C4I-Simulation Interoperability Using the DII COE and HLA

Gene Layman, PhD.
Naval Research Laboratory
4555 Overlook Ave. S.W.
Washington, DC 20375-5000
layman@ninja.nrl.navy.mil

John Daly
ITT Industries
Naval Research Laboratory
Washington, DC 20375-5000
202-763-6766
jjdaly@ninja.nrl.navy.mil

Zach Furness
The MITRE Corporation
1820 Dolley Madison Blvd
McLean, VA, 22102
703-883-6614
zfurness@mitre.org

Jennie Womble
ITT Industries
Naval Research Laboratory
Washington, DC 20375-5000
202-763-9000
womble@ninja.nrl.navy.mil

Keywords: C4I, DII COE, HLA RTI, C4I Federates, Interoperability, NSS, JTLS, Pegasus, GCCS

ABSTRACT: *Technologies and methods have been developed within C4I systems that permit them to function as federates using the High Level Architecture (HLA). The HLA Runtime Infrastructure (RTI) has been shown to run successfully on C4I system hardware that is based on the Defense Information Infrastructure Common Operational Environment (DII COE). The most prominent example to date has been the operation of the RTI with the Global Command and Control System (GCCS) and GCCS/Maritime that both utilize the DII COE. The GCCS HLA interface has been used successfully with simulations such as the Joint Theater Level Simulation (JTLS), the Navy Simulation System (NSS), and the Pegasus Federation. These federations span the range of potential military applications from training, to experimentation, planning, and course of action (COA) analysis. This paper provides an overview of the various federation applications in which GCCS has been used to date, and also discusses the benefits of using HLA and the DII COE to improve C4I-Simulation interoperability.*

1. Background

The Joint Technical Architecture (JTA) [1] requires adoption of the Defense Information Infrastructure Common Operating Environment (DII COE) for DoD operational C4I systems and recommends use of the High Level Architecture (HLA) [2] to achieve interoperability with other systems and simulations.

The problem of interoperatability between simulations and C4I has been thoroughly studied and documented. [3] In the past, most interoperability approaches to link simulations with C4I systems have focused on using unique interfaces between the two components. This has in turn led to a plethora of C4I-Simulation interfaces that are costly to maintain and tend to replicate functionality across many of the interfaces. The advantage of such approaches is that they are often easy to implement and only need be concerned with the interface requirements of the immediate C4I system and simulation. The disadvantage is that a single interface may not be extensible to other C4I systems and simulations.

The HLA provides a general-purpose architecture that promotes simulation reuse and interoperability. An HLA federation is a collection of simulations (i.e., federates) or systems connected together via the HLA Runtime Infrastructure (RTI). A more robust approach to C4I-Sim interoperability is to configure a C4I system to function as a federate within an HLA federation allowing transactions between any numbers of participating simulations. A single HLA-C4I interface could satisfy interoperability requirements between several simulations, assuming that the data requirements were properly represented in the Federation Object Model (FOM).

The DII COE provides the core software for modern C4I systems such as: the Global Command and Control System (GCCS), the Navy Maritime Global Command and Control System (GCCS-M), the Army Global Command and Control System (AGCCS), and the Air Force Theater Battle Management Core Systems (TBMCS). These systems use the common DII COE software and add Mission Application software unique to the host C4I System.

This paper describes the integration of the HLA RTI with DII COE-based C4I systems. Integrating the RTI software into the DII COE, and providing additional interoperability functions will enable C4I systems and simulations to enjoy many of the potential benefits that both architectures have to offer. The implementation and design approaches are described here along with a short description of HLA federation developments that have included C4I federates.

The Defense Modeling and Simulation Office (DMSO) sponsored much of this work to date. Although the majority of the work described in this paper was developed for interoperability between C4I systems and training simulations, the technology has been extended to the analysis, experimentation, and operational domains. It is being used in the development of GCCS Embedded Training [4] under Defense Information Systems Agency (DISA) sponsorship, and in the GCCS Embedded Simulation Infrastructure Program [5] sponsored by the Navy Modeling and Simulation Management Office (NAVMSMO) and DMSO.

