C2 Interoperability: Simulation, Architecture and Information Security

J. A. Hamilton, Jr., Ph.D.
107 Dunstan Hall
Auburn University
Auburn, AL 36849
hamilton@eng.auburn.edu

CAPT John Melear, USN
Code 055
Space & Naval Warfare Systems Command
4301 Pacific Highway, San Diego, CA 92110-3127
melear@spawar.navy.mil

Mr. George Endicott
Code 055A
endicotg@spawar.navy.mil

Abstract

The intensity and frequency of joint and combined operations, including operations other than war (OOTW) as well as the accelerating technological advances in command and control have highlighted C2 interoperability issues. Communications is increasingly software-driven. The makes the role of software architecture increasingly important in designing interoperable communications systems. Simulation has an important role to play in defining and resolving interoperability issues stemming from requirements, policies and procedures. Interoperability is such a challenge, that the security implications of enhanced interoperability are often not considered. This paper will suggest that the CCRP consider publishing a volume on C2 interoperability codifying the body of knowledge developed by the JC2I2G offices.

1. Background

The commanders of the service C2 acquisition centers, Communications and Electronics Command, Fort Monmouth (CECOM), Space and Naval Warfare Systems Command, San Diego (SPAWAR), Electronic Systems Center, Hanscom, AFB (ESC), formed the Joint Command and Control Integration Interoperability Group (JC2I2G). The JC2I2G exists to promote joint interoperability and change processes and structures by initiating "bottom up" change to implement Joint C2 integration and interoperability, and by supporting the unified commands in resolving interoperability issues of service-specific systems. Recognizing the pivotal role the US Joint Forces Command (USJFCOM) as the Joint Force Integrator, the Director, J6 of USJFCOM serves as a principal member of the JC2I2G.

The JC2I2G proposed and the Under Secretary of Defense for Acquisition and Technology, Dr. Jacques S. Gansler, approved the establishment of the CINC Interoperability Program Offices (CIPO) at each C2 acquisition center and the establishment of the Joint Forces Program Office (JFPO). After a start-up period that was focused on "proving the concept," the CIPOs achieved sufficient short-term successes that they can now focus on long-term, non-trivial interoperability issues.

The CIPOs now play a major role between the originators of joint requirements and the designers of service C2 systems. The primary purpose of the Joint Forces Program Office is the horizontal integration of the CIPO efforts across the Unified Commands in

direct support of US Joint Forces Command. As JFCOM's roles and missions evolve with respect to interoperability, so has JFCOM's interaction

In space of just a little more than two years, the JC2I2G organizations have gained significant insight into interoperability issues and solutions through experimentation, prototyping and results-oriented problem solving.

2. The interoperability domain is software driven

Interoperability in Command and Control applications is software-driven. This includes software-driven communication protocol stacks, operating systems and data element standards and security modules to give but an incomplete listing. For this reason, we believe it is reasonable to deal with interoperability by using an engineering lifecycle model.

