efficient testing by leveraging the use of existing assets distributed across multiple NASA Centers. Using standard LVC concepts enable future integration with existing simulation infrastructure.

## Nomenclature

<i>ADRS</i>	=	Aeronautical Data Link and Radar Simulator
<i>ATC</i>	=	Air Traffic Control
<i>HLA</i>	=	High Level Architecture (simulation middleware)
<i>LVC</i>	=	Live, Virtual, Constructive describing the simulation environment
<i>MACS</i>	=	Multi-Aircraft Control System (a pilot and air traffic emulator)
<i>MPI</i>	=	Multi-Purpose Interface
<i>NAS</i>	=	National Airspace System
<i>RUMS</i>	=	Remote User Monitoring System
<i>SAA</i>	=	Sense and Avoid
<i>SAAProc</i>	=	Software container for the Sense and Avoid algorithms
<i>UAS</i>	=	Unmanned Aircraft System
<i>VSCS</i>	=	Vigilant Spirit Control Station
<i>XML</i>	=	Extensible Markup Language (programming language)

---

\* Project Engineer, Aviation Systems Division, MS 243-1.

† Integration Manager, Aviation Systems Division, MS 243-1.

## I. Introduction

THE National Aeronautics and Space Administration (NASA) is conducting research under the Unmanned Aircraft Systems (UAS) Integration in the National Airspace System (NAS) Project (hereby known as UAS-NAS) to investigate technologies and collect evidence supporting the definition of standards that will enable routine UAS access to the NAS. The UAS-NAS project has two primary technical challenges supporting this access, UAS Command and Communications (C2) and Detect and Avoid (DAA).<sup>1</sup> With the guidance of UAS and air traffic control (ATC) subject matter experts, the Federal Aviation Administration (FAA), and industry, the project developed a series of simulations and flight tests to collect the appropriate data in support of defining the standards used in these two areas. In support of these planned data-gathering activities, the UAS-NAS project also stood up the integrated test and evaluation (IT&E) team, charged with development of the test infrastructure and execution of the large-scale integrated events.<sup>2</sup> While the C2 simulation and flight test system architecture were identified early-on as needing little system integration, the DAA testing plan including pilot and air traffic control participants and interaction with aircraft and the unmanned vehicles ground control station (GCS). Hence the IT&E infrastructure development focused on the DAA system test requirements.

Based on the known DAA simulation and flight test requirements the IT&E team designed a distributed live, virtual, and constructive (LVC) test environment for the underlying system infrastructure. LVC environments are widely used by the Department of Defense and throughout the aviation industry.<sup>3,4,5</sup> By modeling the UAS-NAS test system off of an LVC paradigm, the project was able to leverage lessons learned from the DoD and industry concepts as well as utilize NASA's existing LVC technologies.<sup>6</sup> The UAS-NAS LVC distributed environment (known as LVC-DE) is comprised of ATC workstations, aircraft simulators, live aircraft, and unmanned aircraft GCS that, operating together, provide researchers with a relevant NAS environment to test unmanned systems. In order to maximize the use of available resources, the LVC test environment is designed for distribution, enabling technologies developed by researchers and external partners to be integrated into the test environment. This underscores the two driving requirements for the system: the LVC must be flexible to support the integration of technologies as needed for data collection, and the LVC support distribution across NASA Center facilities to allow for integration of test assets (e.g. ATC facilities, aircraft test ranges) where they were located.

This paper documents the development of the LVC test environment used by the UAS-NAS project for its simulations and flight tests. It provides a description of the evolution of the underlying LVC infrastructure as it matured from the initial concept into the system to be used for the second phase of the project. Lastly, it documents the plans to migrate the software to enable its integration with future LVC research platforms.

## II. Background

### A. Stakeholders

RTCA (formally known as Radio Technical Commission for Aeronautics) was chartered by the FAA to operate advisory committees that develop solutions to real-world air transportation problems.<sup>7</sup> In order to safely integrate the multitude of UAS platforms into non-segregated airspace, the FAA and UAS stakeholders have determined that both a robust DAA and a robust and secure C2 Data Link capability need to be established. In response, the FAA established the Unmanned Aircraft Systems Integration Office to support integration of UAS safely and efficiently into the NAS. In addition, RTCA formed Special Committee 228 to develop the Minimum Operational Performance Standards (MOPS) for DAA equipment, with emphasis in an initial phase of standards development on civil UAS equipped to operate into Class A airspace under IFR flight rules. Support for the RTCA SC-228 DAA MOPS are the focus of the LVC test environment development.

