ARCHITECTURES IN THE ANALYSIS OF REQUIREMENTS AND DESIGN OF SIMULATION SYSTEMS

Georgi KIROV and Valentin STOYANOV

Abstract: The paper presents authors' experience in developing and using architectural descriptions, following the US DoD Architecture Framework, to the analysis and the design of an Air Situation Simulator (ASS). The publication describes how the use of C4ISR architectures in the definition of requirements and analysis of complex systems provides for a conventional description of the domain and for detailed study of the processes in the investigated system. The steps and stages in the description of operational, systems, and technical views of the simulation system are presented. The requirements for the system's software development are defined on that basis.

Keywords: C4ISR Structural Approach, Air Situation Simulator, System Design, System Engineering, DoD Architecture Framework, DODAF.

The main purpose of this article is to present the experience from practical application of the architecture approach, as defined in the DoD Architecture Framework and implementation guidance,[1] to the purpose of analyzing requirements and designing an Air Situation Simulator (ASS). This simulator can be presented as a complex system that consists of great number of objects. The simulation is in real time and the processes have some random features.

The main problem in the ASS realization is associated with an operational environment that is rapidly changing, uncertain, and with abundant information. There is a considerable number of threats, conflicts are multi-polar, alliances often shift. Therefore, just maintaining situational awareness in the ASS is a significant challenge for the architecture specialists. Critical functions in the ASS are information fusion and management on different levels, communication, planning and execution monitoring.[2] The addition of requirements to simulate rapid deployment, joint operations, and use of information infrastructure built quickly in hostile environment, casts doubts on the effectiveness of current technologies.

A number of scientific, technical and technological challenges need to be addressed in the search for solutions of these problems. In the last decade, two system approaches have been implemented to provide rules, guidance, and products for developing and presenting architectural descriptions that ensure a common denominator for understanding, comparing, and integrating architectures. These approaches are the Object Oriented approach (OO) and the C4ISR structural approach.[3]

Analysis and Design Approaches

This section briefly presents two alternatives to systems engineering: OO and C4ISR. Although there is no official standard for Object Oriented Methodologies, the Unified Modeling Language (UML) becomes leading and most widespread OO language. UML is modeling language that provides a set of diagram types, their component entities, and the metastructure that relates these entities to each other. UML has an internally consistent and rigorously logical descriptive mechanism, which can be used for architecture description. This approach is appropriate for the stage of the software realization.[4]

The Command, Control, Communications, Computers, Intelligence, Surveillance, and Reconnaissance (C4ISR) Architecture Framework [5] is used for development and presentation of the ASS architecture. C4ISR is known as a structural approach that is based on two types of diagrams: Data Flow Diagram (DFD) and Entity Relationship Diagram (ERD). The C4ISR Architecture Framework provides guidance on describing architectures.

Definition of the Simulation Environment Requirement (C4ISR Architecture Framework)

An architecture description is a representation, in a current or future point in time, of a defined “domain” in terms of its component parts, what those parts do, how the parts relate to each other, and the rules and constraints under which the parts function.

There are three major C4ISR perspectives, i.e., views, that logically combine to describe an architecture. These three architecture views are operational, systems, and technical views.[6] Each of the three architecture views has implications on which architecture characteristics are to be considered and displayed, though there is often some degree of redundancy in displaying certain characteristics from one view to another.

C4ISR provides architecture products that constitute the minimal set of products required to develop architectures that can be commonly understood and integrated within and across organizational boundaries, as well as between elements from different nations.

Operational Architecture View

According to the C4ISR Architecture Framework, the operational architecture view is a description of the tasks and activities, operational elements, and information flows required to accomplish or support a military operation. It contains graphical and textual descriptions of the operational elements, assigned tasks and activities, and information flows required to support the warfighter. It defines the types of information exchanged, the frequency of exchange, which tasks and activities are supported by the information exchanges, and the nature of information exchanges in detail sufficient to ascertain specific interoperability requirements.

System Architecture View

The systems architecture view is a description, including graphics, of systems and interconnections providing for, or supporting, warfighting functions. It identifies which required systems support the operational view requirements. It translates the required degree of interoperability into a set of needed system capabilities and compares current implementations with needed capabilities. It is a description of systems and interconnections providing for, or supporting, operational functions.

