AIAA Space 2017

NASA's Space Launch System: Systems Engineering Approach for Affordability and Mission Success

Dr. John J. Hutt
John.j.hutt@nasa.gov
Dr. Josh Whitehead
James.whitehead@nasa.gov
Dr. John Hanson
john.m.hanson@nasa.gov

NASA is working toward the first launch of a new, unmatched capability for deep space exploration, with launch readiness planned for 2018. The initial Block 1 configuration of the Space Launch System will more than double the mass and volume to Low Earth Orbit (LEO) of any launch vehicle currently in operation – with a path to evolve to the greatest capability ever developed. The program formally began in 2011. The vehicle successfully passed Preliminary Design Review (PDR) in 2013, Key Decision Point C (KDPC) in 2014 and Critical Design Review (CDR) in October 2015 – nearly 40 years since the last CDR of a NASA human-rated rocket.

Every major SLS element has completed components of test and flight hardware. Flight software has completed several development cycles. RS-25 hotfire testing at NASA Stennis Space Center (SSC) has successfully demonstrated the space shuttle-heritage engine can perform to SLS requirements and environments. The five-segment solid rocket booster design has successfully completed two full-size motor firing tests in Utah. Stage and component test facilities at Stennis and NASA Marshall Space Flight Center are nearing completion. Launch and test facilities, as well as transportation and other ground support equipment are largely complete at NASA's Kennedy, Stennis and Marshall field centers. Work is also underway on the more powerful Block 1B variant with successful completion of the Exploration Upper Stage (EUS) PDR in January 2017.

NASA's approach is to develop this heavy lift launch vehicle with limited resources by building on existing subsystem designs and existing hardware where available. The systems engineering and integration (SE&I) of existing and new designs introduces unique challenges and opportunities. The SLS approach was designed with three objectives in mind: 1) Design the vehicle around the capability of existing systems; 2) Reduce work hours for non-hardware/software activities; 3) Increase the probability of mission success by focusing effort on more critical activities

Key aspects of the SLS SE&I approach include: 1) minimizing the number of requirements, 2) elimination of explicit verification requirements, 3) use of certified models of subsystem capability in lieu of requirements when appropriate and 4) certification of capability beyond minimum required capability.

The SLS Program faces a number of unique challenges. Among those are the complex interactions of systems developed primarily at three NASA centers with support from several others. Each center manages prime contractors in conjunction with in-house activities that involve a diverse workforce of civil servants and support contractors. Due to its high performance demands and mass limitations, SLS has high system sensitivity to subsystem interactions. For example, a change in a structural component to add address low structural margins can cause the vehicle structural dynamic response to interfere with a vehicle control system designed by another organization. These interactions could lead to software and/or hardware redesigns late in the design cycle, or they could lead to system failure if the change is not addressed in the final design. Additionally, the use of existing subsystem design can lead to unexpected problems due to interactions with new subsystems or changes in the level of interactions with the new system. The SLS Program has established an SE&I Operating Model to effectively engineer the system through its development, certification, and into the development.

Due to the breadth of the field and differences in terminology used in the industry, the role of systems engineering can be defined in many ways. SLS defines the scope for systems engineering (SE) as a specific discipline within SE&I. This definition does not rigidly conform to commonly-used references for SE&I guidance but implements the fundamental principles. In the SLS Program, SE&I can be defined as all engineering required to:

1) Define the SLS system to be designed to meet its requirements,

2) De-compose the system into hardware and software end items with assigned functionality. (End items are those at a level that can be assigned to an organization to design and develop with a manageable degree of interaction with the other end items.)

3) Manage the technical, cost and schedule interactions of the allocated end items,

4) Integrate the end item designs into a certified system design,

5) Integrate the end item hardware into a flight-certified system,

6) Support the operation of the system.

Because it is not practical for all of the engineers to be proficient in all engineering specialties or disciplines needed to perform these functions, the SLS SE&I model divides this work into several discipline scope groupings with lead engineers assigned to each. These lead engineers directly report to the SLS Chief Engineer. Assigning these disciplines at the system level and at the decomposed end item level leads to a two-dimensional responsibility matrix. This matrix provides a basis for assessing discipline-to-discipline effects at any level of the system (system, element, subsystem, component, part), as well as assessing the end item-to- end item interaction through any discipline. While this structure is not foolproof, it provides a framework for each of these interactions to be methodically assessed.

Although, there is nothing fundamentally unique in this concept, the SLS implementation has some specific features that make this SE&I approach more effective. One of these key features is that systems engineering is considered a "discipline" whose leader is a peer to the specialized discipline lead. Known as the Lead Systems Engineer (LSE), the SE discipline lead is also a direct report to the Chief Engineer. The LSE has the responsibility for the flow of technical information so that all interactions are documented. The LSE operates much like a football quarterback, a leader among peers to implement the direction of the Chief Engineer, acting as coach. The LSE must have broad understanding of the overall system design and engineering disciplines. The LSE applies this broad knowledge to ensure that all interactions are accounted for at an acceptable level of technical schedule and cost risk as design detail increases throughout development.

Because there are countless ways that subsystems can interact, the communication of hardware, software and discipline experts is the only practical way to adequately address all interactions. SLS makes no attempt to replace this interaction with models and processes, but the LSE relies very heavily on key processes and models and to effectively facilitate and capture the evaluations performed by these experts.

In order to engineer the SLS with limited resources, a number of general principles were utilized to establish how the work was to be performed and captured as part of the design and its supporting tests and analysis results:

1) Eliminate major development and verification tests where technical risk is acceptable
2) Allocate responsibility to the end item team for all content that can be performed without affecting other end items or the system
3) Eliminate explicit system level requirements allocated that duplicate other constraints in the system such as engineering models and drawings.
4) Deploy teams on specific tasks whose members are empowered to represent the affected end items and disciplines
5) Implement a flexible "systems view" of configuration management that comprehensively captures the system design details and changes

This paper will describe implementation of key aspects of SLS systems engineering, compare it to a typical systems engineering implementation, including a discussion of relative risk. Examples of each implementation within the SLS Program will be provided.