2. Legacy Simulation-C4I Connectivity

In the past, simulations were generally restricted to interface with C4I systems using standard messages through normal C4I communications channels. These channels were intended for communication with other C4I systems, not simulations. This technique produced severe restriction on the flexibility of the interface, and assumed (correctly in some cases) that the software of the C4I systems could not be modified to accommodate more robust interfaces with simulations for technical reasons and rigid configuration control.

The resulting simulation-C4I system interfaces were restricted to “stimulating” off-line C4I systems with one-way communications with no intrinsic feedback. Non-real-time operations were difficult, if not impossible, to achieve and simulation exercise controls were unavailable within the C4I systems.

Individual “point-to-point” interfaces for two specific systems generally can be effective for the purpose intended but do not lend themselves to reuse. Attempts at a “universal” interface to configure all data interactions to/from a simulation and a C4I system have met with mixed success. These efforts culminated in the DMSO-sponsored effort in 1996 to produce the Modular Reconfigurable C4I Interface (MRCI). [6]

3. The DII COE

The DII COE [7] provides a configuration managed software environment for C4I systems to draw upon in developing tailored applications. By creating an environment in which different C4I systems utilize common components (such as map displays, message processors, etc.) redundancies between different C4I systems are eliminated and interoperability between systems is improved.

This architecture provides the possibility of performing DII COE system level modifications that can vastly improve the ability of DII COE compliant C4I systems to interoperate with simulations. The standardization of C4I computer software architecture allows these systems to utilize common components and reduces the need of these systems to develop unique components to accommodate interoperability.

Figure 1 shows the main elements of a DII COE based C4I system. At the top layer, the C4I system specific Mission Applications will tend to maintain functionality that is not widely used by other applications. However, within the COE exist a number of software elements ("segments") that are available for various applications to leverage for their particular use.

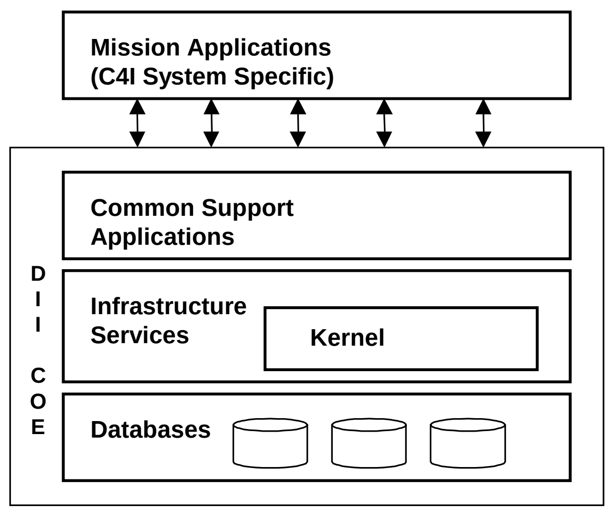

Figure 1. The DII COE

The DII COE segments are organized into Kernel, Infrastructure Services and Common Support Applications as shown. They include such general services as: message processing, office automation, data access services, and security, as well as more C4I specific applications such as map servers, force track management, communications, message generation and tactical decision aids.

4. Migration to HLA Connectivity

Research into C4I-Simulation interfaces has highlighted the need to move beyond merely stimulating a C4I system by having the simulation mimic another C4I system. There is significant benefit to the direct exchange of data objects between simulations and C4I, and the ability to insert data directly into the C4I system databases. Primary requirements for C4I-Simulation interfaces are: realism, distributed control of simulated data, and the ability of the C4I system to be on-line performing normal operational functions.

