APPLYING A THREE-TIERED MODELING APPROACH TO THE ANALYSIS OF NEXT GENERATION C4ISR SYSTEMS

Roger Smith, Jennifer Iding, Seth Lytle
SPARTA Inc.
Orlando, Florida
Roger.Smith, Jennifer.Iding, Seth.Lytle@sparta.com

KEYWORDS

Conceptual Model, C4ISR, UML Sequence Diagrams, Extend Modeling

ABSTRACT

The requirements documents for new weapons systems refer heavily to their ability to collect and share information through an extensively networked Command, Control, Communications, Computers, Intelligence, Surveillance, & Reconnaissance (C4ISR) system. Analyzing the capabilities of this system requires defining it in a concise and understandable manner. We have begun this process by creating a conceptual model of the components within the C4ISR system. This model exists at a higher level than specific engineering representations of hardware, software, and humans. The conceptual model provided the structure necessary to create Unified Modeling Language (UML) sequence diagrams of the military mission processes. UML sequence diagrams tie together the resources and relationships from the conceptual model with the specific activities that are prescribed for the system. These diagrams then provided a blueprint from which to create a discrete event model of the activities that take place when vehicles conduct a specific mission. This allowed us to understand the contention for resources that occurs within one mission and across activities of multiple missions.

This paper describes the derivation of a concise conceptual model, the creation of UML sequence diagrams, and their implementation in the Extend discrete event simulation (DES) software package. This sequence of models allowed us to create early performance models of the C4ISR system and evaluate the realism of stated performance requirements.

NEXT GENERATION C4ISR SYSTEM

The acquisition of a major new weapon system presents a tremendous challenge for the engineers who must understand the requirements for the system and capture that in various forms for analysis, requirements, design, and production. Military systems are often described through the decomposition of physical objects and through linear process flow descriptions. In order to analyze these systems, engineers must create models that define the core components of the system, their relationships, and the constraints that exist within them. One early phase approach to this problem is to begin with a conceptual model that captures relationships, constraints, and sequences in an implementation-independent form.

The next generation of military systems will all be linked into information networks that provide near-real-time access to everything that is known about the battlefield (Figure 1). This knowledge is expected to provide a significant combat advantage to friendly forces, allowing them to identify and attack enemy threats while minimizing both fratricide and collateral damage.

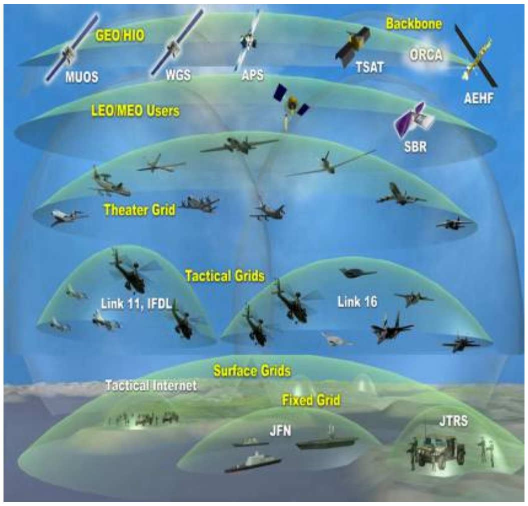

Figure 1. Next-Generation C4ISR systems provide globally shared access to battlefield information.

The envisioned information networks are flexible enough to maintain data exchange while platforms are moving and adaptable enough to continue functioning when nodes on the network are destroyed. This type of dynamic performance requires much more advanced communications equipment, protocols, and end-node systems than are used on currently fielded systems. Many companies are engaged in research and engineering projects targeted at solving this problem. One part of solving this problem is modeling the performance of the system to identify data throughput issues and to design self-healing networks. However, our experience is that the information available about a future C4ISR network is limited to statements about the required performance and mission descriptions illustrating the system in action. During the early phases of a program when modeling of its capabilities can provide guidance in designing the system, there is no information about the potential performance, configuration, or architecture of the system from which to construct a model.

Our approach to this problem has been to create three progressively detailed types of models to build up our understanding of the system from its mission descriptions and performance requirements to an understanding of the performance of the entire C4ISR system.