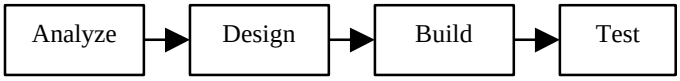


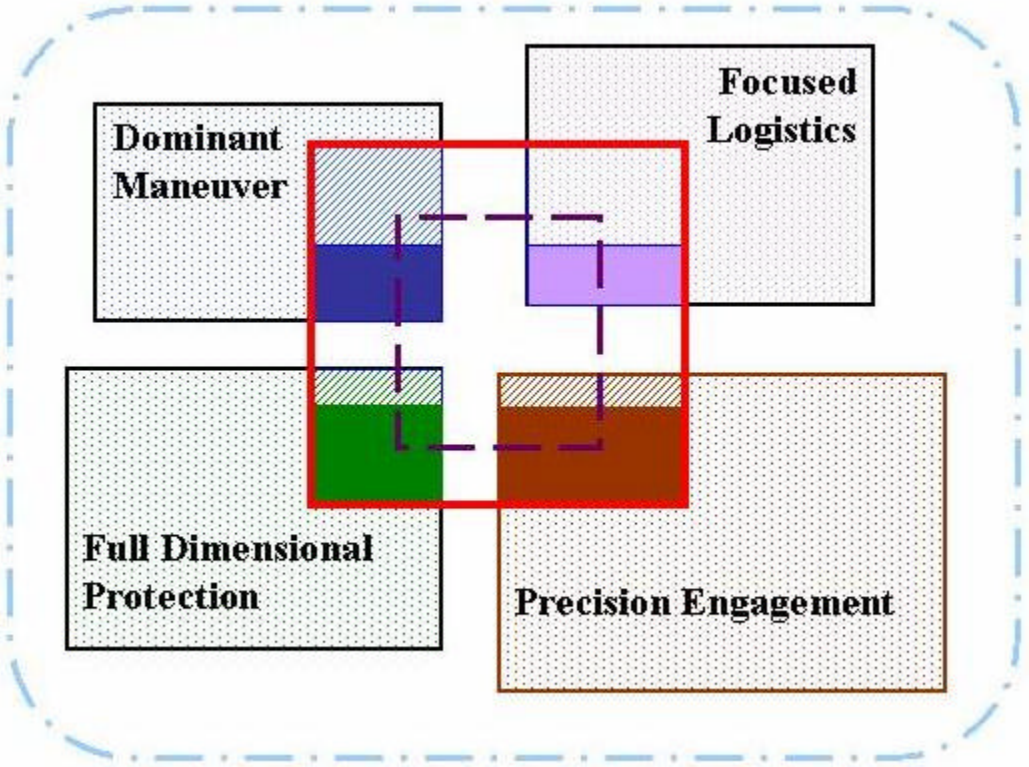
Figure 1. Engineer Thought Process.

In order to use the requirements -> design -> implementation -> test -> maintain software engineering model, it is important to first understand the domain to which this model will be applied (Figure 1).

Figure 2. Interoperability Domain.

This domain was defined during a USD (AT&L) directed study of DoD interoperability tools that continue to proliferate [Rosen and Parenti 2001]. A key insight is the realization that there are two dichotomies that affect this domain. The first dichotomy occurs between service requirements and joint command needs. The second dichotomy occurs between the combat arms/warfare communities/rated communities and the acquisition community as shown in Figure 2 above.

Any engineering approach begins with requirements definition. Interoperability requirements have to stretch across the domains shown in Figure 2. Simulation-based acquisition holds unique potential for interoperability requirements definition. However, successful application of SBA requires a program manager to negotiate his/her way across a disjoint domain as outlined previously. The successful development of a data model as shown in Figure 3, for the Global Information Grid requirements is an example of successful interoperability requirements definition [Hamilton, Murtagh and Deal 1999].



Key

Solid Thick Line: Joint Information Exchange Requirements (JIERs) necessary for interoperability. "MUST Share" Subset.

Thin Solid Section: "Planning" Information within each functional area

Thick Striped Section: "Survival" Information within each functional area

Large Dashed Line: Subset of information which can feasibly be shared between the new system and legacy system(s). "CAN Share" Subset.

Dash/Dot Line: Theoretical Boundary for information which might be shared

Figure 3. Global Information Grid Data Model [Hamilton, Murtagh and Deal 1999].

In Figure 3, the information from each of the four JV2010 functional areas is shown in rectangular boxes of different sizes. Several subsets of information are also identified; these subsets will be discussed in more detail later in this narrative. Note that the diagram is not "drawn to scale." For example, it is not our intent to imply that Precision Engagement will require more information than the other areas; we are just trying to illustrate that different amounts of information may be required for each area. This same caveat applies to all data subsets represented on the diagram.