Phase 1 DAA MOPS were released on 14 July 2017. The Operational Environment for the Phase 1 is the transitioning of a UAS to and from Class A or special use airspace, traversing Class D and E, and perhaps Class G airspace. The MOPS included characteristics of the air-to-air radar characteristics, requirements for the DAA algorithm, pilot display guidance, and the definition of "Well Clear" as it pertains to its application within the DAA algorithm.

Phase 2 of the DAA MOPS is building on the standards developed during Phase 1, extending to operations in Terminal airspace, characteristics of lower size, weight, and power (SWaP) air-to-air sensors, ground based detect and avoid (GBDAA) and whether those have an impact on the Phase 1 "Well Clear" definition.

### B. LVC Concept

The LVC portion of the test environment refers to the components of the test that can be regarded as “live”, “virtual”, or “constructive”. A “constructive” simulation generally has no interactive human involvement in simulated conditions. Instead, scenarios unfold using rule-based decisions that control the interactions between simulated actors. “Virtual” simulations involve human participants operating simulated systems (e.g. a pilot flying a flight simulator). A “live” test environment involves human participants operating real systems.<sup>3</sup>

Figure 1 provides a high-level concept of operations for the LVC-DE test environment developed for project simulations and flight tests. The LVC environment enabled the test engineers to integrate existing ATC workstations and simulation infrastructure resident at NASA Ames Research Center with the flight of aircraft in the restricted airspace at Edwards Air Force Base (where NASA Armstrong Flight Research Center resides). The underlying LVC infrastructure provides the connectivity between the distributed facilities and the mechanism for integrating the data from the disparate systems together into a single system. The LVC usage of abstracted integration through a well-defined interface hides the original source of the data (whether from a live aircraft or virtual flight simulator) feeding the data to ATC and pilot displays as well as DAA algorithms.

While the live, virtual, and constructive components of a test environment only comprise a portion of what is required to run a simulation or flight test, the test environment is widely known as an LVC.<sup>‡</sup> Typical LVC core functionalities are described in the next section.

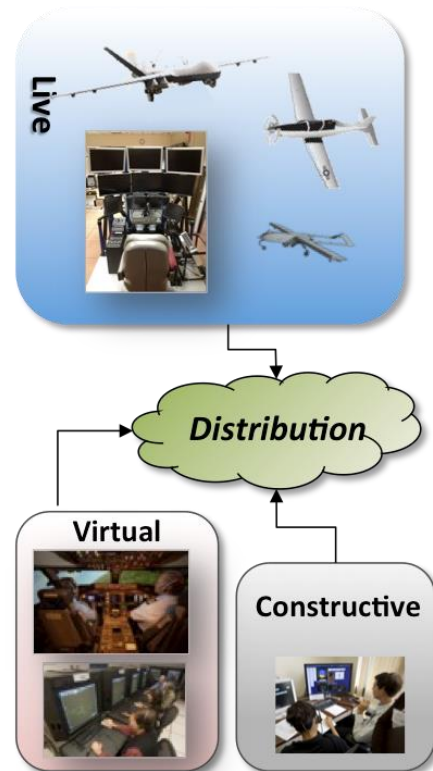
### C. LVC Core Components

#### 1. LVC Middleware

LVC Middleware can be considered the backbone of an LVC system. It provides the messaging distribution among the various client software. For instance, aircraft position data provided by a live aircraft, or flight simulator would be routed to the appropriate ATC workstation or pilot display via the LVC middleware messaging capability. The LVC middleware also supports a “publish/subscribe” capability allowing for targeted routing of data through the system. The LVC Lab at NASA Ames uses the IEEE 1516 standard Pitch portable Run Time Infrastructure High Level Architecture and Federation Object Model middleware to provide the message routing among the distributed facilities.<sup>8</sup> These processes were inherited from NASA’s Virtual Airspace Simulation Technology RealTime (VAST-RT) simulation infrastructure and allowed for rapid prototyping of the initial LVC-DE.<sup>9</sup> Figure 2 shows an expanded the LVC-DE concept that utilizes the LVC middleware. Other LVC middleware solutions include DoD’s Distributed Interactive Simulation<sup>10</sup>, AviationSimNet<sup>11</sup>, Test and Training Enabling Architecture<sup>12</sup>, and Data Distribution Service (DDS)<sup>13</sup>.