CONCEPTUAL MODELING OF C4ISR SYSTEMS

A conceptual model (CM) is an implementation independent representation of a system that is intentionally designed to generalize details of the system and identify large components, their functionality, and relationships (Valle 1999). Such a model is a useful first step in wrestling with a complex system about which a great deal is known, but little of that knowledge has been organized.

When a system is extremely complex with many unique capabilities and component relationships, this complexity can deter all attempts to understand its capabilities. This is an excellent opportunity to create a conceptual model that focuses on general capabilities rather than specific details. It creates a manageable representation of the system that can be studied and expanded as more detail about the system is understood and organized (Lacey 2001).

The CM identifies the large classes of objects or components that make up the system or those with which external interfaces exist. It also includes the relationships between these components and the general behaviors that exist within each. Ideally, a CM of a complex system is much easier to understand. These conceptual models are also used as categorization schemes because they allow people to classify components, behaviors, and relationships into a few very manageable structures (Borah 2002).

In our case, the CM was a first step in understanding the C4ISR system and it allowed us to take the next step in representing the behaviors of the system and tying those behaviors to the hardware and software resources within the system. This is the beginning of a process for identifying the necessary resource levels for the system and for evaluating high-level architectures of the system.

Unlike a requirements-based approach to defining a system, the CM is top-down rather than bottom-up. In many cases, the bottom-up approach captures many thousands or even millions of requirements and configurations for the system. This information is usually unorganized and presents a completely unmanageable problem. Beginning with thousands or millions of data points, it is impossible to construct a system to achieve all of those requirements. The conceptual model approaches the problem from the top-down, identifying the major categories of functions and relationships for the system and presenting a problem that is manageable from the beginning. This general model then serves as a framework within which to add specific requirements and to create unique variations.

The CM for the C4ISR system that we are interested in is shown in Figure 2. It contains only seven major components, 26 sub-components, and connections that focus on the relationships between the components and the network.

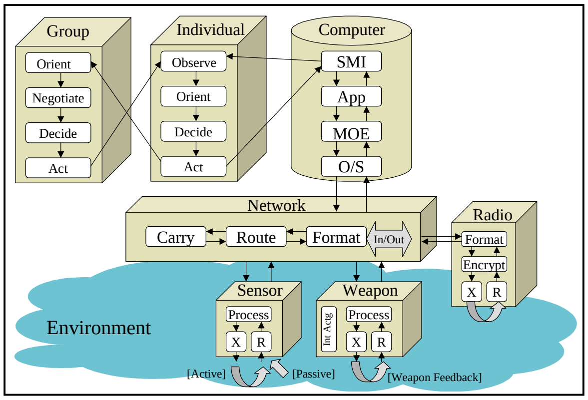

Figure 2. Conceptual Model of the C4ISR network.

Identified Components and Relationships

The *network* component represents the conduit for wired communications. This includes the network wires, routers, hubs, switches, and even the in-computer network interface cards. The *radio* represents the conduit for all wireless communications, both voice and data. This is responsible for the formatting, encrypting, transmission, and reception of information. The *sensor* component transmits energy (active) and receives energy (active and passive) that is processed before being communicated through the network and radio. The *weapon* component transmits/sends munitions and may receive data about weapon performance. It also tracks its internal resources. The *computer* component represents the functions of the operating system, the military operating environment, specific applications, and the military user interface.

Humans are not part of the "as-constructed" C4ISR system, but they are an important part of the "as-operated" system. Therefore, *individuals* and *groups* are included in this model to capture the impacts that they have on performance. The individual human component performs the traditional observe, orient, decide, and act (OODA) loop of activities. The group component makes decisions and provides decisions to be carried out by a specific human.

The *environment* is external to the C4ISR system, but can have significant impacts on the system. Therefore, the environment will be included in some analytical tools when its effects are pertinent to the problem being studied.

MISSION PROCESS DEFINITIONS

The functionality of the C4ISR system is defined in formal Mission Processes (MPs) that are derived from operational concept documents for the project. In particular, MPs illustrate the complex interactions of architecturally significant processes at the System of Systems level that require hardware/ software development to support the concept. Several of these MPs are summarized in Table 1.

Table 1. Summaries of Core C4ISR Mission Processes.