3. Simulation and interoperability

Simulation, in its current state of the art, cannot substitute for interoperability testing. The fidelity requirements are simply too great to be cost effective for a useful test of communications interoperability. Resolution in simulation is a question of detail and is closely related to fidelity [Hamilton, Nash and Pooch 1996]. Fidelity is a measure of how closely the simulation approximates the real world [Bailey and Kemple 1992]. We can bound our notion of fidelity by defining perfect fidelity as a simulation that is indistinguishable from reality. A generalized means for achieving perfect fidelity in non-trivial simulations is still in the realm of science fiction.

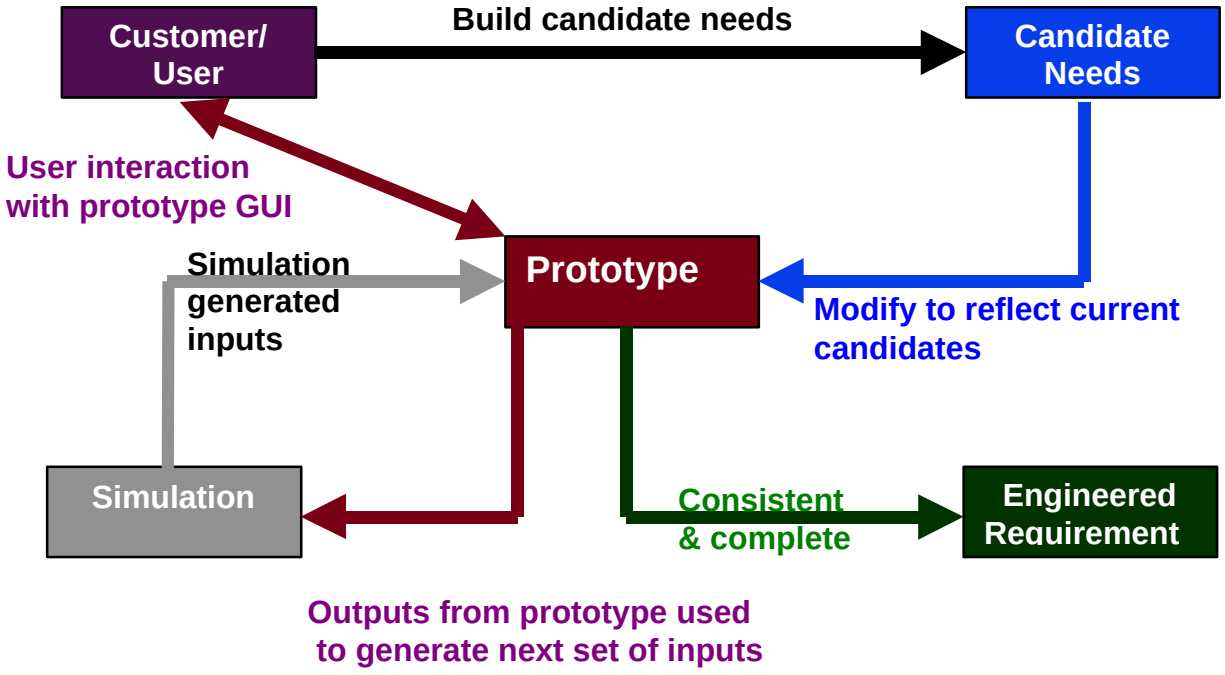


Figure 4. Simulation to support rapid prototyping.

Simulation can be effectively used for requirements definition. As shown in Figure 4, simulation can drive the prototyping to refine requirements. A prospective user is presented with the prototype system with inputs generated by a simulation. User reactions to the prototype are then cycled through the simulation to present new inputs to

the user. This in turn is used to refine requirements with the end goal of being a well-defined requirement.

The Navy's Space and Naval Warfare System Command, Office of the Chief Engineer has been experimenting in this area using a German developed "executable architecture" tool called Bonaparte. They have been successful in simulating a U.S. DoD defined Operational Architecture, and illustrating the impacts of model or parameter changes within the model. They have focused on the impact changes have on the user defined "Information Exchange Requirements."

A notional network simulation is shown below in figure 5. Figure 5 shows the high level design of computer network suitable for driving a prototyping effort.. For network simulation, significant events may be individual bits when studying physical layer phenomena; packets for most performance models; or messages for higher order traffic studies.