#### 2. HLA Toolboxes

While the HLA has a well-defined message interface, each software component connecting to the LVC environment for a simulation may have its own legacy message interface. Toolboxes translate messages from software components to comply with the defined HLA interface. There were two primary reasons for the use of Toolboxes instead of developing the interface directly in the component software: the software component may be commercial or government off-the-shelf (i.e. the development team does not control the software in order to implement the interface); and the software component may be used to connect to multiple different versions of middleware. The LVC environment utilizes Toolboxes to connect to constructive target generators, the LVC Gateway, and flight simulators.



**Figure 1. LVC Environment Concept of Operations. An LVC environment promotes the integration of multiple live and virtual data sources.**

<sup>‡</sup> It should be noted that categorizing components of a simulation as live, virtual, or constructive can be problematic. Since the degree of human participation in a simulation is widely variable, as is the degree of equipment realism, there is no clear division between these categories.

### 3. LVC Gateway

The LVC Gateway provided connectivity to an external software component where connecting directly to the HLA environment was not desired. Instead of each connecting remotely to the HLA, an LVC Gateway was used to route local message traffic and provided a single connection from the remote facility to the HLA. Components connecting to the LVC Gateway published messages according to the LVC interface control document. Any number of sites can be added to the LVC environment by connecting additional LVC Gateways to the architecture.<sup>14</sup> Figure 2 also shows the usage of the LVC Gateway connecting clients at a remote facility.

## III. LVC Development – Phase 1

### A. LVC Characterization

The first instantiation of the LVC –DE was used to characterize the distributed network latencies. One of the goals of the LVC environment was to provide a simulation infrastructure that emulates an operational air traffic control environment.

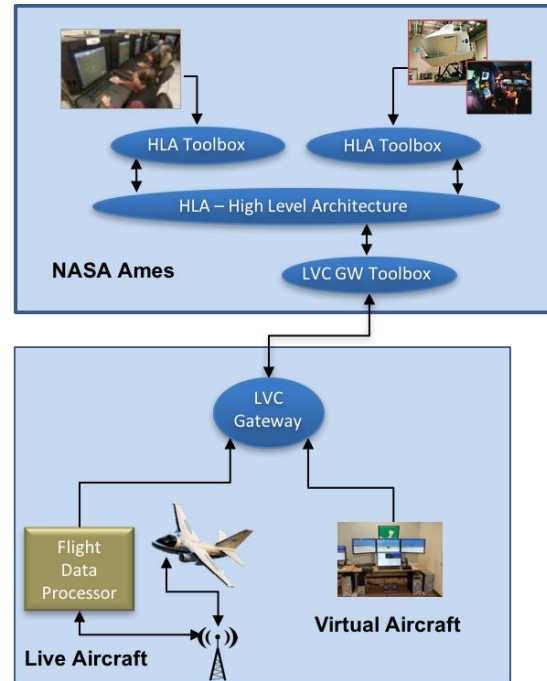
In order to inform the development of the future UAS flight test environment, the initial LVC environment characterization tests had two primary objectives:

- 1) Measure the latency of sending aircraft position updates from the source to the LVC Gateway
- 2) Measure the latency of sending aircraft position updates between LVC networked facilities.