These requirements were recognized early on by the Naval Research Lab (NRL) and the Defense Advanced Research Projects Agency (DARPA) in the Synthetic Theater of War (STOW) 97 project. [8] An interface development between the GCCS DII COE and the STOW HLA architecture suggested the answer to this problem could be to configure the C4I system to function as an HLA federate. This would leverage off of the existing use of HLA in the exercise and eliminate the need for yet another stove-piped interface.

This new approach was the opposite of the then current “stimulation” interoperability paradigm. Instead of making the simulation and interface combination “look” like another C4I system, why not make the C4I system “look” like another simulation on the LAN?

In order for this interoperability protocol to be robust and reusable, an expansive set of rules and supported interactions is required. Additionally, this protocol needs to be readily available, supported, and familiar to simulation developers. The HLA provides many of these guiding principles.

To enable a C4I system to interoperate directly with an external simulation via the HLA, some core software modifications to the C4I system are required. Additional services are necessary to facilitate advanced uses of simulated data within the C4I systems.

A significant step in the development of standardized C4I Federates is facilitated by the integration of the HLA into the DII COE. Utilizing the rules and conventions in the HLA, and adding specific applications and services to the DII COE for simulation support and interoperability, removed traditional C4I interface restrictions and makes extensive C4I-Simulation interoperability achievable.

5. Benefits of HLA for C4I-Simulation Interoperability

By utilizing the HLA, C4I systems can gain the same benefits that simulations have such as; easy access to a wealth of public data that is already being exchanged between simulations over a well documented and standardized interface.

The effort required to expand the set of data to be exchanged between new simulations in an HLA federation involving C4I systems will generally be less than if the interface is developed from scratch. These advantages can be seen in Figure 2. The C4I system, acting as a Federate ("C4I System A"), has access to all of the data made available over the RTI while the C4I system utilizing it's own point-to-point interface ("C4I System B"), may only have access to a subset of that data. If C4I System A represents the Common Operational Picture (COP) of a joint exercise, these advantages are immediately apparent.

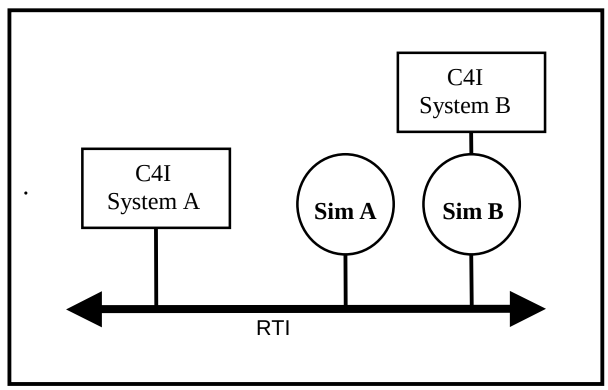

Figure 2. C4I/Simulation Federation

Additionally, a C4I federate that is part of an operational C4I network can capture real world tactical situations from which a simulation might be initialized. This may be very important if the simulation is used for running Course of Action (COA) analysis and needs to be initialized repeatedly over the duration of an exercise. The C4I federate may also archive, replay and analyze the simulated COP, generate commands, and provide a wide variety of C4I management, analysis, and decision support tools that can be used to enhance the overall federation.

6. DII COE Simulation Services

The DII COE HLA implementation consists of two software segments: the RTI Segment and the C4I Ambassador Segment. A schematic architectural diagram is shown in Figure 3. The RTI Segment is scheduled to become part of the standard release for DII COE during the year 2001.

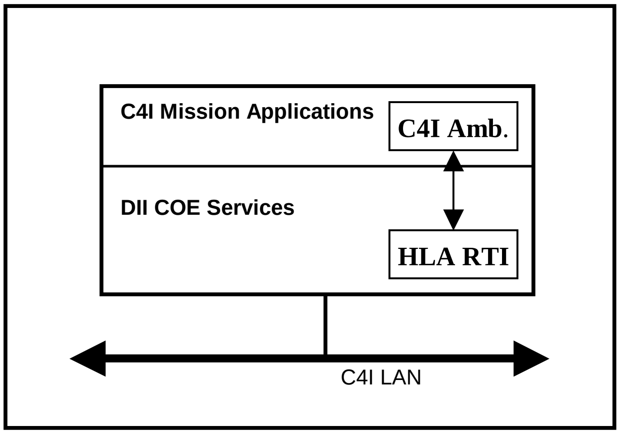

Figure 3. C4I Federate Architecture

The RTI provides standard network and data management services but does not process data content. The C4I Ambassador processes that data and interacts with the C4I system databases, DII COE services, and mission applications at DII COE level 7 compliance.

