Abstract

In today's world of producing large scale systems for customers in an environment where requirements are constantly evolving, it is imperative to develop and apply new and novel approaches to architecting our systems. In the past, the missions and requirements large scale systems needed to meet were reasonably stable over many years. The concept of operations and associated requirements were flowed one-way from customer to system developer, the typical staged system and architecture reviews were held, and this was sufficient to assess correctness of the architecture and design. Those systems, while complicated, were mostly independent entities with relatively simple interfaces to other systems, often integrated after the fact rather than intentionally designed to work together. Today's systems are an order of magnitude more complicated as the missions and associated requirements get more stressing, requirements to interface and collaboratively work together with other systems are a main focus [1], and computing hardware and software capabilities increase exponentially to meet these missions and requirements. To be able to ensure new systems being proposed and developed meet these new demanding needs, we must apply a multi-faceted approach to architecting to ensure we build the right systems for customers. This approach needs to include: 1) Advanced modeling techniques to capture the missions and multiple ‘stakeholders’ needs in their context, 2) Advanced system modeling techniques to capture the essence of the System needs and performance based on the defined missions and stakeholder’s needs, 3) Ability to automatically assess and re-assess those models for multiple architecture options against stakeholder’s needs including KPPs, TRL levels, cost, risk, quality, correctness, etc. with continually evolving requirements, 4) Ability to automatically assess those architecture models against sound architecting principles such as complexity, openness, internal consistency, composibility, etc., and 5) Ability to convey and visualize the proposed architectures and overall solution to the customer in their frame of reference. This paper will discuss these techniques and how they can be applied to modern system architecting.

1. Introduction

Modern large scale system development is an order of magnitude more difficult than system development of the past. “It is generally agreed that increasing complexity is at the heart of the most difficult problems facing today’s systems of architecting and engineering.” [2] Complexity grows due to systems being more interconnected and interdependent while trying to balance 8 key characteristics: Sustainable, Scalable, Safe, Smart, Stable, Simple, Secure, and Socially Acceptable [3]. Just looking at the software aspect of system development clearly shows this pattern. Figure 1 shows the exponential growth of software code size for NASA flight software.
Figure 1- Growth Trends in NASA Flight Software [4]

Figure 2 shows a similar story for avionics software.

Figure 2- SW Size in Avionics Software [5]

Software is not the focus of this paper, but these software statistics provide concrete evidence that our systems are much more complicated than they were in the past and that this trend is likely to continue in the future. That comes with unwanted consequences, specifically increased cost and the biggest
concern, increased system failures. Figure 3 shows that when complexity increases, cost increases along with failure rates.

![Total Flight System Cost as Function of Complexity](image)

Figure 3 - Total Flight System Cost as Function of Complexity [27]

To address these trends of increasing complexity and associated overruns in cost, we can apply new and modern approaches early on in the lifecycle of the program to address these. These are:

1) Advanced modeling techniques to capture the missions and multiple stakeholder’s needs in their context
2) Advanced system modeling techniques to capture the essence of the System needs and performance based on the defined missions and stakeholder’s needs
3) Ability to automatically assess and re-assess those models for multiple architecture options against stakeholder’s needs including KPPs, TRL levels, and cost, risk, etc. with continually evolving requirements
4) Ability to automatically assess those architecture models against sound architecting principles such as complexity, openness, internal consistency, composibility, etc.
5) Ability to convey and visualize the proposed architectures and overall solution to the customer in their frame of reference.

2. Modern Focus Areas

Modeling The Mission and Stakeholder’s Needs

This initial phase of modeling “sets the stage” for the entire program, and it is imperative to approach it with the correct focus. Upfront mission modeling is not new and we often put significant time into this effort when we perform Operations Analysis early on in the program lifecycle. What is new and novel in this area is bridging gap between the customer and the engineers implementing the system.
Our novel approach would build on this solid foundation of mission modeling efforts that are currently a best practice, but frame it and stage it to be better leveraged in the downstream system design and implementation phase. Modeling the customer’s missions requirements “offer engineer’s appropriate mechanism to understand what the customer want, analyze the requirements, assess feasibility, consult the rational solution, clearly point out the solution, confirm the specification, and reduce the possibility that the specific project may fail.” [13]