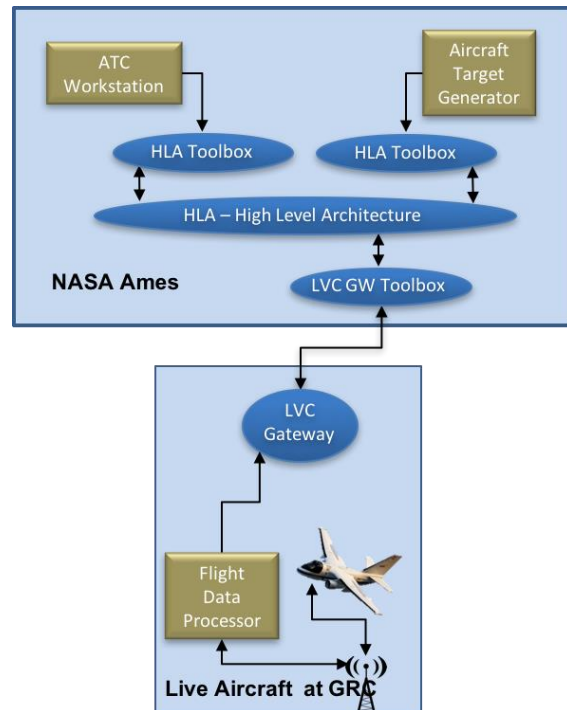
Figure 3 shows the LVC architecture and results for an example data collection run. A simplified version of the LVC-DE was used to connect NASA Ames and NASA Dryden (ne Armstrong) These data, along with the characterization between each of the candidate test locations provided a general understanding of the system in terms of its ability to transmit the appropriate data in a timely manner. For a detailed description of the characterization testing and results, please see the LVC-DE Characterization Report.<sup>15</sup>

### B. Integrated Human in the Loop Simulation

The integrated Human in the Loop (IHITL) simulation provided data to the UAS researchers to evaluate the state of the simulation environment development by integrating and testing key DAA technologies into a research GCS. The technical goals for the IHITL were to: 1) evaluate and measure the effectiveness and acceptability of DAA systems (algorithms and displays) to inform and advise UAS pilots; and 2) evaluate and measure the interoperability and operational acceptability of UAS integration concepts for operating in the NAS. A third Project goal was to characterize the simulation and test environment in order to evaluate the state of the simulation



**Figure 2. High-level LVC Middleware Connectivity.** The usage of LVC middleware to route data through the LVC environment.



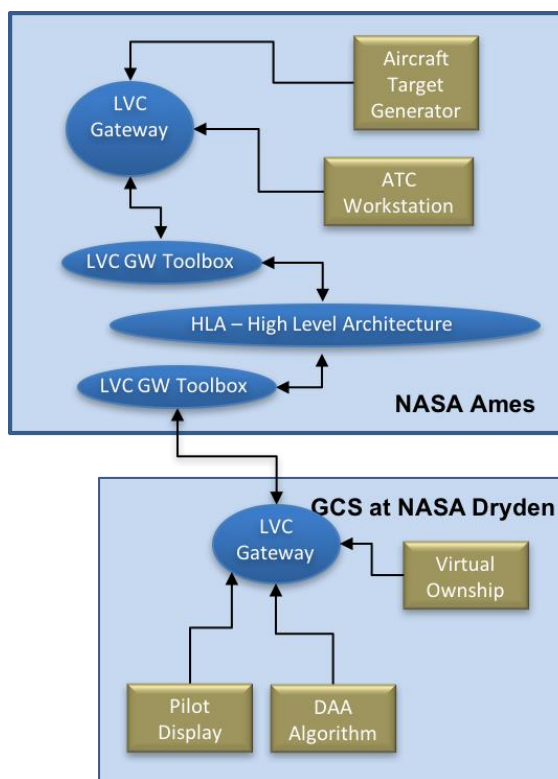
**Figure 3. LVC Characterization Architecture.** The usage of LVC middleware to route data through the LVC environment.

architecture with respect to future UAS research activities. Figure 4 shows the expanded LVC-DE architecture used to facilitate the IHITL testing. Of interest for this discussion is the integration of the DAA algorithm and the GCS into the LVC environment.