6.1 The RTI Segment

The RTI Segment fundamentally uses the DoD standard released RTI software. It is being reorganized to meet DII COE software organization and naming conventions, as well as slightly modified to meet special requirements of all COE software segments.

The use of a standard RTI will isolate any DII COE or Mission Application changes from the RTI Segment. Required modifications will be restricted to the C4I Ambassador or other application utilizing the RTI. Likewise, upgrades to the RTI can be implemented as available, with consideration of the lead-time in the COE configuration management process.

The RTI interfaces with the DII COE at the Kernel level. Specifically, the RTI depends on the Operating System and uses network services (TCP/IP).

6.2. The GCCS Ambassador Segment

The C4I Ambassador is a generic term. Each Ambassador provides interfaces to the same common DII COE services but also contains interfaces to a unique set of Mission Applications within the host system. The Mission Application interfaces are the primary differences between Ambassadors.

The GCCS Ambassador, for example, is a tailored instance of the C4I Ambassador. It is a new GCCS segment that enfranchises a GCCS Workstation/LAN to act as an HLA federate. It provides the linkage between the RTI Segment and a number of specific DII COE and GCCS applications and databases.

The GCCS Ambassador provides a simulation data and transaction interface to the DII COE Infrastructure Services and Common Support Applications. This component of the interface is common for all Ambassadors in DII COE compliant C4I systems. It also provides the data and transaction interfaces to GCCS Mission Applications such as the GCCS COP via the Track Database Manager (TDBM).

Some of the prominent features of the C4I Ambassador include:

- Contains code defining C4I HLA data objects for use with the applicable Federation Object Model (FOM).
- Contains converters to match HLA data object interactions to C4I applications data structures or data objects.
- Determines internal C4I data routing to/from DII COE and Mission Applications.
- Contains HLA user interfaces to the federation and to the simulation.
- Provides control over the data update frequency and user perception of the data.

7. Federation Object Model Development

A Federation Object Model (FOM) defines the data formats and type of transactions allowed among federates. This is a series of defined data objects that are created to match the data exchange requirements for supported functionality between the federates. A Simulation Object Model (SOM) is the subset of the FOM that pertains to a particular federate.

In the development of C4I Federates, several different approaches were used to develop both the FOM and the software in the C4I Ambassador to support them.

7.1 Support for Legacy Simulations

For the case of legacy simulations with limited abilities to modify internal data formats, FOM data objects may be defined that directly or closely match the simulation's definition of objects and various types of interactions. The Joint Theater Level Simulation (JTLS) federation is such an example.

This approach, while the easiest to implement for the legacy simulation, requires extensive code in the C4I Ambassador unique to the particular simulation. While useful for a specific purpose, it does not scale well, and ties the FOM changes directly to hard code in the Ambassador segment.

7.2 GCCS Standard SOM

Utilizing experience in C4I federation developments, the NRL developers with the assistance of all participants in C4I federations to date have begun to develop the foundation of a "standard" GCCS Simulation Object Model (SOM – subset of the FOM pertinent to a specific federate). Within the GCCS SOM, objects are instantiated that mimic object-like activities occurring within the C4I Common Operational Picture (COP).

This makes the object data transfer to/from the simulation and C4I more like the actual real-world C4I events, and lends itself more easily to object attribute divesture (ownership transfer), a key component in C4I-Simulation COP synchronization.