In the domain of computer networks a second is an eternity. A great deal can happen between one second and the next. It is not unrealistic to expect that modeling thousands of messages may require hundreds of thousands of programming instructions in the simulation. Once the physical network has been modeled and implemented in software, then analysis can be conducted by varying the configuration and/or the workload as shown in figure 5.

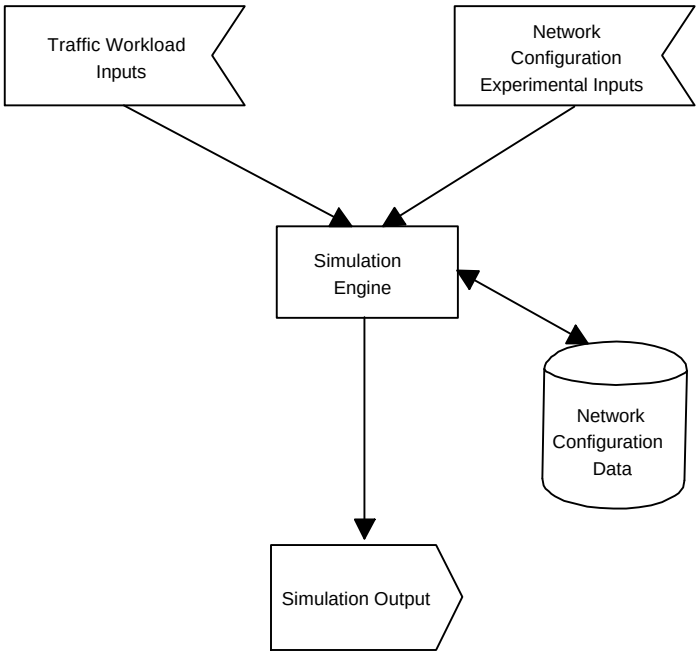


Figure 5. Computer network simulation model.

4. Software architecture

From requirements, the next step is design. Architectural methods can be applied to high-level design [Deal, Hamilton and Caudle 97]. Diverse hardware-based communications systems require an overall software architecture in order to interoperate. As noted in IEEE Standard 12207.0-1996 *Software Lifecycle Processes*, software architecture describes the top-level structure of the over-arching system and describes the software components. Specifically, developers adhering to the standard are required to development and document a top-level design for the interfaces external to the software item and between the software components of the software item. This is an essential first step in achieving interoperability between any two systems.

Software architecture is an emerging topic. [Hofmeister, Nord and Soni 2000] claim that software architecture provides a design plan – a blueprint – of a system, and it is an abstraction to help manage the complexity of a system. That blueprint must be managed however. So-called standards-based approaches are fine, but in fact, systems compliant to the same standard still may not interoperate. [Libicki 1995] observes that the first purpose of a standard is to achieve interoperability.

Unfortunately, standards do not guarantee interoperability. As [Bass, Clements and Kazman 1998] noted that dozens of systems deployed to Bosnia do not interoperate despite being designed to adhere to the Technical Architecture for Information Management (TAFIM). Bass et al., quote Rear Admiral John Gauss, then commander of the Defense Information Systems Agency's Joint Interoperability Engineering Office as saying "We have built a bunch of state-of-the-art, open systems, TAFIM-compliant stovepipes."

Software architecture is built on standards, but provides the high-level design to implement interoperable systems. It has been argued that software architecture is in fact a sub-architecture underlying the system architecture view as defined in the C4ISR Architecture Framework documents. For a detailed discussion of sub-architectures and their relationship to the architecture views in the C4ISR Architecture Framework documents, see [Hamilton 98].

5. Interoperability and information security

There are multiple ill-defined relationships between interoperability and network security. It is clear the computer network defense measures can present challenges to interoperability in terms of national policy, physical system implementation and trusted system relationships. It is unclear whether greater interoperability between national assets makes them more vulnerable to computer network attack. There are significant technical issues associated with communication system interoperability. In military communications, significant non-technical issues relating to national security policy and release authority also come into play.