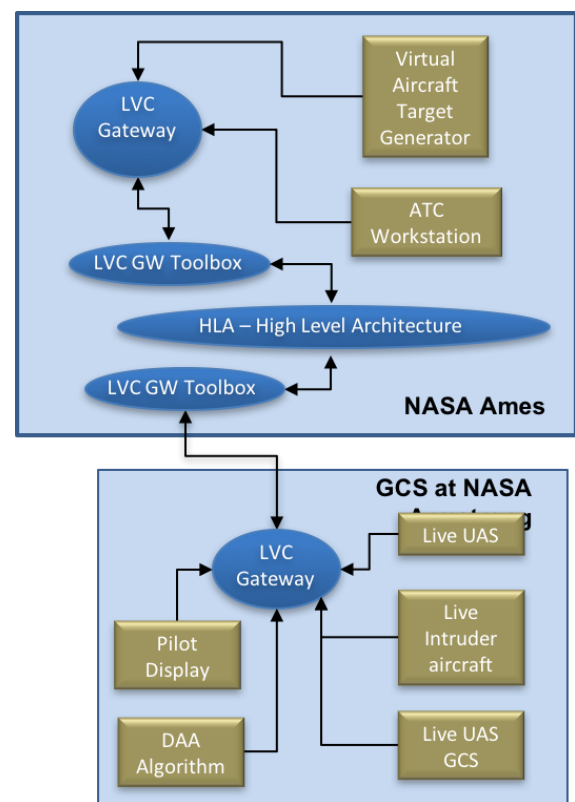
### C. DAA Flight Testing

The integrated flight testing were data collection efforts that supported the draft and final RTCA SC-228 DAA MOPS. The flight testing events were designed to enable collection of data in a realistic operating environment, including the inherent uncertainties of real winds and on-board sensors. However, since the testing includes the flight of unmanned aircraft, which cannot presently fly in the NAS without restrictions and waivers from the FAA, the integrated test team developed a distributed environment that combines live, virtual, and constructive (or background) traffic and intercept scenario to promote the safe testing of the concepts and technologies.

This combined live/virtual flight test architecture was used during the first integrated flight test (known as Flight Test 3 within the UAS-NAS project). Figure 5 shows the further expansion of the LVC-DE architecture to incorporate both virtual and live intruder aircraft along with the live UAS flight. From the perspective of the Air traffic controller participants and the UAS pilot, the “real” live intruder aircraft were indistinguishable from the virtual intruder aircraft.



**Figure 4. LVC Human in the Loop simulation Architecture.** The integration of client algorithm and display software into the core LVC environment.



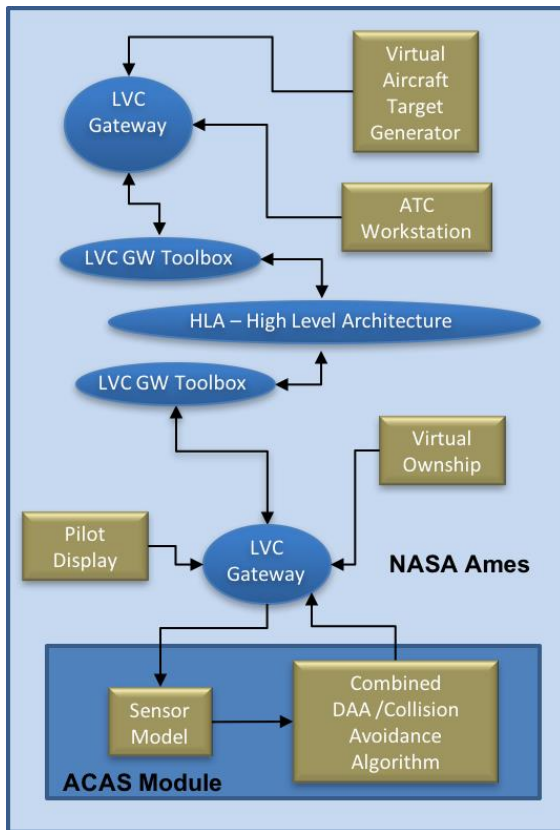
**Figure 5. LVC Integrated Flight Test Architecture.** The integration of live and virtual traffic into a single system.

## IV. LVC Development – Phase 2

### A. ACAS Integration

One of the early LVC integration tasks with respect to the second Phase of the UAS-NAS project was to integrate the Airborne Collision Avoidance System (ACAS) software. ACAS X is a proposed replacement for the Traffic Alert and Collision Avoidance System (TCAS) for manned aircraft ACAS Xu is the unmanned aircraft variant. ACAS Xu is actively developing a combined Collision Avoidance and DAA alerting capability. The ACAS software was provided to the IT&E development team for integration into the LVC environment for simulation. Along with the ACAS software, the team also integrated sensor uncertainty models to enhance the simulation realism.

Since the ACAS software is run on-board the UAS aircraft operationally, where it inherently receives live sensor data (with real uncertainty), the LVC developers decided to integrate the uncertainty models with the ACAS software into a single model. Figure 6 shows the LVC architecture encompassing this design.