For example, a C4I system owns the set of object attributes for the COP. COP objects could be created and passed to a simulation performing a (future) course of action analysis (COA). The ownership of a certain attribute may be handed off to the simulation so that it could update potential events as part of the COA, and send the resulting information back across to the C4I system.

At this juncture, the primary SOM objects implemented within the Ambassador are those associated with platform and unit tracks used within the COP. Some GCCS-to-simulation commands have been implemented to allow two-way interactions and provide a level of simulation control within the GCCS. The evolution of the GCCS SOM will continue and its makeup will depend upon the requirements and capabilities of the simulations with which the GCCS is federated.

Advanced interactions such as ownership divesture, and platform creation and destruction across the C4I-Simulation federation are more straightforward with this SOM. A final benefit; the supporting code in the COE C4I Ambassador can remain fairly stable from federation to federation when used with this SOM.

For these reasons, future simulation developments that are required to interoperate with DII COE compliant C4I systems should seriously consider incorporating the GCCS SOM as a baseline C4I interface requirement within their FOM.

7.3 Support for Special Purpose FOMs

Several special purpose FOMs have been supported to date by the NRL research as important within the COE C4I Ambassador:

- Real-time Platform Reference (RPR) FOM (Distributed Interactive Simulation based format)
- LATR FOM (Large Area Tracking Range data format)

These special purpose FOMs, representing stable formats, are supported in the COE C4I Ambassador to enable C4I participation in specific HLA federations including DIS capable simulations and instrumented training ranges.

The advantage of using one of the standardized FOM's when contemplating using a C4I HLA federation is that the C4I Ambassador and RTI COE segments will already support them with minimal software modification. As more uses are explored for the use of the HLA in C4I, the list of FOM's supported can grow through revisions to the COE software.

8. C4I Federation Applications

The technology reported herein has been used in a number of C4I/Simulation federations.

8.1 STOW 97 and 98

The initial C4I federation development was conducted at NRL for the STOW 97 and 98 exercises. [8] An early version of the RTI was installed within a PC, along with the rudiments of a C4I Ambassador, as middle-ware between the STOW simulations and the GCCS and GCCS-M networks.

Although only one-way transactions occurred in the STOW federations, C4I/Simulation interoperability was considered one of the major achievements of the STOW exercises. Additionally, the GCCS Archive and Reconstruction capabilities were used extensively during the exercises and for many of the post-exercises briefings.

The STOW 97 and 98 Exercises proved the feasibility of C4I federates and provided the insight for continuing C4I federation developments and the integration of HLA with the DII COE.

8.2 GCCS/JTLS/NATO Federation

The GCCS/JTLS/NATO federation was the first to achieve two-way transactions. The GCCS Ambassador permitted track information sent directly from the simulations, to be entered directly into the GCCS track database through an HLA RTI embedded within GCCS. GCCS generated maneuvering orders (paths of intended movement) that were sent to JTLS, which in turn modified the simulated track movements.

Multiple C4I federates and force perception controls were achieved in this federation. Various perceptions of the scenario created by the simulations were sent to multiple GCCS federates. For instance, one GCCS user would receive only a "Blue Force" perception of the scenario while another GCCS user would get only the "Red Force". This power to configure an HLA C4I Federate perception in a distributed simulation based exercise has great potential for operational exercise use and war gaming.

8.3 GCCS-M/NSS Federation

The Naval Simulation System (NSS) and the GCCS Ambassador utilize the GCCS FOM in this federation to allow two-way communication between GCCS and NSS. Various force movement orders are sent from GCCS to the NSS simulation. Simulated track information is then sent to GCCS in return.

Perception routing, facilitated by the RTI and the GCCS Ambassador, provides a simulation the capability to insert simulated data into selected C4I workstations (federates) on an operational LAN (i.e., can be operating in real world mode). Separate GCCS federates within this federation display different partial views of the scenario.

