# Modeling & Simulation accelerates complex system design, test, and verification process

Uma S. Jha, Ph.D.
Space & Airborne Systems, Raytheon Co
El Segundo, California

Erik Chowdhury, Ph.D.
Space & Airborne Systems, Raytheon Co
El Segundo, California

*Abstract*— **Modeling and Simulation (M&S) has been a key contributor to the validation of key system concepts, trade-off analysis, and dimensioning of the system performance before it could be realized in cost effective and timely manner. Leading system developers harness M&S capabilities for competitive advantage and operational/business efficiency by leveraging advances in VLSI technologies (processing horse power, memory, storage, high throughput buses), graphics, display, and networking technologies and its utility spans not only across concept development, and trade-off analysis but also in system design, test and evaluation, failure analysis, and, to some extent, identifying production issues.**

***Keywords—Modeling, Simulation, Trade-off analysis, CONOPS, Design & development, Performance evaluation, Testing & Verification, Prototyping, Failure Analysis, Algorithm Development***

## I. INTRODUCTION

The main objective of the M&S is to make informed decisions based on use cases, systems analysis, and trade-off studies to minimize developmental glitches and to facilitate prudent programmatic decisions making. The novel goal of M&S is to support the entire product life cycle from initial concept development to design, integration, test, validation, verification, cannibalization, and follow-on products. The purpose, type, fidelity and complexity of analyses performed across the program life cycle of concept development through product realization requires an scalable M&S model, which can add value at each stages of the development process and in addition provide a collaborative and open framework to take advantage of state-of-the-art tools, technology, and process improvements.

In modeling and simulation realm all functional and behavior models of a system attributes and characteristics are represented in digital domain and can be manipulated digitally. Thus physical and behavioral models based system can be tested in a computer generated virtual, digital world in the same manner as the real systems are tested in the real world. An additional benefit of this approach is the ability to integrate multiple simulations into high-fidelity system of systems test beds. Additionally, the advancements in Very Large Scale Integration (VLSI) technology and software tools to support high- fidelity target and environment simulations as well as immersive visualization environments can be leveraged to facilitate wide variety of concept of operations (CONOPS) development and trade-off analysis.

The simulation fidelity typically grows as the system progresses towards the maturity and end-to-end system performances and interactions are validated. While initial studies often rely on analytical tools, real-time simulations are necessary for assessing complex interactions (such as the scheduling and resource loading of multimode airborne radar). These simulations will rely on lower-fidelity representations of the system since the detailed design may not be fully understood or completely developed. In addition the simulation fidelity (both all-digital and Hardware in the Loop (HWIL)) is only as good as the simulated threats and operating environments being modeled.

Due to advancements in VLSI technology, the ability to model key aspects of a system design before it is built, and to simulate that design's performance over time, has made it possible to significantly improve system performance while alleviating the need to build expensive hardware for test purposes. The enhanced offerings and capabilities of computing, networking, graphics, and storage technologies should be leveraged in architecting a simulation, which can link multiple simulations and operate them collectively towards achieving a common goal. The genesis of a M&S framework at large system integration house is to achieve physically distributed simulation nodes to interact in an all-inclusive simulation with human interfaces. This effort has allowed for effective utilization of the corporate wide company resources providing a very tolerant and highly available system with adequate redundancy at each level of simulation for stake holders. Today, we apply a more integrated approach that is based on rapidly configurable simulations that can grow in fidelity as a system design matures from its initial trade studies through SIL/HWIL testing. These simulation and modeling effort has reduced the risk of design and test errors, alleviated cost overruns, and improved the quality of analysis in the development of complex systems.

## II. BACKGROUND

During early 80's in a complex system development environment, the constituent stake holders brought their own favorite tools and processes, while systematic but due to the lack of the processing horse power and integrated software development environment did not allow for scalability and seamless integration. Sub-systems developed in silo did not integrate well and scale adequately to manage evolving



978-1-5090-0790-5/16/$31.00 ©2016 IEEE

requirements. It lacked the capabilities to take advantage of the technological advancements to churn out the best possible solution. Requirements in general came in form of a worksheet; systems team developed their algorithm and performance compliance using mathematical tools (e.g. Matlab, Mathematica …); software team used their own code development processes and tools, and hardware team integrated, tested and sold off to the ultimate customer. The whole process was not sufficiently integrated resulting in loss of traceability and accountability through the development cycle.