MP1 Battle Command uses automated tools to enable the Commanders, Staffs and Higher Headquarters to apply leadership and decision making to available information supplied by the system network. The Battle Command process allows leaders to shape and sustain their decision actions by seamlessly synchronizing elements within and between moving or stationary echelons regardless of location within the battlespace area.
MP2 Sensor Management is the planning, launching, monitoring and active retasking of available battlespace sensors to complete the ISR collection plan and feed raw sensor data into the system network for fusion. The intent of this process is to provide seamless overlapping coverage of the operations area with sensors and shooters.
MP3 Networked Fires is fully integrated from theater to platform, allowing the military units to rapidly establish, change and terminate connection links between battlespace sensors, launch systems and Joint systems to achieve a wide variety of lethal and non-lethal effects. The fires and effects coordination accelerates target processing and distribution and will make real-time engagement decisions using automated systems capable of leader intervention.
MP4 Maintain Operational Picture displays timely fused data on terrain, weather, civilian, enemy, and friendly forces tailored to each echelon and user-specific needs. This common operational picture facilitates collaborative planning and visualization of the battlespace enabling situational awareness.
MP5 Maintain Networks plans, creates and supports the information structure for the end-to-end movement of communications and data through the network, focusing on the Quality of Service (QoS) of the system. Network Management (NM) ensures the effective and efficient operation of the network and Information Dissemination Management (IDM) provides the correct information to the correct system or person at the correct time in the correct format.
MP10 Multi-mode Training is not a stand-alone system, but rather an embedded capability in all systems to manage, conduct, and assess collective and individual training for the military focused on the Mission Essential Task Lists (METL).
MP12 Perform Combat Identification is the effective and real-time identification of all entities within the battlespace area in order for each user to gain situational awareness through their operational picture. The system uses integrated methods and technologies among Army, Joint and Coalition forces to achieve combat ID.
MP18 Robotic Operations uses technologies to maneuver, support and sustain Unmanned Aerial Vehicles (UAV), Robotic Ground Vehicles and remote control of manned systems. This process performs many of the “dangerous” missions to replace the involvement of the soldier, thus achieving mass effects without mass deployment of individuals.
MP20 Intelligence Operations process encompasses the fusing of information and intelligence to provide relevant, accurate and timely intelligence to the network by leveraging various resources. This process enables the commander to see first, understand first, and act first.
MP21 Information Assurance ensures the integrity, availability, identity, authentication, and confidentially of friendly, Joint, Multinational, and Coalition information and systems. The process has steps to prevent hacking and network degradation, and a self-healing capability to ensure the continued flow of critical information.
MP24 Information Management is compromised of military doctrinal information requirements for information flow into and out of the Global Information Grid (GIG), including storage, discovery and mediation services. Central to this process is the interaction and fusion of the Battle Command System applications, IDM, and the Network Centric Information Environment.

The complete family of MPs are comprised of a series of steps that, when completed in a certain order, contribute to accomplishing a mission. Each of the MPs sequentially executes specific steps that are limited by threshold requirements and time values. All of the MPs feed off each other through a distributed database to push and pull the information needed to complete each step within each Mission Process.

The MPs link together in intricate patterns that define the activities conducted by a military entity/unit of action to achieve a desired effect. The desired effects can range from launching an airborne sensor platform to inducing psychological distress in the enemy. Using UML sequence diagrams to link the MPs and their individual steps illustrates our top-down modeling approach (Figure 3). As time flows downward, the UML shows the progression of a mission and the essential MPs at any one time.

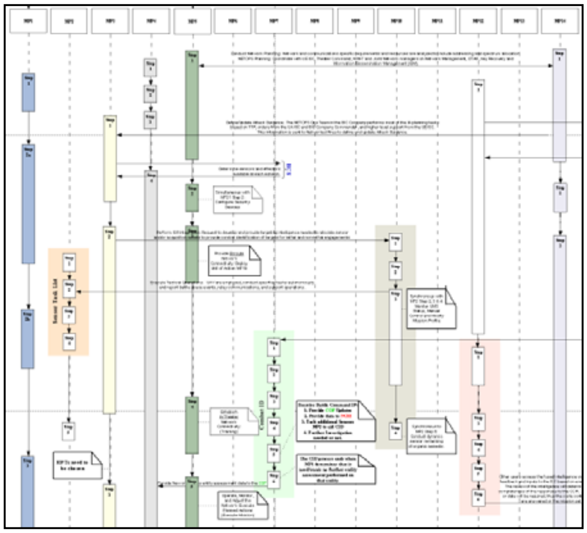

Figure 3. UML sequence diagrams allow the modeler to view simultaneous MPs, triggered events, and data flow between processes as a particular mission progresses through time.