The simulated track data is inserted into the GCCS track database, via the RTI, and seen and processed only within specific designated GCCS workstation federates. This is accomplished without disrupting normal operations of other GCCS workstations on the C4I network that are not participating in the federation.

Furthermore, the RTI and GCCS Ambassador implementation allow the simulations to run in non-real-time simultaneously with normal operations running in real-time. No special network connectivity other than a standard C4I TCP/IP network is required. The RTI/GCCS Ambassador allows positive control of simulated data on a live, fully functioning operational C4I network.

This federation authenticated the concept of simulation-based training or planning to occur on operational C4I networks. Additionally, the concept of mixing real time C4I operations network activity with non-real time simulation based planning and decision support was explored and confirmed.

8.4 GCCS/Pegasus Federation

The GCCS Ambassador utilizes the Pegasus FOM for this federation. The simulations included in this federation were: EADSIM, Eagle, NSS, SLAMEM, and CMTAT. The GCCS COP provides the only graphical view of the Federation output.

The Pegasus federation runs with many objects at very high data rates. Previous experiences with non-real-time federations had not taxed the capability of the GCCS track database to handle track updates submitted at high frequency and non-selectively. This federation allowed stress testing of GCCS maximum input rates (number of platforms x update rates x non-real-time ratio) up to 1000 times real-time.

Lessons learned in the Pegasus federation resulted in the initiation of the development of a data-sampling module for the C4I Ambassador, to control maximum update rates for various types of tracks and buffer the C4I track database. This function can be used with federations that run much faster than real time and contain a large number of entity updates.

8.5 Embedded C4I Federations

NRL is investigating the concept of a C4I federation wholly contained within a GCCS system. This capability figures prominently in the ongoing collaborative mission planning development within the C4I Embedded Simulation Program. [5]

An embedded C4I federation capability can support Mission Applications requiring the use of simulated data and distributed across an operational LAN or WAN. One such target application being addressed is a distributed Mission Planning Application.

The RTI establishes a GCCS "virtual sub-LAN" as an additional means of distributing planning data and application controls between operators on separate GCCS workstations. The RTI also allows existing C4I applications such as: scenario generators, archive and replay functions, map viewers, and mission analysis tools to be integrated into the Mission Planning Application in a distributed, collaborative fashion among user's workstations.

8.6 GCCS-NSS Federation for Global 01

The GCCS-NSS Federation is undergoing additional modifications that will allow it to be used to support Course of Analysis (COA) analysis during the Navy Wargame "Global 01" during August 2001.

A restriction experienced in previous Global exercises was that data from the GCCS COP had to be manually entered into NSS prior to initiating COA runs during the exercise. This method was often laborious, time consuming, and prone to mistakes. An operational result of this manual process was often the NSS could not be used in decision support due to the time required to initialize.

Real world COP data on current unit locations and status will be sent from the GCCS system via the RTI and Ambassador to the NSS. There it will be reconciled with more detailed data residing within the NSS database and initialize COA simulation runs quickly.

This interface lends itself to being easy to update (as data in the COP changes) and less prone to errors. It will also provide a means of gaining an understanding of how C4I data could be used to initialize COA runs using the same HLA and DII COE components that already are supporting other C4I-Simulation applications.

9. Conclusion

Robust C4I-Simulation interoperability can be accomplished within HLA federations that contain C4I federates. C4I federates are established by embedding the RTI within the DII COE along with additional interoperability software that manages the simulated data within the C4I system.

C4I federate and federation developments to date have demonstrated the following:

- The RTI can be embedded into the DII COE as a COE service, and utilized when paired with C4I Applications.
- The GCCS Ambassador can link GCCS COP functionality to a simulation via the RTI in a bi-directional mode.
- GCCS can function as a fully enfranchised HLA federate.
- C4I federates have object ownership potential.
- A number of C4I federates can simultaneously exist on a single GCCS LAN, for varied purposes.
- C4I federates can coexist with real-world operations occurring within the C4I system.
- An HLA federation can be used internal to a C4I system for communication among elements of a distributed C4I Application.
- The GCCS/RTI integration allows data base transfers as well as formatted message communications. C4I systems can initiate indirect control (e.g., NSS, JTLS commands from GCCS applications).
- C4I systems can provide scenario initialization and updates from the C4I COP.