The cost overrun and project delay variation depended on the development stage, when the issue surfaced and whether a patch could be stitched together or not. The Figure-1 depicts the cost overrun impact [1] as a function of the project stage and its severity as it treks towards the completion.

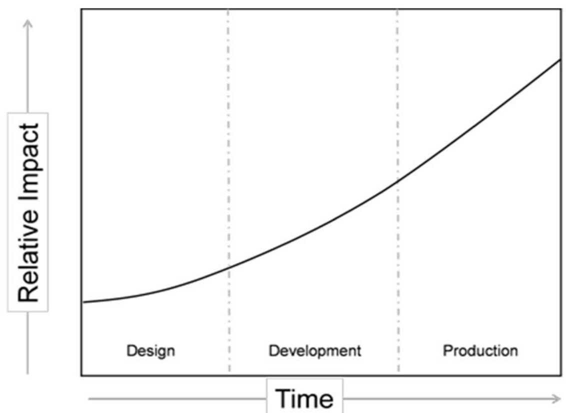


Figure - 1: Impact of issues on cost & timeline

To overcome this, various processes and tools were considered and experimented to manage traceability from requirements to product design, testing, and sell- off phases utilizing DOORS, Ada programming language including water fall, V model, Spiral model as well as other development processes. Although these methodologies showed improvements in execution of the projects from the ground up but did not solve the issues of integrating the heterogeneous complex system prevalent in most of the Department of Defense (DoD) contracts in which some subsystems lasted across the multiple generations of system of systems platforms while others keep evolving to counter new threats and improved capabilities in the battlefield.

## III. SYSTEMS ENGINEERING FRAMEWORK

To improve the efficiency of product development engineering community have devised Waterfall, V, Spiral, and Agile methodologies, which entails the systems engineering life cycle and management plan through development phases. Although the basic principle of these management plans have served the community well but has not evolved sufficiently to take advantage of the modern tools and processes perfected in other disciplines. The strong emphasis on the requirements definition at the very early stage of development prohibits any major changes even if the cost, features and/or performance could be improved substantially utilizing recent advancements in the underlying processes or technological innovations. The cost and schedule impediments makes it difficult to revisit and explore needed enhancements or improvements. Many times hardware prototypes are built just to explore the concepts and capabilities thus incurring huge investments and very little flexibility. The next few sections describes the key features of these methodologies with its pros and cons.

## IV. WATERFALL DEVELOPMENT PROCESS

The process flows in serial manner– the next task can only be started if the current task has been completed. At the end of each task, a review takes place to determine if the task has met all the requirements to continue further. It's shown pictorially in Figure-2.

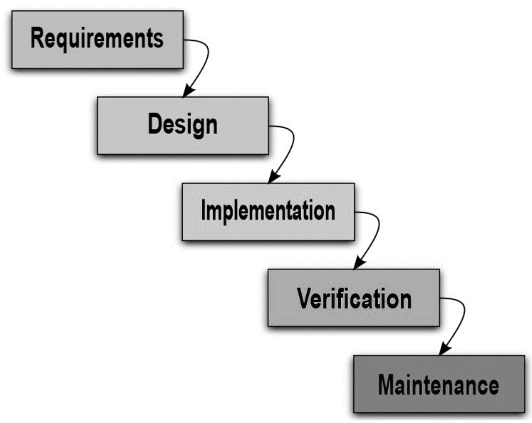


Figure - 2 Typical Waterfall Process

This process is advantageous to the projects, where requirements are pretty well defined at very early stage of the program and minimal changes are expected during the late phases of the project. It is well suited for small programs, where cost and schedule are the prime drivers. This process does not work well with large and complex programs, where requirements continue to evolve and/or not well understood at early stages of the program. This process is linear and rigid since it does not allow for executing more than one task simultaneously to improve the timeline and execution efficiency as well as less accommodating to late changes and modifications.