Technical Architecture View

The technical architecture view is the minimal set of rules governing the arrangement, interaction, and interdependence of system parts or elements, whose purpose is to ensure that a conformant system satisfies a specified set of requirements.[7]

The technical architecture view provides the technical systems implementation guidelines upon which engineering specifications are based, common building blocks are established, and product lines are developed. The technical architecture view includes a collection of the technical standards, conventions, rules and criteria organized into profiles that govern system services, interfaces, and relationships for particular systems architecture views and that relate to particular operational views.

Operational Architecture View

ASS High-Level Operational Concept Graphic **(OV-1)**

The High-level Operational Concept Graphic of the Air Situation Simulator (ASS) is the most general of the architecture-description products and the most flexible in format. Its main utility is as facilitator of human communication. It is intended for presentation to high-level decision makers. The ASS Operational Concept Graphic describes visually missions, high-level operations, organizations, and geographical distribution of assets.[8]

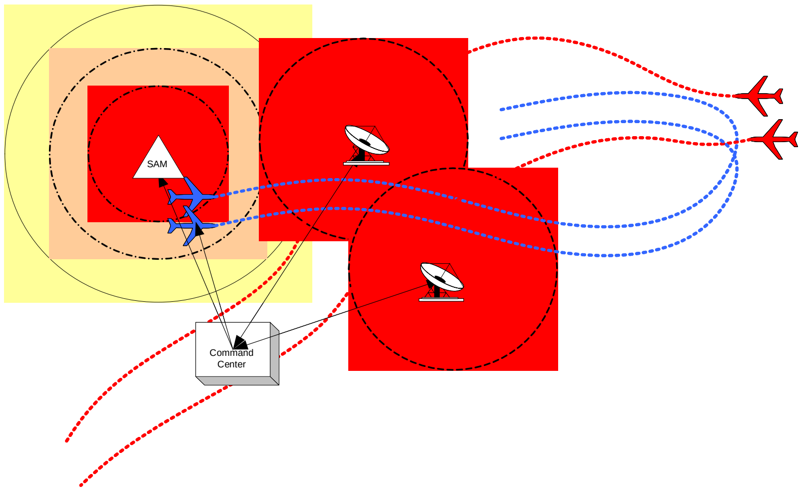

Figure 1: High-Level Operational Concept Graphic.

The Anti-Air Warfare (AAW) system protects restricted area against intruders. When intruder (threat) is detected, an aircraft is sent to intercept it. According to different level of hostility, the engagement may take different forms:

- During war time (war), the interceptor shoots the intruder without warning;
- During peace time, the interceptor attempts to contact the intruder and shoots at it only if the latter does not reply.

Figure 1 shows ASS icons that can be tailored and used to represent various classes of players in the architecture – enemy and own aircraft, that represent air operations or mission, SAM elements – radars, command posts, control systems. The lines connecting the icons show simple connectivity and what information is exchanged.

***Operational Node Connectivity Diagram* (OV-2)**

Main features of this product are the operational nodes and elements, the needlines among them, and the characteristics of the information exchanged. Each information exchange in the ASS is represented by an arrow (indicating the direction of information flow), which is annotated to describe the characteristics of the data or information, e.g., its substantive content, media (voice, imagery, text and message format, etc.), volume requirements, security or classification level, timeliness, and requirements for information system interoperability (Figure 2). Information-exchange char-