The HLA RTI will be submitted as a segment for test and release within the DII COE. The C4I Ambassadors will be released as Mission Applications within the individual C4I systems.

References

[1] Joint Technical Architecture Version 3.1, Modeling and Simulation Domain Annex, 31 March 2000 http://www-jta.itsi.disa.mil/

[2] High Level Architecture, Defense Modeling and Simulation Office (DMSO) web site http://www.dmso.mil/index.php?page=64

[3] Timian, D.H., Hieb, M.R., Lacetera, J., Tolk, A., Wertman, C., and Brandt, K.: “Report Out of the C4I Study Group”, Paper 00F-SIW-005, 2000 Spring Simulation Interoperability Workshop, 2000.

[4] Layman, G. E. & Giovanis G.: “Embedded Training for the Global Command and Control System” Interservice/Industry Training, Simulation & Education (I/ITSEC) Conference, November 1999; Orlando, Florida

[5] Layman, G.E., Weatherly, J. and Daly, J.: “C4I Embedded Simulations,” Paper 01S-SIW-082, 2001 Spring Simulation Interoperability Workshop, 2000.

[6] Hieb, M.R., Cosby, M., Griggs, L., McKenzie, F., Tiernan, T., and Zeswitz, S.: “MRCI: Transcending Barriers between Live Systems and Simulations,” Paper 97S-SIW-197, 1997 Spring Simulation Interoperability Workshop, 1997.

[7] Defense Information Infrastructure Common Operating Environment Home Page http://diicoe.disa.mil/coe/

[8] Layman, G.E., Conover, J., Kunkel, P., and Robins, D.: "JMCIS/GCCS Interoperability with External Simulations," Paper 97S-SIW-132, 1997 Simulation Interoperability Workshop, Spring 1997; Orlando, Florida

Author Biographies

Dr. Gene Layman is a Research Engineer at the Naval Research Laboratory and directs a Command and Control Development Team that has been active in GCCS and GCCS-Maritime developments since their conceptions. Earlier he managed the Radar Branch and the Guidance Branch at the Naval Surface Weapons Center. Before that, Dr. Layman was a Program Manager in the Missile and Ordinance Division at Texas Instruments, Inc. and a Circuit Designer at The Boeing Company on the Minuteman Program. He has led or participated in the design of more than two-dozen military systems with more than half becoming fully operational.

John J. Daly, is a Research Engineer with ITT AES at the Naval Research Laboratory, and works in the Command and Control Development Team with responsibilities for C4I/Simulation Interoperability, the development of an Electronic Warfare Mission Application for the GCCS-Maritime, and Embedded Training Technologies for C4I. A retired Navy Officer, John previously worked on the staff of the Director, Defense Information Systems Agency, as well as in the Modeling and Simulation Directorate on C4I and simulation interoperability.

Zach Furness is a Principal Engineer with the MITRE Corporation and is the Deputy Chief within the Warfighter Division of the Defense Modeling and Simulation Office (DMSO). Mr. Furness oversees all C4I-Sim projects that are sponsored by DMSO and serves as chair of the DII COE Modeling and Simulation Technical Working Group (M&S TWG). Prior to supporting DMSO, Mr. Furness was the MITRE Project Lead for the Joint Training Confederation (JTC) and has over 10 years experience in simulation and C4ISR in support of military training, analysis, and acquisition.

Jennie A. Womble is a Senior Software Engineer with ITT AES at the Naval Research Laboratory, and works in the Command and Control Development Team in GCCS and GCCS-Maritime software developments. She has over 10 years experience in developing leading edge technology for the Navy and is the principle software developer of the GCCS Ambassador