## V. "V" MODEL DEVELOPMENT PROCESS

V-Model brings the parity to the development and integration, verification and validation tasks, which is symbolized by the two sides of the "V".

The development process starts from the upper left corner and finishes at the upper right corner as shown in Figure-3. In the beginning of the process requirements, architecture blueprint and design rules are established and subsequent to this design process starts. In the right-hand side of the branch, testing, verification and validation tasks are performed. Like in

Waterfall-Model each phase must be completed before the next phase begins, V-model actually is a modified version of a Waterfall-Model. Like the Waterfall-Model testing process and fixing faults can be done at any stage in the life cycle, but the cost of finding and fixing faults increases dramatically as development process marches on. The quantity and intensity of the test levels may be modified according to the specific needs of the project and for every development stage there is a corresponding test level.

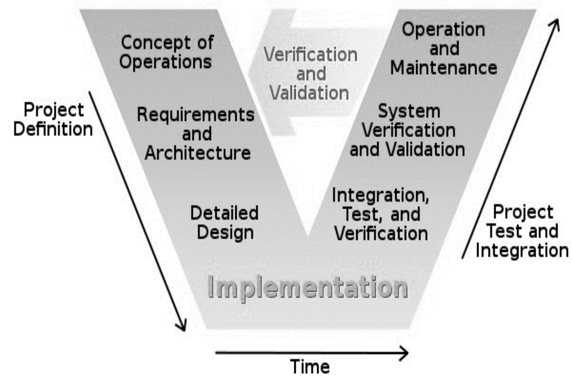


Figure - 3: Typical "V" Model Process

The bug fixes right after its detection mitigates the cost and schedule overrun. Everyone in the process has their stake in the program development and responsibility for quality assurance and testing lies with each and every one. Testing activities like requirements, test specifications, processes, tools, and environment are defined in detail before system design starts - this serves in developing a very good understanding of the project deliverables at the initial stage. Due to the tight coupling of test environment and design cycle, this model is very rigid and least flexible, as requirements changes would percolate through the test and design cycles. This model is suitable to large and complex programs, which requires large resources and exhaustive tracking mechanism. This model signifies the criticality of the testing during the development cycle for early detection and provides the cheaper alternative for fixing it.

## VI. SPIRAL MODEL PROCESS

This development process is iterative in nature and combines the features of the prototyping and the waterfall models. Iterative development process allows for incremental deliverables and milestones thus minimizes the risk of jeopardizing system objectives by delivering key components of the system to get customer feedback and buy-in at early stage of the program.

In spiral model testing, verification and validation are adapted to enable incremental development processes, and continuous integration testing and regression including reusability of the processes at each stages of the development. Figure-4 shows the process of spiral development pictorially.

Flexibility of the model allows the development to be tailored based on the complexity of the product. Since the issues are discovered earlier in the process it allows for better estimate and handle on cost and schedule milestones. It's good for large mission critical projects, where prototyping is essential to manage the risk of the project. The upfront cost and customization don't allow for much leverage for future products and generally requires domain specific expertise and know-how. In nutshell this may still serve as a good option for projects with unpredictability, which requires prototyping and risk mitigations.