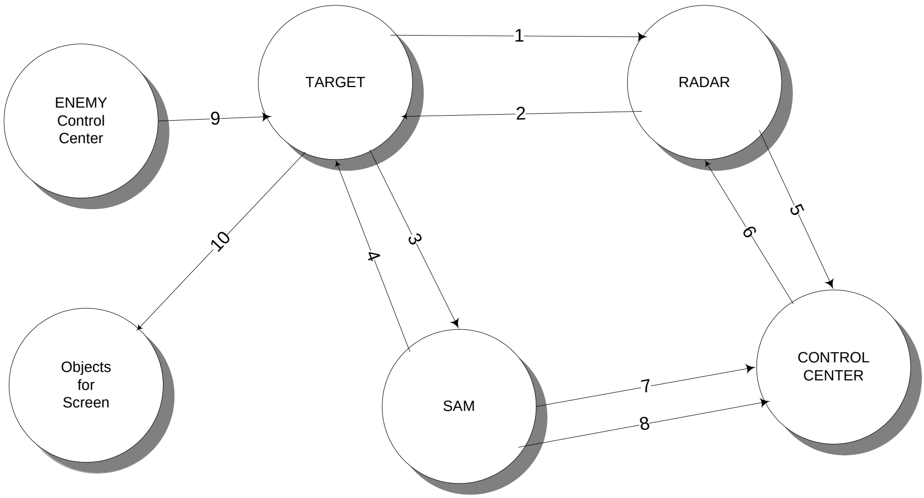

Figure 2: Operational Node Connectivity Diagram (OV-2).

acteristics can be shown selectively on the diagram or, more comprehensively, in a matrix format.

Information Exchange Matrix (OV-3)

Using the defined activities as a basis, Information Exchange Requirements (IERs) express the relationship across the three basic entities of the ASS operational architecture (activities, operational elements, and information flow) with a focus on the specific aspects of the information flow. IERs identify who exchanges what information with whom, why the information is necessary, and in what manner. IERs identify the elements of warfighter information used in support of a particular activity and between any two activities. The node of the producing operational element and the node of the consuming operational element are identified. Relevant attributes of the exchange are noted. The nature of the Operational IER Description lends itself to being described as a matrix.

Organizational Chart (OV-4)

The Command Relationships Chart illustrates the relationships among organizations or resources in the ASS architecture. These relationships are important to show in an operational view of the architecture because they illustrate fundamental roles and management relationships. In the ASS case the experts define three main functions: Sense, Command, and Act (Figure 3).

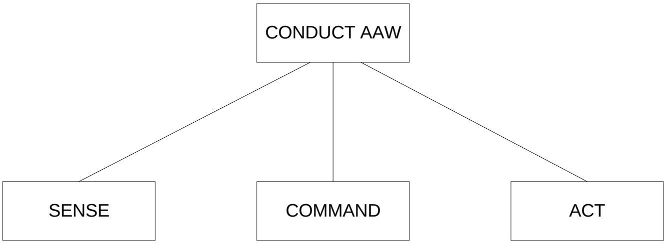

Figure 3: Hierarchy of Air Defense Activities.

Activity Model (OV-5)

The Activity Model of the simulator describes the applicable activities associated with the architecture, the data and information exchanged between activities, and the data and information exchanged with other activities that are outside the scope of the model.[9] The ASS Activity Model captures the activities performed in a business process or mission and their ICOMs (Inputs, Controls, Outputs, and Mechanisms). Mechanisms are the resources that are involved in the performance of an activity. In addition, the Activity Model identifies the mission domain in the ASS model and the viewpoint reflected in the model. Activity definitions and business flows should be provided in additional text, as needed. Annotations to the model identify the nodes where the activities take place or the costs associated with performing each activity.

The activity (process) model represents at each level of decomposition the data exchanged between functions. IDEF0 and Data Flow Diagram (DFD) are the tools used for description of the process model (Figures 4 and 5).

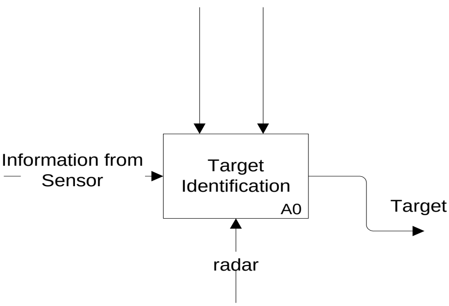

Figure 4: Target Identification.

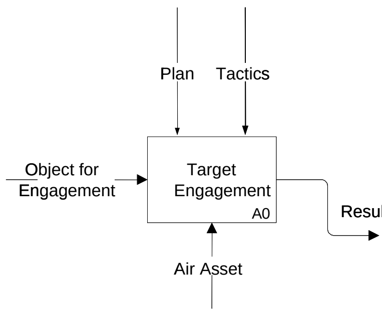

Figure 5: Target Engagement.

Operational Activity Sequence and Timing Descriptions (OV-6a, 6b, and 6c)

The main purpose of the following figures is to illustrate IDEF0 and the relationships between the different types of models: process, data, and rule.