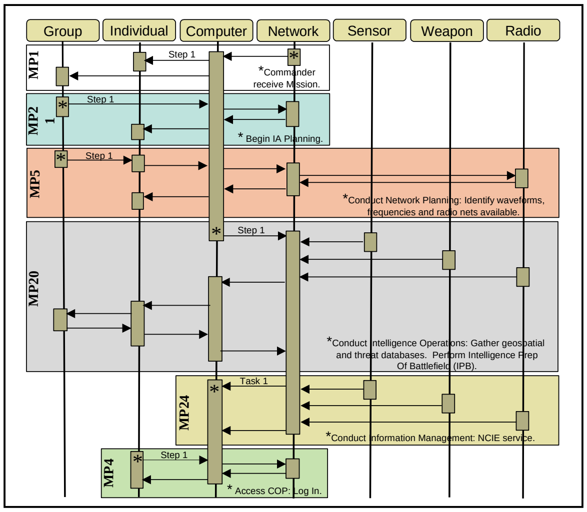

Figure 4. UML sequence diagram of mission processes operating across C4ISR resource classes.

A snapshot of time from the UML gives the modeler a view of simultaneous processes, triggered MP events, common linkages between particular MPs, and possible bottlenecking or network overload on a System of System scale. This model is a first step toward representing the functionality of the conceptual model and showing how those resources contribute to and limit the execution of a specific mission. Figure 4 illustrates how we might marry together the process-focused model in Figure 3 to the resource-focused conceptual model in Figure 2.

MODELING PROCESS FOR COMPLEX SYSTEMS

When modeling large and complex systems, modularity and repeatable patterns within the model design become key components for flexibility and extensibility. In some ways, this design perspective is not far from the hierarchical design in modern computer programming where inheritance drives the data flow.

In our C4ISR system, data is passed from one node to the next, and they in turn, may trigger other events. For instance, the computer system on one of the vehicles might receive a message from another node on the battlefield and need to display the information to the commander on the vehicle, and also disseminate the information to others in the area. To handle this procedure, the computer will make a copy of the message and route it to the appropriate locations. When modeling this transaction, one can duplicate the message and keep the uniqueness of the object through its attributes in tack. The attributes of the item passed through the discrete event simulator provide the information needed to tell the computer where the object came from and where it should be routed next.

The current version of the model creates a new mission every 20 minutes through an event generator, at which point the mission parameters are created. The constant arrival rate of new missions allows us to control the level of parallel work that we are studying during each run of the model. However, each of the sub-processes that the items must pass through have their own associated distributions with variability for the amount of time that each sub-process will take. The path of an item (message, report, track, etc.) is determined by data coded into the item itself. It is not a hard-coded part of the simulation software. All of the MPs are designed to decode the attributes and to use this information to both route them and process them at specific nodes. This significantly reduces the amount of information that must be stored on each item that passes through the DES. The real logic and discernment of resource pairing comes from the MPs themselves. The logic in the Router section of the model directs the items to their respective mission paths, which is determined by the attributes. This logic is primarily derived from the MP-based UML and flow chart diagrams described in Figures 2 and 3. There are many possible paths for the items, but each items' attributes carry the necessary information to direct its path through the logic tree in the simulation. Resources are assigned to each item as it passes through any of the MPs. These resources are required by the item upon arrival into each MP step, and are then released back into the aggregated pool after the process has completed.

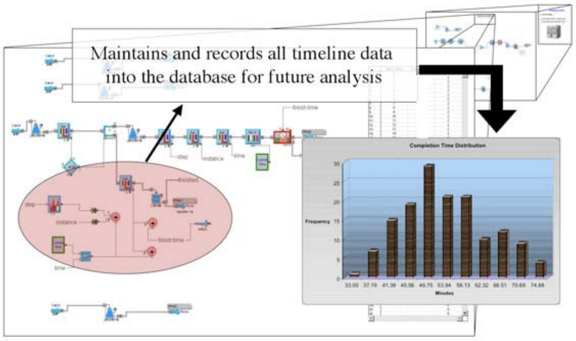

Figure 5. A hierarchical view of how information is routed and recorded to both maintain a simulation scenario as well as provide data for useful time-study analysis.