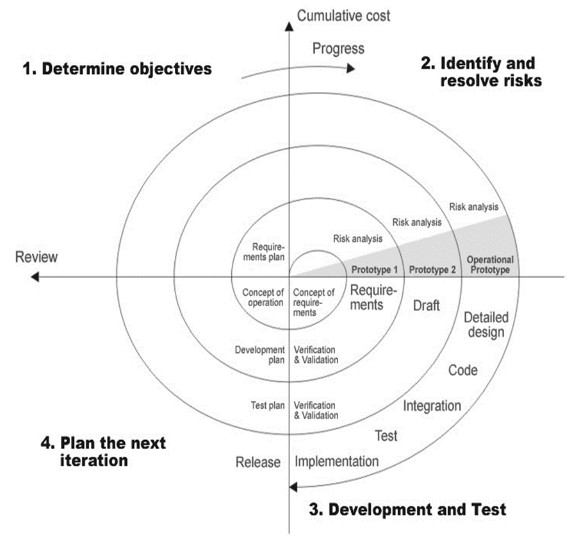


Figure - 4: Spiral Model Process

## VII. AGILE MODEL PROCESS

Agile model development is a framework for software engineering, which promotes design iterations throughout the life-cycle of the project. The goal is to analyze, test and validate the customer requirements as early as possible to get customer buy-in. Agile process improves the product quality throughout the product life cycle and well suited to the software development. The process thrives on short release cycles lasting few weeks and at each iteration the system requirements through track & monitor are traversed as shown in Figure – 5. The whole idea is to have a quick response to the requirements changes at each stages of the program. This saves time and money, gives the full visibility to the program managers and the end customers throughout all phases of the product development. The emphasis is on the development and thin on the documentation – the notion is to reduce the development time but still produce a quality product. This may not be suitable for large complex products as short milestones may not be possible and might incur a lot of resource wasting.

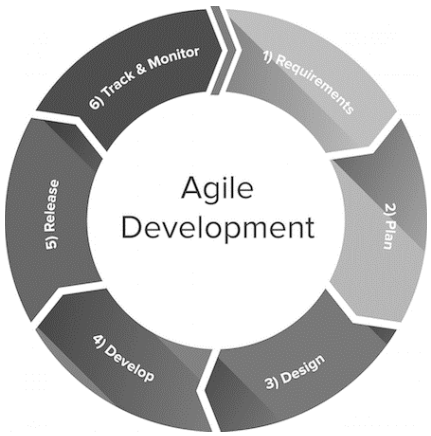


Figure – 5: Typical Agile Model

## VIII. MODELING & SIMULATION FRAMEWORK

Modeling and Simulation capabilities are developed to benefit all stakeholders comprising of systems, test, verification, and sell-off personnel. In a complex radar system development scenario a simulation framework must accommodate and add value across its user base and be able to scale and adapt based on stakeholder's emerging requirements. Special attention is also given to the heterogeneity of the SW development processes, disparate Hardware (HW) capabilities and plethora of interfaces so that it is able to reuse legacy modules and HW pieces to minimize the cost and resource requirements. At Raytheon we have utilized the development processes described in earlier sections to build the internal Modeling and Simulation capabilities, where Software in the Loop (SWIL), Hardware in the Loop (HWIL), Radar System in the Loop (RSIL), and Radar Mobil Test (RMT) bed environments plays a major role throughout the development cycle to minimize the development cost and accommodate any requirements changes and validation using the most appropriate method. One of the element, which plays an important role is the Radar Target Generator (RTG) described in the next section. The RTG allows for presenting a realistic target environment in the cost effective manner for algorithm development, performance verification, and some cases sell off to our customers.

## IX. RADAR TARGET GENERATOR

Increases to weapon system and threat complexity has limited the number of system requirements that can be cost effectively satisfied through traditional flight event sequence testing and has significantly added to the cost of live firing events for major defense programs. The use of high-fidelity radar target generation will allow extensive testing of system sensors in a controlled and repeatable environmental conditions.

The Radar Target Generator is a special purpose simulation tool to create in real time a high fidelity model of complex targets with multiple scattering points. Complex targets can be simulated with high range resolution scattering points with controllable amplitude, Doppler and range extent. This capability is especially useful in modeling targets for high resolution missile simulations. The current RTG, as shown in Figure-6, is designed to inject RF signal into the RF Processor (RFP) for each of the radar channels via an input coupler to test radar functionalities in real-time. The RTG design consists of two components: the digital Target Complex Generator (TCG) and the analog Radar System Interface Unit (RSIU). The TCG processes radar stimulus commands (Stims), received from the Radar Command and Control (RCC) computer, against a scenario of simulated targets, and computes the range, range rate, amplitude and phase for each scatterer. The RSIU generates uncompressed digital returns, converts the return digital signal representation to analog and then up-converts to RF for test injection into the radar receiver. The RF signal from RTG will combine with the system noise of the radar receiver. These real-time simulated signals can also be combined to the live target returns within the receiver to provide "sim-over-live" operation.