**Figure 6. LVC ACAS Xu Simulation Architecture.** The integration of the ACAS software and sensor uncertainty model.

### B. SMART NAS Integration

Looking towards the need for integration into other research platforms, the IT&E development team is working closely with the Shadow Mode Assessment Using Realistic Technologies for the National Airspace System (SMART-NAS) Test Bed engineers to support future integration. The SMART-NAS Test Bed integrates the LVC capabilities of the other projects with additional advanced simulation capabilities that enable high-fidelity simulation of integrated ATM concepts.

The UAS-NAS LVC-DE would benefit from a SMART-NAS Test Bed integration by inheriting the system-wide airspace and decision support tool systems the Test Bed has already subsumed. Alternatively, the Test Bed would gain access to the UAS aircraft and pilot display interfaces already incorporated into the LVC-DE. The SMART-NAS Test Bed system is relatively new, and is using advanced LVC concepts in its architecture and design. Hence the UAS-NAS project is looking at migrating its LVC design to leverage these new concepts. The first step is to migrate the LVC-DE towards the use of DDS middleware (and away from HLA and separate LVC Gateways). This would allow the LVC-DE and the Test Bed to immediately pass messages with each system without the need for toolboxes or message translation processes.

## V. Conclusions and Next Steps

TBD.



## References

- 
- <sup>1</sup> National Aeronautics and Space Administration, NASA Technology Development Project Plan, Unmanned Aircraft Systems (UAS) Integration in the National Airspace System (NAS), 31 January 2013
- <sup>2</sup> Murphy, J. R., Hayes, P. S., Kim, S. K., Bridges, W., and Marston, M., "Flight Test Overview for the UAS Integration in the NAS Project," AIAA Paper 2016-1756, January 2016
- <sup>3</sup> DoD: "Modeling and Simulation Master Plan", DoD 5000.59P, Oct 1995
- <sup>4</sup> Henninger, Amy E., Cutts, Dannie, Loper, Margaret, et al, "Live Virtual Constructive Architecture Roadmap (LVCAR) Final Report", Institute for Defense Analysis, Sept, 2008
- <sup>5</sup> Bezdek, W. J., Maleport, J., Olshon, R., "Live, Virtual & Constructive Simulation for Real Time Rapid Prototyping, Experimentation and Testing using Network Centric Operations," AIAA 2008-7090, AIAA Modeling and Simulation Technologies Conference and Exhibit, August 2008
- <sup>6</sup> Lutz, R., LeSueur, K., Fast, P., Graeff, R., Simolte, A., Rutledge, J., and Motti, R., "A Persistent LVC Simulation Environment for UAS Airspace Integration," The Interservice/Industry Training, Simulation & Education Conference (IITSEC), December 2010.
- <sup>7</sup> RTCA website: <http://www.rtca.org>, June 2017
- <sup>8</sup> Software website: <http://www.pitch.se/prti>, 25 June 2017
- <sup>9</sup> Lehmer, R., and Malsom, S., "Distributed System Architecture in VAST-RT for Real-Time Airspace Simulation," American Institute of Aeronautics and Astronautics (AIAA), 2004.
- <sup>10</sup> "1278.1-2012 IEEE Standard for Distributed Interactive Simulation - Application Protocols". *IEEE*. Retrieved 24 June 2017
- <sup>11</sup> AviationSimNet Standards Working Group, "AviationSimNet Specification", Version 2.2, MITRE Corporation, June 2010
- <sup>12</sup> Edward T. Powell, Jason Lucas, Kurt Lessmann, and George J. Rumford, "The Test and Training Enabling Architecture (TENA) 2002 Overview and Meta-Model," published in the Proceedings of the Summer 2003 European Simulation Interoperability Workshop, 03E-SIW-041, found at <http://www.sisostds.org/>
- <sup>13</sup> "Data Distribution Service (DDS), Version 1.0". Object Management Group. December 2, 2004. Retrieved June 25, 2017
- <sup>14</sup> Murphy, J. Jovic, S., Otto, N., "Message Latency Characterization of a Distributed Live, Virtual, Constructive Simulation Environment," AIAA Infotech@Aerospace Conference, January 2015.
- <sup>15</sup> NASA: "Live, Virtual, Constructive Distributed Test Environment: Characterization Report," NASA UAS-04.05.0001.01, September, 2013.