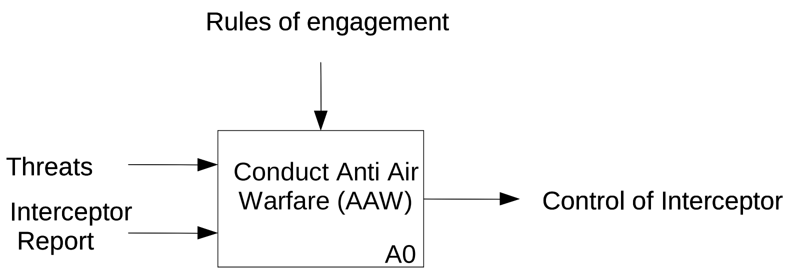

Figure 6: IDEF0 Process Model.

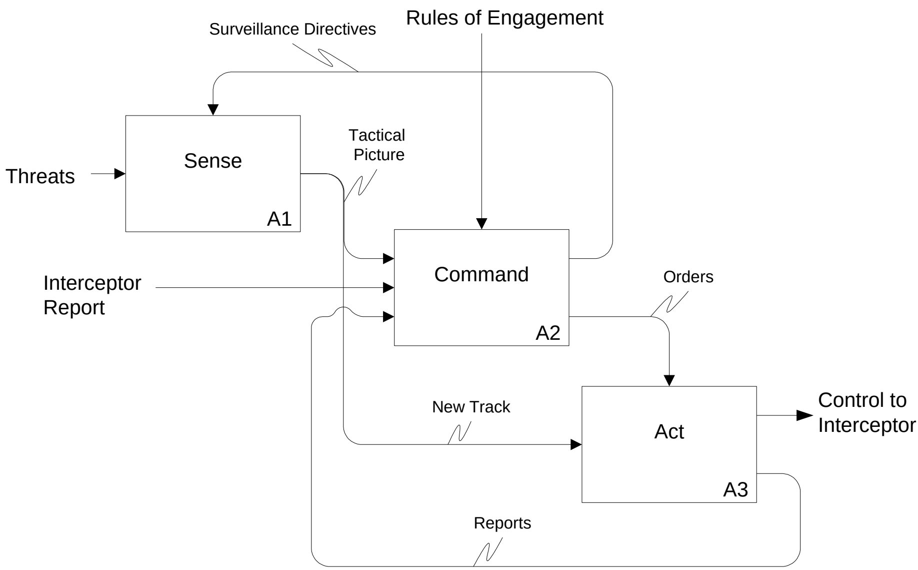

Figure 7: IDEF0 Process Model in Details.

The data and information exchanged between activities, and the data and information exchanged with other activities that are outside the scope of the model are shown in Figure 8 and Figure 9. They are based on DFD.

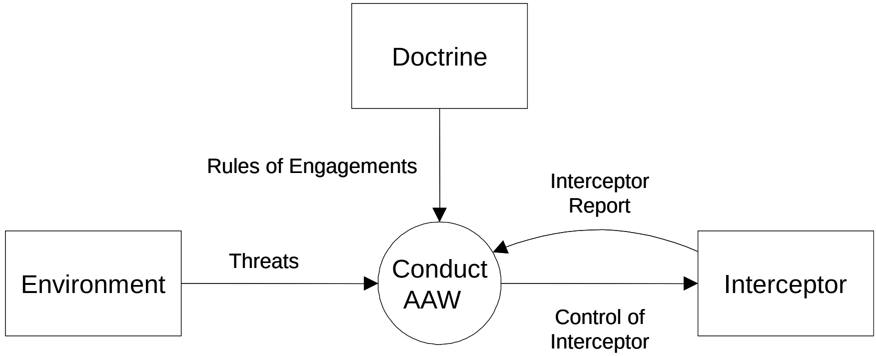

Figure 8: AAW DFD Process Model.