Once coalition networks are established, the vulnerability of information systems may increase. Internal propagation of a worm with the characteristics of say "Nimbda" or "Code Red" can generate internal broadcast storms behind the network firewalls. This question has profound implications for homeland defense. Requirements play a key role here since interoperability can lead to increased end-point vulnerabilities across the National Information Infrastructure.

As Admiral Dennis Blair, USN, stated while serving as Commander-in-Chief, US Pacific Command: "There are also GCCS incompatibilities in combined operations; for example, GCCS Joint and GCCS-Korea. These two systems share some common operational picture data, but do not share information via files, e-mail, and other web service tools. Obstacles to combined interoperability lie in information release restrictions. Our allies understandably restrict release of their classified information. Likewise, we want to control release of U.S. classified information. To achieve effective combined interoperability, we must develop much more capable security procedures and sophisticated tools to allow information exchange while protecting our national and allied data [Blair 2001]."

Admiral Blair went on to observe, "Yes, we can improve information assurance in the theater; however, to do so requires a heavy investment in people and additional hardware. The payback is not always as easily recognizable as with the production of new airplanes, ships, or tanks. You cannot touch and feel information protection, but a loss of critical or time sensitive information or a denial of service can be far more detrimental to national security than a single weapon system [Blair 2001]."

An emerging area of complexity is DoD support to the Homeland Security infrastructure. Local and State jurisdictions have different, and sometimes unique, information requirements, which can lead to confusing and conflicting information exchanges. Furthermore, communications systems tend not to interoperate across jurisdictions, and in some cases, across functional areas (police, fire, medical, etc.). As an example, during Joint Task Force – Olympics, the Commander's Control Center had representatives from forty-two jurisdictions to coordinate information and operations. This did not include the Intelligence and Releasability cell.

Adding to the problem set is the issue of information assurance. In DoD parlance, this means the protection and coherent distribution of sensitive or classified information. Security clearances exist in the Federal world, and some larger police departments have Federal clearances for their intelligence units, but it is by no means consistently applied across local jurisdictions.

The DoD response is to participate in many task forces, designed to address the above referenced issues and develop solutions. DoD is also proposing several experiments to address the interoperability concerns, with the goal of developing one or more solutions sets.

Conclusions

Homeland defense requires interoperability across many existing systems across many different agencies. From a practical standpoint, this is a much harder problem then simply designing a system against a well-defined standard (i.e. TCP/IP); or even designing a system to interoperate within an existing system of systems (i.e. GCCS). Programmatically, DoD is not well organized to support retrofitting interoperability capability into existing systems.

From an engineering viewpoint, it is often the case that there exist some parts of one system for which there is no equivalent part in the other system. Tactical Data Links are an excellent example of this problem. Technology advances that render the proposed solution obsolete before it is ever fielded can hamper a technical solution. Worse, the two systems, managed by two different program offices, may be on different upgrade paths with different implementation schedules.

Homeland defense must increase the priority given to C2 interoperability within the Defense Department. If DoD C2 systems cannot interoperate among themselves, then interoperation with non-DoD agencies is that much harder. DoD programmatic requirements limit some of the flexibility program managers have in adjusting to many changing requirements from many different agencies; rapidly advancing technology. Combine this with perennially uncertain funding and program managers face tremendous challenges in just fielding a system, let alone an interoperable system that uses state-of-the-art technology.

The CIPO's and the JFPO have been migrating research and activities from the pursuit of "fixing legacy applications and environments" to a "born joint" approach. This is a requirements based approach to interoperability. The majority of so-called "legacy" interoperability problems will never be solved until replacement systems are brought on-board. For this reason, interoperability clearly lies in the future, in the requirements that are being developed today. To achieve interoperability, those requirements will need to define a communications software architecture; be validated by communications simulation techniques and will need to address information security issues. Interoperability is an increasingly important aspect of command and control and worthy of specific emphasis by the CCRP.