Since test targets are injected at RF directly into the RF processor, the RTG is required to translate digital targets formed at baseband up to the RF frequency band. This requires significant circuitry for signal conditioning to meet receiver requirements, as well as significant digital circuitry for timing and control functions.

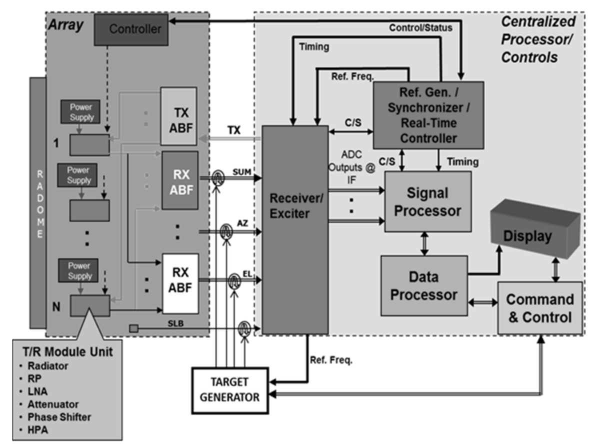


Figure -6: Radar system test interface with a RTG.

Also, system integration with RTG requires significant calibration effort to properly align the RF injection signal with the tactical processor. This alignment procedure is both time consuming and expensive and must be performed during each test event. This obviated the need to eliminate or significantly simplify the alignment procedure. Since RTG is not used as part of element test and evaluation (ET&E), it is not required that all system components or functionality be included when utilizing the RTG for system level test. For example, since the RTG is not intended to test specific antenna and/or receiver performance, the entire receive processing path from X-band through the Analog-to-Digital (A/D) converter may be bypassed. An initial assessment of these two improvement opportunities highlighted the benefit of eliminating the RF signal injection and moving to an all-digital RTG approach whereby the test target can be injected directly into the digital data stream as processed by the Digital Processor (DP). While a move from RF Injection of targets to Digital Injection of targets, the core mission of the RTG remains intact. Doing this eliminates the need for the analog conditioning module for frequency translation and thus eliminates the requirements to calibrate RTG prior to any test event. The RTG will continue to provide high fidelity target representations to support testing of Detection, Tracking, Characterization, and Discrimination Algorithms while closing the entire Radar Loop.

## X. CONCLUSIONS

Modeling and Simulation is one of the most versatile tools in the arsenal of the system developers, especially for complex system development. This allows for stakeholders to perform a comprehensive evaluation of system before it can be built in cost effective and timely manner. This also facilitates the customers and systems integrator to carry out the trade-off analysis, performance dimensioning, alternate design considerations, and future improvements considerations at very early stage of the program to minimize unnecessary risks and late stage surprises. The advances in technology, tools, and process have consistently enhanced the capabilities to model complex real time physical systems employing high performance distributed computers and broadband network with high fidelity input signal models and real life engagement scenario. The signal processing algorithms developed in quasi real time scenario allows system developer to design a system with right mix of cost, time, and performance offerings. These capabilities keep on improving as the digital and broadband technology continue to evolve following the Moore's law. System upgrades and performance enhancements can be established and demonstrated to key stake holders in a controlled and deterministic way without building a system. It is not farfetched that a system could be sold off solely on the basis of simulation results avoiding costly flight and real assets.


## ACKNOWLEDGMENT

Authors would like to recognize the support from the Raytheon's Systems and Software group as well as their valuable suggestions during the preparation of this manuscript.


## REFERENCES


1. The key role of business analysis in project success and achieving business value, Alan McSweeney, June 2012