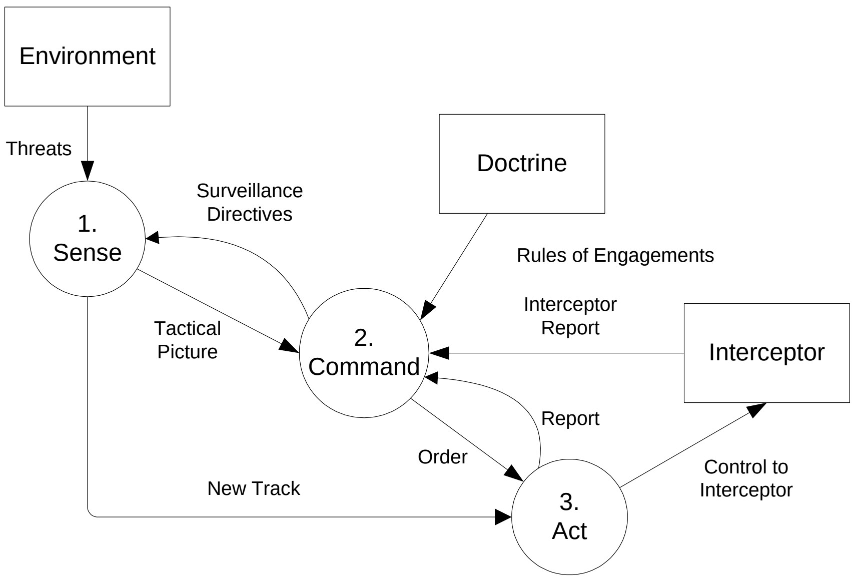

Figure 9: AAW DFD Process Model.

Many of the critical characteristics of the architecture are only discovered when its dynamic behavior is defined and described. That relates to the timing and sequencing of events that capture operational behavior of a business process. Three types of models are needed to refine and extend the architecture's operational view to ade-

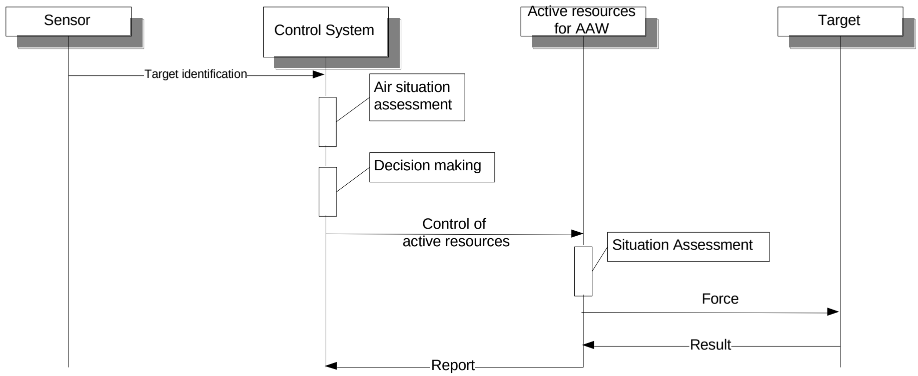

Figure 10: ASS Operational Event Trace Diagram.

quately describe the dynamic behavior and performance characteristics of the architecture. These three models are:

- Operational Rules Model (OV-6a);
- Operational State Transition Description (OV-6b);
- Operational Event/Trace Description (OV-6c) (Figure 10).

The Operational Rules Model is part of the architecture's operational view and extends the capture of business requirements and concept-of-operations information introduced by the Logical Data Model.

The Operational State Transition Description and the Operational Event/Trace Description describe business-process responses to sequences of events. Events may also be referred to as inputs, transactions or triggers. When an event occurs, the action to be taken may be subject to a rule or set of rules as described in the Operational Rules Model.

The Operational Event Trace Diagram is sometimes called a sequence diagram. It shows interactions in terms of messages, or information transfers, between operational nodes arranged in time sequence. The Operational Event/Trace Description can be used by itself or in conjunction with an Operational State Transition Description to describe dynamic behavior of processes.

Data Model

The data model specifies the different types of data handled by the system and their relationships. The most used tools in this stage are IDEF1 and Entity Relationship Diagrams (ERD). Figure 11 presents eight entities that correspond to the ICOM of the IDEF0 model.