References

[Bailey and Kemple 1992] Bailey, M. P., Kemple, W. G., "The Scientific Method of Choosing Model Fidelity," *Proceedings of the 1992 Winter Simulation Conference*, Society for Computer Simulation, 1992, pp 791 - 797.

[Bass, Clements and Kazman 1998] Bass, L., Clements, P., Kazman, R., *Software Architecture in Practice,* Addison-Wesley, Reading Mass., 1998, p 392.

[Blair 2001] Blair, Admiral D.C., testimony before the Senate Armed Services Committee 27 March 2001.

[Deal, Hamilton and Caudle 97] Deal, J.C., Hamilton, J.A., Jr., Caudle, J., "Unknown Lands and Uncharted Waters, The Army Enterprise Architecture," *3rd International Symposium on Command and Control Research and Technology*, June 17 - 20, 1997, National Defense University, Fort McNair, Washington, D.C., pp 426 - 449.

[Hamilton 1998] Hamilton, J.A., Jr., "Achieving HLA Compliance Via the Joint Technical Architecture – Army," Summer Computer Simulation Conference, July 19 - 22, 1998, Reno, Nev., pp 509 - 513.

[Hamilton, Murtagh and Deal 1999] Hamilton, J.A., Jr., Murtagh, J.L., Deal, J.C., "A Basis for Joint Interoperability," 1999 Command & Control Research & Technology Symposium, US Naval War College, 29 June – 1 July

[Hamilton, Nash and Pooch 1996] Hamilton, J.A., Jr., Nash, D.A., Pooch, U.W., *Distributed Simulation*, CRC Press, Boca Raton, Fla., 1997, p. 215.

[Hofmeister, Nord and Soni 2000] Hofmeister, C., Nord, R., Soni, D., *Applied Software Architecture*, Addison-Wesley, Reading Mass., 2000, p. 4.

[Libicki 1995] Libicki, M.C., *Standards, the Rough Road to the Common Byte*, Center for Advanced Concepts and Technology, National Defense University, Fort Lesley J. McNair, DC, 1995, p 2.

[Rosen and Parenti 2001] Rosen, J. D. and Parenti, J. L., "Aligning Interoperability Tools Within the DoD Battlespace," 2001 DoD Software Technology Conference, Salt Lake City, Utah, 29 April – 3 May 2001.

Authors

John A. "Drew" Hamilton, Jr., Ph.D., is an associate professor of computer science and software engineering at Auburn University. He has a B.A. in Journalism from Texas Tech University, an M.S. in Systems Management from the University of Southern California, an M.S. in Computer Science from Vanderbilt University and a Ph.D. in Computer Science from Texas A&M University. Prior to his retirement from the US Army, he served as the first Director of the Joint Forces Program Office and on the Staff and Faculty of the United States Military Academy. CRC Press publishes his book, *Distributed Simulation*, written with LTC David A. Nash and Dr. U. W. Pooch. email:hamilton@eng.auburn.edu

Captain John Melear, US Navy is the Director of the Space and Naval Warfare Systems Command's Commander-in-Chief Interoperability Program Office (CIPO). Previously he served as the Commanding Officer of the Commander Naval Surface Force, U.S. Pacific Fleet headquarters Naval Reserve unit and the Navy Material Management Support Office Naval Reserve unit. He also served as Executive Officer of two Naval Reserve units. Captain Melear has a B.S. from the United States Naval Academy, and is pursuing a Masters in Software Engineering at the Naval Postgraduate School. He holds two

major naval warfare qualifications as a Surface Warfare Officer and as a Naval Aviator (Radar Intercept Officer). email:melear@spawar.navy.mil

George Endicott is the Deputy Director of the Space and Naval Warfare Systems Command's Commander-in-Chief Interoperability Program Office (CIPO). Previously he served as the Deputy Director of the SPAWAR Architecture Directorate in the Office of the Chief Engineer (code 051). Mr. Endicott has served in a variety of high-level assignments in both OSD and SHAPE. He is a recognized expert in C41 architecture and data interoperability. email:endicotg@spawar.navy.mil