The primary purpose of the simulation is to represent independent missions that coexist simultaneously. Many scenarios can potentially be run at the same time, or may be run independent from one another. The time, resource levels, and progress of each scenario are recorded in a database for future analysis (Figure 5). With this data, we can deduce where bottlenecks exist and where resources are being under-utilized. Both cases are characterized as an inappropriate matching of valuable resources to mission needs. These areas indicate where the system may need to be redesigned or where mission responsibilities may need to be reallocated. By adjusting the system and the military functions systematically, we can identify the right ratio of resources required to perform a mission.

CONCLUSION

This paper describes our efforts to understand a very complex C4ISR system based on requirements statements and descriptions of its mission application. We have applied three different modeling approaches to the problem in an attempt to manage the complexity of the system and to provide a framework that we can use to study the system in more detail as the design matures. Conceptual modeling, UML Sequence Diagrams, and DES tools are all tools familiar to operations analysts, systems engineers, and software developers. Combining these to progressively represent and understand a complex system is a unique approach to systems analysis. We are attempting to represent the problem from the top-down rather than bottom-up because the level of complexity of the system is much higher than exists in current C4ISR systems and threatens to overwhelm the ability of a small team to understand its operations. The method of combining these models is still emerging and we are still learning how to apply it to all of the requirements of the program.

REFERENCES

Booch, G.; Rumbaugh, J.; & Jacobson, I. 1999. *The Unified Modeling Language User Guide*. Reading, MA: Addison Wesley Publishing.

Borah, J. Spring 2002. Conceptual Modeling – The Missing Link of Simulation Development. *Proceedings of the Spring 2002 Simulation Interoperability Workshop*. Orlando, FL. (02S-SIW-074)

Duck, A.; Timian, D.; Auth, M.; Dunbar, R.; and Karr, C. Fall 2004. Army Future Force Intelligence, Surveillance, and Reconnaissance (ISR) System-of-System (SoS) Conceptual Model. *Proceedings of the Fall 2004 Simulation Interoperability Workshop*. Orlando, FL. (04F-SIW-003)

Lacy, L. et al. 2001. Developing a Consensus Perspective on Conceptual Models for Simulation Systems. *Proceedings of the Spring 2001 Simulation Interoperability Workshop*. Orlando, FL. (01S-SIW-074)

Shayo, C and Olfman, L. 1998. The Role of Conceptual Models in Formal Software Training. *Proceedings of the 1998 ACM SIGCPR Conference on Computer personal Research*. Boston, MA.

Valle, T. March 8, 1999. Battlespace Abstract Model (BAM). Internal Document to the JSIMS Program.

Weiner, J. & Gutierrez, P. Fall 2003. Conceptual Modeling of a Legacy Constructive Simulation: A Use Case. *Proceedings of the Fall 2003 Simulation Interoperability Workshop*. Orlando, FL. (03F-SIW-064)

Xue-hui, W; Lei, Z.; & Ke-di, H. Fall 2004. Improving Simulation Conceptual Model. *Proceedings of the Fall 2004 Simulation Interoperability Workshop*. Orlando, FL. (04F-SIW-036)

BIOGRAPHIES

ROGER SMITH is Chief Engineer with SPARTA Inc. working on next-generation simulation architectures and standards. He is the creator and instructor for a series of simulation courses that have educated hundreds of simulation professionals. He serves on advisory boards for Sandia National Laboratories, University of Florida, and University of Central Florida, and on the editorial boards of the *ACM Transactions on Modeling and Computer Simulation* and the *IRI Research Technology Management*. He holds degrees in Mathematics, Statistics, Computer Science, and Management.

JENNIFER IDING is a Computer Engineer with SPARTA Inc. She is exploring the relationships within multiple defined mission processes and the connections to C4ISR resources within new weapon systems. She has a B.S. in Physics from The Ohio State University and is pursuing an M.S. in Physics at the University of Central Florida.

SETH LYTLE is developing discrete event simulations that analyze the performance of new C4ISR systems. These are being used to determine whether the system can achieve its operational requirements. He has a B.S. in Industrial and Systems Engineering from Georgia Tech and an M.S. in Industrial Engineering from the University of Central Florida.