**Modeling the Systems Requirements in Context and Connected to the Mission**

As mentioned above, system requirements modeling is typically not associated with the full scope of the mission modeling. The mission requirements are taken as an input and often are not even provided with the full mission context. This creates a system requirements model that is stove-piped from the up-front materiel solution analysis. To greatly add value to the modeling of the system requirements, a clear connection of these requirements and how they tie to the mission model needs to be established. Without this linkage “each proposed design, whether traditional or not, must undergo the same, rigorous, time consuming and likely expensive process” [17] to understand if it meets the intended mission. Integrating different models: “Enables the viewing of complex relationships by multiple teams... enables requirements review by allowing diverse stakeholders to see all the necessary traces and artifacts on easily understood views and mitigates the problem of synchronization of artifacts (e.g. such as use cases and requirements) that are recorded and managed using different tools” [18]

**Modeling the System Architecture Options and Automatically Assessing Options Against the Customer Needs**

This third phase of modeling builds upon the first two, and creates a very powerful set of integrated models, allowing us to explore options with stakeholders in a highly integrated and automated manner. This is a leap in capability from the current best practice, and allows rapid and automated assessment of the mission, associated mission requirements, derived system requirements, and the architectural options of that system, all in a linked set of relationships.

These key aspects allow the solution space for the stakeholders to rapidly and continually be reassessed as the mission and the associated architecture and requirements change. “As the system evolves over time, the rules are checked in subsequent analyses to flag deviations from the architecture, usually introduced unwittingly during ongoing development.” [22] It also allows the system developer to explore the affordability of the customer’s mission helping them to fully understand the cost of each mission requirement and allows the system developer to suggest requirements relaxation where significant savings can be achieved. This is extremely important as Figure 4 shows, by Milestone B (typically Preliminary Design Review) 70% of the life cycle cost is committed, while less than 10% has been expended.
Assessing the Architecture Options against Common Architecture Quality Attributes

This is similar to the previous phase but focuses on quality attributes that are common to any architecture for any mission. While the acceptable values of each of these quality attributes vary from system to system, the quality attributes themselves are generally the same. All the same discussions of automatically assessing these attributes based on a particular architecture configuration that were mentioned above apply. The main reason to discuss this as a separate approach is the ability to apply this assessment in a common way against all architectures. They, in themselves, can be included into the overall architecture assessment discussed in Section 0. Some examples of architecture Quality Attributes include reliability, intentional and unintentional complexity, affordability, supportability, configurability, coupling, and cohesion.

Visualization of the Mission Modeling and Proposed Architecture Solution Against the Customer Requirements

In many aspects, this final phase of the overall approach is most important and may minimize the value of all the other facets of the approach if not implemented. In the end, if each architecture option and how it meets each aspect of the mission isn’t clearly conveyed, the customer cannot make the best informed decision. A set of requirements and analysis models providing raw output of whether it meets or fails certain requirements, and how that impacts the overall mission, is not an effective approach to visualizing the options to the customer. “Techniques of visual analytics are taken into account in order to make use of human abilities to detect e.g. irregular patterns in the behavior…. implementation in a virtual world can enhance the access to all relevant information.” [25].

3. Conclusion

While each phase of the approach here may not be new and novel, it is the integrated approach of all of them together in a unified effort to assess the proposed architectures and requirements against the mission that creates a new and powerful way to assess architectures. The focus is to reduce cost and improve performance for the system that will perform the customers’ mission. The primary way we accomplish this is by making the most informed decision up front between the customer and the system developer, and making sure all the criteria are balanced and presented to the customer for an informed decision. This primarily reduces cost and increases quality by: 1) Making the right decision up front for
the system architecture using tight integration between the System architecture, System requirements, and Mission needs 2) Providing an ability to automatically assess and reassess multiple architecture options against stakeholder needs as they continually change, and 3) Accurately and intuitively presenting those options to the customer enabling them to select the right choice based on the different modeling domains, reducing the chance of selecting the wrong architecture early on in the acquisition process.

**Bibliography**