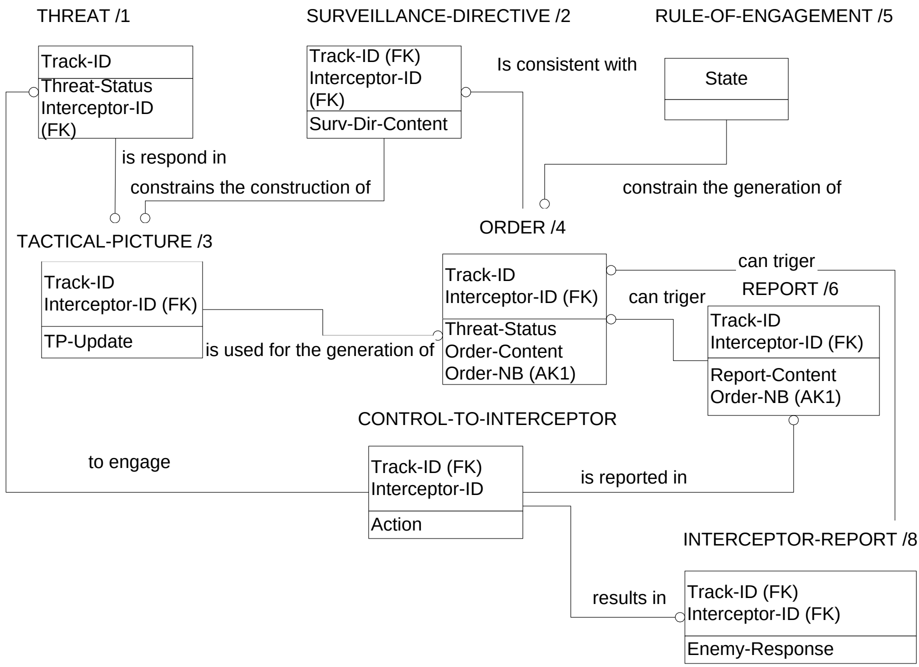

Figure 11: AAW IDEF1x Data Model.

Conclusions

This summary of the use of the C4ISR Architecture Framework in the design an Air Situation Simulator (ASS) shows how the architect uses tools and techniques of structural analysis to produce a coherent set of products. Among the main advantages of this approach are:

- Use of common methodology for architecture design;
- Improved system analysis;
- Agility and shorter decision making cycles;
- Optimization of the information exchange;
- Readily available support for oversight of the implementation.

The next steps of our study and the design of ASS will be to create system and technical views of the architecture and software simulation.

Notes:

[1] *C4ISR Architecture Framework*, *Version 2.0* (U.S. Department of Defense, C4ISR Architecture Working Group, December 1997).

[2] *LISI 97 Reference and Capabilities Maturity Model*, Draft (The MITRE Corporation, September 1997).

[3] C4ISR stands for 'Command, Control, Communications, Computers, Intelligence, Surveillance, and Reconnaissance' (systems).

[4] *Unified Modeling Language Notation Guide*, *Version 1.0* (Santa Clara: Rational Software Corporation, 1997).

[5] *Integrated Architecture Panel Final Report* (C4ISR Integration Task Force, 3 July 1996).

[6] Robert E. Lee, *Strawman Joint Operational Architecture*, Briefing to the C4ISR Architecture Working Group (22 June 1999).

[7] A. Rausch, *GIG Database Development Update* (Washington, DC: National Security Research, Department of Defense, 19 April 2002).

[8] *C4ISR Architecture Framework*.

[9] Lee, *Strawman Joint Operational Architecture*.

GEORGI KIROV is born in 1973. He holds M.Sc degree (1996, Computer Technologies) from the Technical University of Sofia and Ph.D. degree from the Institute of Computer and Communication Systems at the Bulgarian Academy of Sciences (2002, Soft Computing models in computer networks). Dr. Kirov is Research Fellow at the Department of Knowledge-Based Control Systems of the Institute of Control and System Research at the Bulgarian Academy of Sciences. He conducts research on the use of intelligent technologies in telecommunications networks, system research and programming languages. *E-Mail:* kirov@icsr.bas.bg.

VALENTIN STOYANOV is born in 1956. He holds M.Sc degree from the Military University of Shoumen, Bulgaria. Mr. Stoyanov is a research fellow at the Department of Knowledge Based Control Systems of the Institute of Control and System Research at the Bulgarian Academy of Sciences. *E-Mail:* v_sto@yahoo.com.